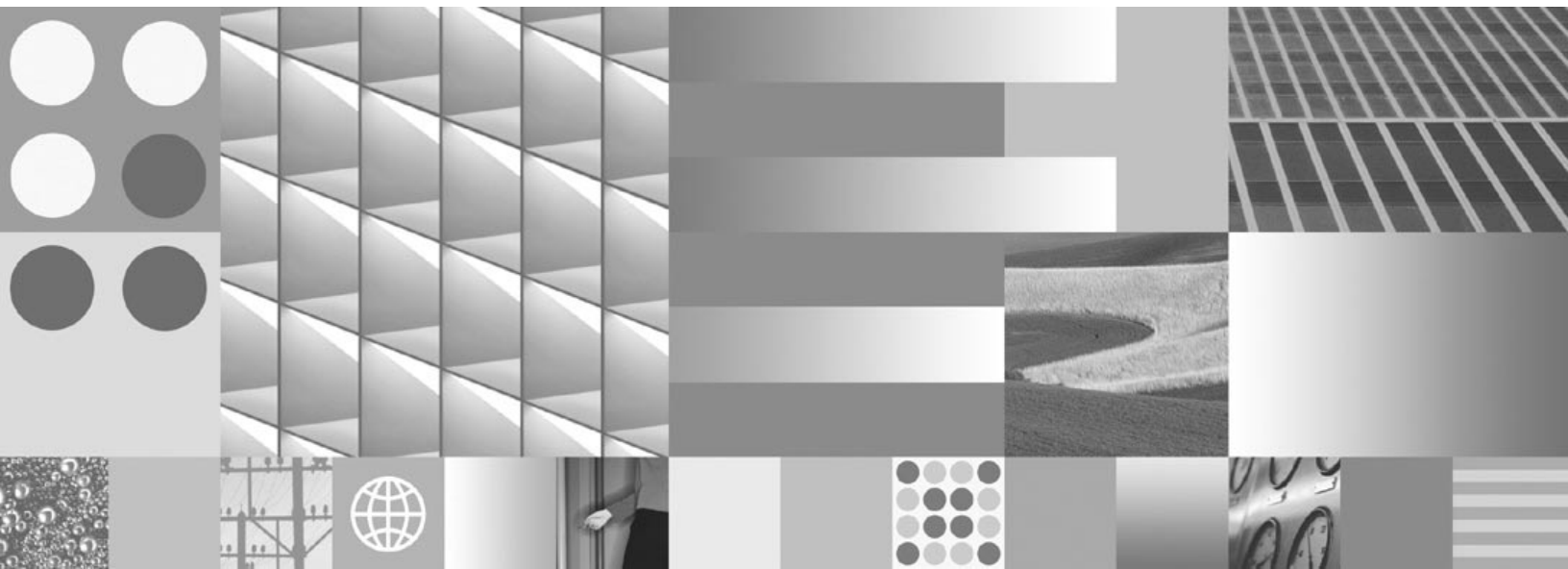


**System Tools Reference Manual**





**System Tools Reference Manual**

**Note**

Before using this information and the product it supports, read the information in "Notices" on page 1351.

This edition applies to version 4.1 of IBM FileNet Image Services (product number 5724-R95) and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 1984, 2008. All rights reserved.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

## About This Manual 16

<b>Manual Organization</b>	<b>17</b>
<b>Document revision history</b>	<b>18</b>
<b>What to Read First</b>	<b>19</b>
<b>Related Documents</b>	<b>20</b>
<b>Accessing IBM FileNet Documentation</b>	<b>21</b>
<b>Conventions</b>	<b>22</b>
CAUTIONs, Notes, and Tips	22
Platform Flags	23
Typing Instructions	23
Emphasis	24
Syntax	24
Parameters You Provide	24
Options	25
Computer Output	25
<b>IBM FileNet Education</b>	<b>26</b>
<b>Comments and Suggestions</b>	<b>27</b>

## Introduction 28

<b>Tools Overview</b>	<b>28</b>
<b>Subsection Descriptions</b>	<b>39</b>
Description	39

Use	39
Syntax	39
Flags and Options	39
Commands	39
Examples or Sample Output	40
Checklist	40
Procedure	40
Related Topics	40

## **Cross Reference 41**

<b>Backup Preparation and Analysis</b>	<b>41</b>
<b>Batches</b>	<b>41</b>
<b>Cache</b>	<b>42</b>
<b>Configuration</b>	<b>43</b>
<b>Core Files</b>	<b>43</b>
<b>Databases</b>	<b>44</b>
<b>Data Dictionary</b>	<b>45</b>
<b>Document Committal</b>	<b>45</b>
<b>Document Deletion</b>	<b>45</b>
<b>Document Services</b>	<b>46</b>
<b>Document Retrieval</b>	<b>46</b>
<b>Enterprise Backup/Restore (EBR) tools</b>	<b>47</b>
<b>Event Log</b>	<b>47</b>
<b>File Manipulation</b>	<b>48</b>

<b>FileNet Software</b>	<b>48</b>
<b>Index Database</b>	<b>49</b>
<b>Integral Single Document Storage (SDS)</b>	<b>50</b>
<b>Lightweight Directory Access Protocol (LDAP)</b>	<b>51</b>
<b>Logging</b>	<b>52</b>
<b>Memory</b>	<b>52</b>
<b>Messages</b>	<b>52</b>
<b>Microsoft SQL Server Databases</b>	<b>53</b>
<b>MKF Databases</b>	<b>53</b>
<b>MSAR Tools</b>	<b>54</b>
<b>National Language Translation</b>	<b>55</b>
<b>NCH</b>	<b>55</b>
<b>NCH Database</b>	<b>56</b>
<b>Oracle Databases</b>	<b>56</b>
<b>Performance</b>	<b>57</b>
<b>Permanent Database</b>	<b>58</b>
<b>Printing</b>	<b>59</b>
<b>Quick Logging</b>	<b>59</b>
<b>Remote Communications</b>	<b>60</b>
<b>Request Handlers</b>	<b>60</b>
<b>Storage Media</b>	<b>61</b>
<b>Security</b>	<b>62</b>
<b>Security Database</b>	<b>62</b>

**Space Use 63**  
**Storage Library Server 63**  
**System Information 64**  
**Tape and Tape Drives 64**  
**Transient Database 65**  
**Troubleshooting Tools 66**  
**Unified Logon Tools 67**  
**WorkFlo Database 68**

## **Tool Descriptions 69**

**911 69**  
**add\_osvr 126**  
**bes\_check 131**  
**bes\_clean 138**  
**bes\_debug 148**  
**bes\_setid 151**  
**check\_page 154**  
**checkwrt 158**  
**compressdir 160**  
**cordebug 163**  
**cormon 166**  
**CPT\_test 175**  
**cranuser 186**



<b>CSM_exim</b>	<b>190</b>
<b>CSM_tool</b>	<b>204</b>
<b>cstat</b>	<b>246</b>
<b>cti</b>	<b>249</b>
<b>dbp</b>	<b>252</b>
<b>dbverify</b>	<b>255</b>
<b>dclview</b>	<b>262</b>
<b>ddexim</b>	<b>266</b>
<b>deldocs</b>	<b>282</b>
<b>del_osvr</b>	<b>288</b>
<b>dialout</b>	<b>295</b>
<b>docchk</b>	<b>303</b>
<b>doccnt</b>	<b>310</b>
<b>DOC_dumpfb</b>	<b>314</b>
<b>docfetch</b>	<b>324</b>
<b>DOC_tool</b>	<b>326</b>
<b>EBR_clean</b>	<b>379</b>
<b>EBR_genscript</b>	<b>379</b>
<b>EBR_orreset</b>	<b>380</b>
<b>EBR_tdir</b>	<b>380</b>
<b>EBR_ulmk</b>	<b>380</b>
<b>eladisp</b>	<b>381</b>
<b>endbackup</b>	<b>401</b>

<b>enlarge_ncol</b>	<b>403</b>
<b>export_cdb</b>	<b>409</b>
<b>FileNet System Manager</b>	<b>414</b>
<b>flat</b>	<b>417</b>
<b>fn_edit</b>	<b>447</b>
<b>fn_ldif_xfer</b>	<b>448</b>
<b>fn_msg</b>	<b>464</b>
<b>fn_perm</b>	<b>466</b>
<b>fn_pso_driver</b>	<b>467</b>
<b>fn_pso_podf_admin</b>	<b>469</b>
<b>fn_pso_switch</b>	<b>474</b>
<b>fn_setup</b>	<b>478</b>
<b>fn_util</b>	<b>478</b>
<b>fnddcfg</b>	<b>479</b>
<b>fndev</b>	<b>480</b>
<b>FNL_disp</b>	<b>481</b>
<b>fnlogon</b>	<b>489</b>
<b>gaddr</b>	<b>493</b>
<b>gcp</b>	<b>498</b>
<b>GDBcheckdb</b>	<b>503</b>
<b>GDBdebug</b>	<b>505</b>
<b>GDB_exim</b>	<b>511</b>
<b>get_rdbms</b>	<b>519</b>

<b>getreports</b>	<b>523</b>
<b>getstamps</b>	<b>532</b>
<b>getstatus</b>	<b>546</b>
<b>glogin</b>	<b>565</b>
<b>gls</b>	<b>568</b>
<b>gsh</b>	<b>570</b>
<b>ident</b>	<b>572</b>
<b>import_cdb</b>	<b>576</b>
<b>initbackup</b>	<b>579</b>
<b>initfnsw</b>	<b>583</b>
<b>INXdebug</b>	<b>593</b>
<b>INX_export</b>	<b>595</b>
<b>INX_tool</b>	<b>598</b>
<b>ipc_tool</b>	<b>610</b>
<b>ixdb_stat</b>	<b>643</b>
<b>kermit</b>	<b>650</b>
<b>killfnsw</b>	<b>656</b>
<b>ldap_exp</b>	<b>662</b>
<b>LDAP_import</b>	<b>671</b>
<b>LDAP_password</b>	<b>690</b>
<b>less</b>	<b>692</b>
<b>log_create</b>	<b>696</b>
<b>log_dir</b>	<b>699</b>

<b>log_extract</b>	<b>702</b>
<b>manifest</b>	<b>706</b>
<b>migrate_cdb_changes</b>	<b>709</b>
<b>MKF_ddl</b>	<b>712</b>
<b>MKF_debug</b>	<b>721</b>
<b>MKF_dump</b>	<b>738</b>
<b>MKF_export</b>	<b>756</b>
<b>MKF_fixup</b>	<b>764</b>
<b>MKF_import</b>	<b>770</b>
<b>MKF_order</b>	<b>780</b>
<b>MKF_rename</b>	<b>785</b>
<b>MKF_shutdown</b>	<b>790</b>
<b>MKF_startup</b>	<b>793</b>
<b>MKF_stats</b>	<b>795</b>
<b>MKF_tool</b>	<b>801</b>
<b>MKF_verify</b>	<b>816</b>
<b>MKF_zeroaij</b>	<b>820</b>
<b>MKF_zerobij</b>	<b>823</b>
<b>move_disk</b>	<b>826</b>
<b>msar_io_test</b>	<b>829</b>
<b>msar_sync_test</b>	<b>831</b>
<b>nch_check</b>	<b>832</b>
<b>nch_flush</b>	<b>839</b>

<b>nch_tool</b>	<b>841</b>
<b>NLT_build_maps</b>	<b>872</b>
<b>ntdm_exp</b>	<b>881</b>
<b>oddump</b>	<b>886</b>
<b>odrecover</b>	<b>932</b>
<b>oraloglist</b>	<b>940</b>
<b>osschk</b>	<b>942</b>
<b>perf_mon</b>	<b>954</b>
<b>perf_report</b>	<b>968</b>
<b>ppm_log</b>	<b>1028</b>
<b>PPMOI</b>	<b>1030</b>
<b>PRI_tool</b>	<b>1056</b>
<b>QLG_dump</b>	<b>1075</b>
<b>QLG_flush</b>	<b>1078</b>
<b>QLG_start</b>	<b>1080</b>
<b>remove_docs_ref</b>	<b>1084</b>
<b>remove_surf</b>	<b>1095</b>
<b>rollcall</b>	<b>1100</b>
<b>RSEC_imp</b>	<b>1103</b>
<b>SC_convert</b>	<b>1116</b>
<b>SEC_debug</b>	<b>1120</b>
<b>SEC_imp</b>	<b>1129</b>
<b>SEC_init</b>	<b>1142</b>

<b>SEC_map</b>	<b>1146</b>
<b>SEC_rm_config</b>	<b>1155</b>
<b>SEC_tool</b>	<b>1163</b>
<b>sgs</b>	<b>1183</b>
<b>SNT_update</b>	<b>1196</b>
<b>spacerpt</b>	<b>1203</b>
<b>sqlplus</b>	<b>1215</b>
<b>ssn</b>	<b>1231</b>
<b>st_msar_convert</b>	<b>1232</b>
<b>stamp</b>	<b>1234</b>
<b>stampro</b>	<b>1236</b>
<b>stdoccpy</b>	<b>1238</b>
<b>stdocimp</b>	<b>1249</b>
<b>stmigrate</b>	<b>1257</b>
<b>stsurfupdate</b>	<b>1262</b>
<b>sync_write_test</b>	<b>1267</b>
<b>sys_log</b>	<b>1274</b>
<b>tapeser</b>	<b>1276</b>
<b>TAP_tool</b>	<b>1280</b>
<b>test_raw_partition</b>	<b>1284</b>
<b>TLIB_tool</b>	<b>1293</b>
<b>uncompressdir</b>	<b>1294</b>
<b>vl</b>	<b>1297</b>

**WALSEC\_debug 1300**

**whatsup 1309**

**WQMpack 1315**

**WQS\_move\_q 1321**

**WQS\_tool 1325**

**WRT\_clean 1341**

**XPR\_print 1346**

## **Notices 1351**

**Trademarks 1354**

**U.S. Patents Disclosure 1355**

## **Index 1356**

# About This Manual

The *FileNet Image Services System Tools Reference Manual* for Release 4.1 describes system tools you use to diagnose and manage your IBM® FileNet® Image Services (IS) software. Some tools are used only by support personnel whereas many others are used by system administrators to analyze, diagnose, update, and repair Image Services components.

This section describes the following:

- [“Manual Organization” on page 17](#)
- [“Document revision history” on page 18](#)
- [“What to Read First” on page 19](#)
- [“Related Documents” on page 20](#)
- [“Accessing IBM FileNet Documentation” on page 21](#)
- [“Conventions” on page 22](#)
- [“IBM FileNet Education” on page 26](#)
- [“Comments and Suggestions” on page 27](#)



## Manual Organization

This manual is divided into the following main sections:

- **“Introduction” on page 28** lists and briefly describes all the tools in this manual. Tools are listed in alphabetical order. In addition, this section describes the format of the individual tool subsections and the syntax conventions used throughout the manual.
- **“Cross Reference” on page 41** provides a list of topics and procedures cross referenced by tools.
- **“Tool Descriptions” on page 69** describes each tool in detail. Tool names appear in alphabetical order. A general tool description is followed by information on when to use the tool, its syntax, sample output and examples, important precautions, and references to related tools and manuals.

## Document revision history

IS version	Date	Comment
4.1	June 2008	Documentation refresh. Incorporated changes to dbverify. Updated DOC_tool cache hits info.
4.1	Nov. 2007	Documentation refresh. Bluewashing.
4.1	June 2007	Initial release.

## What to Read First

We suggest that you read “About this Manual” (this section) followed by [“Introduction” on page 28.](#)

When you have isolated a problem and want to know which tool to use to continue your analysis or problem resolution, refer to [“Cross Reference” on page 41.](#)

When you have identified the tools you want to use, read the detailed information in [“Tool Descriptions” on page 69](#)

## Related Documents

You might also refer to the following documents as you use the tools to analyze and diagnose your system. Use the information above for accessing the IBM FileNet documentation to find the following documents:

*System Administrator's Handbook*

*System Administrator's Companion for UNIX*

*System Administrator's Companion for Windows Server*

*System Configuration Overview*

*System Messages Manual*

*Enterprise Backup/Restore User's Guide*

*Index and WorkFlo Database Contents Manual*

*MSAR Procedures and Guidelines*

Reference materials provided by your RDBMS vendor (for example, Oracle, IBM DB2®, or Microsoft® SQL Server).

## Accessing IBM FileNet Documentation

To access documentation for IBM FileNet products:

1. Navigate to the Information Management support page ([www.ibm.com/software/data/support](http://www.ibm.com/software/data/support)).
2. Select the appropriate IBM FileNet product from the “Select a category” list.
3. From the Product Support page, click Product Documentation under Learn.
4. From the Product Documentation page
  - a. If necessary, click the Doc Link for the appropriate component product to display the document list.
  - b. Click the icon in the appropriate release column to access the document you need.

## Conventions

Media is any material on which data is stored (magnetic disk, optical disk, magnetic tape). When we refer to storage media, we are generally discussing optical disks or MSARs.

We call your attention to information throughout this manual using the following conventions.

### CAUTIONs, Notes, and Tips

To identify important information, this manual uses the following message types:

---

**CAUTION** CAUTION boxes signal possible unexpected consequences of an action, such as loss of data or time.

---

---

**Note** Note boxes draw your attention to essential information that you should be sure to read.

---

---

**Tip** Tip boxes introduce ideas that might make your work easier.

---

## Platform Flags

To help you identify information that applies to a specific Image Services server platform, this manual uses the following labeled flags:

**UNIX**

This flag identifies information that applies only to UNIX®-based Image Services systems.

**WIN**

This flag identifies information that applies only to Image Services for Windows® Server systems.

**HPUX**

This flag identifies information that applies specifically to HP-UX-based Image Services systems.

**AIX**

This flag identifies information that applies specifically to AIX-based Image Services systems.

**SOL**

This flag identifies information that applies specifically to Solaris Operating Environment-based Image Services systems.

## Typing Instructions

To indicate commands, values, or other information you enter at your keyboard, we use the following indentation and typeface:

**CSM\_tool**

If the command is too long, but you must type it continuously with no carriage returns, we use the following style:

```
select owner, table_name from all_tables where owner='F_SW'  
or owner='F_SQI';
```

Each line following the first is indented.

**Note** When typing a lengthy command on a UNIX command line, you must type a backslash (\) character (called a continuation character) on all but the last line to indicate that the command continues.

---

## Emphasis

Bold typeface within text emphasizes an individual word or phrase. Take special note of bold text, as in the following example:

If you do not specify a directory, compression is applied to **all** files starting with the **current directory**.

## Syntax

Syntax definitions are indented from the body text:

```
bes_check
```

### Parameters You Provide

Parameters that require you to provide information are shown within angle brackets (< >):

```
Checksum <cache_id> <ssn> <object_id> <page>
```



## Options

Optional parameters and keywords are within square brackets:

```
CREATEobjects <cache_id> <ssn> <start_object_id> [<number_
objects> [<bytes>]]
```

When a choice of available options within a parameter is required, the available choices are separated by a vertical bar within square, curly, or angle brackets. In the following example, the choice is either `-p` or `-c`:

```
cormon [-p | -c]
```

## Computer Output

This manual shows console displays (such as file contents, system messages, or output from program execution) as shown in the following example:

```
corona(root)> bes_debug
Batch Services Bes:Corona:FileNet
bes_debug completed: 0 batch found in inprogress queue.
bes_debug completed successfully.
```

For lengthy console displays, this manual bounds the sample output with a single bold line above the beginning of the report and below the end of the report.

## **IBM FileNet Education**

IBM FileNet provides various forms of education. Please visit Global Learning Services on IBM's Web site at ([www-306.ibm.com/software/sw-training/](http://www-306.ibm.com/software/sw-training/)).

## Comments and Suggestions

Send your comments by e-mail to [comments@us.ibm.com](mailto:comments@us.ibm.com). Be sure to include the name of the product, the version number of the product, and the name and part number of the book (if applicable). If you are commenting on specific text, include the location of the text (for example, a chapter and section title, a table number, a page number, or a help topic title).

# Introduction

This Introduction lists and briefly describes the system tools available with release 4.0 of the Image Services software. Tools run under all supported Image Services platforms unless otherwise specified.

To help you find the right tool for your task, refer to [“Cross Reference” on page 41](#). You can find information about each tool in [“Tool Descriptions” on page 69](#), which includes a detailed subsection for each tool. (See [“Subsection Descriptions” on page 39](#) for the subsection format.)

## Tools Overview

The following is an alphabetical list of Image Services system tools. A brief description of its function accompanies each tool.

Tool	Description
<a href="#"><u>911</u></a>	911 is a shell script that takes a quick snapshot of an Image Services system in crisis. (Runs on both UNIX-based and Windows Server-based IS servers)
<a href="#"><u>add_osvr</u></a>	Adds a Storage Library server to the system
<a href="#"><u>bes_check</u></a>	Makes a copy of an archive database if media becomes unreadable or consolidation of the media on which the archive databases reside becomes necessary (Runs under Image Services for AIX/6000 only)
<a href="#"><u>bes_check</u></a>	Performs a consistency check between objects in BES cache and records in the transient database batch tables
<a href="#"><u>bes_clean</u></a>	Deletes all or selected records in the transient database batch tables

Tool	Description
<a href="#"><u>bes_debug</u></a>	Displays status of batches in the “in progress” queue (for example, the type of service for which the batch is waiting)
<a href="#"><u>bes_setid</u></a>	Updates the batch_id or batch_name_id values in the batch control table of the transient database
<a href="#"><u>check_page</u></a>	Provides diagnostic information about an image
<a href="#"><u>checkwrt</u></a>	Verifies pages in the cache that are associated with a pending write request
<a href="#"><u>compressdir</u></a>	Compresses all files in a directory  (Does not run on Image Services for Windows Server or Image Services for the Solaris Operating Environment platforms)
<a href="#"><u>cordebug</u></a>	Displays COR handler request information
<a href="#"><u>cormon</u></a>	Monitors the current state of the client-server connection on the FileNet system  (Does not run on Image Services for Windows Server platforms)
<a href="#"><u>CPT_test</u></a>	Collects Courier performance test information  (Does not run on Image Services for Windows Server platforms)
<a href="#"><u>cranuser</u></a>	Supports ISRA/LDAP customers with anonymous logins
<a href="#"><u>CSM_exim</u></a>	Exports and imports objects in cache
<a href="#"><u>CSM_tool</u></a>	Provides commands to manipulate cache and obtain cache statistics
<a href="#"><u>cstat</u></a>	Reads a core file to determine the program that failed and created the core file  (Does not run on Image Services for Windows Server platforms)
<a href="#"><u>cti</u></a>	Counts the number of unique index values in an Oracle database
<a href="#"><u>dbp</u></a>	Displays diagnostic information from Document Services
<a href="#"><u>dbverify</u></a>	Verifies that records in DOCTABA and DOCS tables compare correctly
<a href="#"><u>dclview</u></a>	Creates views on DOCTABA corresponding to document classes
<a href="#"><u>ddexim</u></a>	Saves and restores document class and index information found in the index database

Tool	Description
<a href="#"><u>deldocs</u></a>	Deletes documents from index and/or permanent databases
<a href="#"><u>del_osvr</u></a>	Deletes a Storage Library server from the system, or moves media from one Storage Library server to another
<a href="#"><u>dialout</u></a>	Connects a FileNet server to a serial port. When connected to a modem, provides dial out capability from a central site to a remote FileNet system  (Does not run on Image Services for Windows Server platforms)
<a href="#"><u>docchk</u></a>	Checks documents on media for valid format
<a href="#"><u>doccnt</u></a>	Counts and verifies the number of documents in database tables
<a href="#"><u>DOC_dumpfb</u></a>	Provides information on fast batch objects in cache, including document ids, cache locations, and hexadecimal dumps
<a href="#"><u>docfetch</u></a>	Retrieves an image into page cache
<a href="#"><u>DOC_tool</u></a>	Provides statistics on media and in slots in a Storage Library server
<a href="#"><u>EBR_clean</u></a>	Reclaims shared memory and interlocks for a failed EBR backup or restore
<a href="#"><u>EBR_genscript</u></a>	Generates EBR dataset definition files, device specification files, and builds EBR backup and restore scripts
<a href="#"><u>EBR_label</u></a>	Labels and relabels EBR backup tapes in a stand-alone tape drive or tape library and disk files used for backup
<a href="#"><u>EBR_orreset</u></a>	Resets an Oracle database, which has terminated abnormally, to a working state
<a href="#"><u>EBR_tdir</u></a>	Displays EBR tape label on a tape in a stand-alone tape drive or tape library, or on a disk file
<a href="#"><u>EBR_ulmk</u></a>	Unlocks MKF databases locked for an off-line EBR backup that did not complete
<a href="#"><u>eladisp</u></a>	Displays usage statistics collected by Event Logging Abstract (ELA)
<a href="#"><u>endbackup</u></a>	Terminates current processes and returns a server to multi-user mode after a backup operation
<a href="#"><u>enlarge_ncol</u></a>	Increases the precision and scale of an existing numeric column in a Microsoft SQL Server database and a DB2 database

Tool	Description
<a href="#"><u>export_cdb</u></a>	Copies the latest Configuration Database (.cdb) file as part of the first step in the process of cloning the IS system configuration of multiple IS systems from a single master .cdb file
<a href="#"><u>FileNet System Manager</u></a>	Collects performance information from the FileNet products. It is tool tied to a variable called <b>listener</b> in the set command of the <a href="#"><u>perf_mon</u></a> tool.
<a href="#"><u>flat</u></a>	FileNet LAN analysis tool (flat) analyzes output from network analyzer programs and network packet traces  (Does not run on Image Services for Windows Server platforms)
<a href="#"><u>fn_edit</u></a>	Invokes a graphical user interface (GUI) utility called the FileNet System Configuration Editor
<a href="#"><u>fn_ldif_xfer</u></a>	Gives customers the ability to use their LDAP tools to generate the LDIF files for use with FileNet software
<a href="#"><u>fn_msg</u></a>	Decodes an error message tuple
<a href="#"><u>fn_perm</u></a>	Sets up file and directory permissions on Windows servers
<a href="#"><u>fn_pso_driver</u></a>	Used exclusively during the Programmable Security Object recovery process to convert all standard SQL scripts and system scripts into customized scripts
<a href="#"><u>fn_pso_podf_admin</u></a>	Used exclusively during the Programmable Security Object recovery process to directly update the master Programmable Object Data File (PODF)
<a href="#"><u>fn_pso_switch</u></a>	Enables you to change the user name of the standard FileNet software user
<a href="#"><u>fn_setup</u></a>	Sets the permissions for all files under /fnsw and /fnsw/local directories
<a href="#"><u>fn_util</u></a>	Starts, stops, creates, and updates the MKF and RDBMS databases
<a href="#"><u>fnddcfg</u></a>	Resets passwords on FileNet user IDs
<a href="#"><u>fnddcfg</u></a>	Configure FnScsidd
<a href="#"><u>fndev</u></a>	List SCSI optical disk drives
<a href="#"><u>FNL_disp</u></a>	Displays interlocks (ILK) and process control records (PCRs)
<a href="#"><u>fnlogon</u></a>	Provides interface to security service  (Does not run on Image Services for Windows Server platforms)

Tool	Description
<a href="#"><u>gaddr</u></a>	Displays addresses of host machines accessible from glogin or gsh tools (Supported on Image Services for AIX/6000 systems only)
<a href="#"><u>gcp</u></a>	Copies files between two systems on the same or different local area networks
<a href="#"><u>GDBcheckdb</u></a>	Called internally by FileNet startup software to synchronize the RDBMS SQL server
<a href="#"><u>GDBdebug</u></a>	Collects GDB data for problem determination (Intended for use <b>only</b> at the direction of your service representative)
<a href="#"><u>GDB_exim</u></a>	Exports and imports databases and tables between RDBMSs (Not intended for customer use)
<a href="#"><u>get_rdbms</u></a>	Collect performance and configuration statistics on RDBMS databases (Does not run on Image Services for Windows Server platforms)
<a href="#"><u>getreports</u></a>	Generates performance reports (Does not run on Image Services for Windows Server platforms)
<a href="#"><u>getstamps</u></a>	Collects stamps for all subsystems on the Image Services server on which you run the program. This tool also compares the stamps with the current database of SCRs that identify the available system fixes for a specified platform and Image Services software release.
<a href="#"><u>getstatus</u></a>	Collects performance and configuration statistics on FileNet datasets and MKF databases (Does not run on the Image Services for Windows Server platforms)
<a href="#"><u>glogin</u></a>	Provides login capability to a remote system (Runs on Image Services for AIX/6000 platforms only)
<a href="#"><u>gls</u></a>	Lists the directories of remote systems (Runs on Image Services for AIX/6000 platforms only)
<a href="#"><u>gsh</u></a>	Runs commands on a remote system (Runs on Image Services for AIX/6000 platforms only)



Tool	Description
<a href="#"><u>ident</u></a>	Locates and lists a file by a specified date, author, header, etc.
<a href="#"><u>import_cdb</u></a>	Imports the latest Configuration Database (.cdb) file into the IS configuration directory as part of the last step in the process of cloning the IS system configuration of multiple IS systems from a single master .cdb file
<a href="#"><u>initbackup</u></a>	Prepares a server's processes for backup
<a href="#"><u>initfns</u></a>	Stops and starts the FileNet software without rebooting in preparation for a backup
<a href="#"><u>INXdebug</u></a>	Provides performance analysis data about index services
<a href="#"><u>INX_export</u></a>	Exports existing document index information associated with an IS document class to make it available for import by a Content Engine (CE) system (CFS-IS)
<a href="#"><u>INX_tool</u></a>	Displays and releases index services (INX) capability locks
<a href="#"><u>ipc_tool</u></a>	Provides information on the addresses used by abstracts, the processes that call abstracts, system interlocks, and semaphores
<a href="#"><u>ixdb_stat</u></a>	Gathers statistical information about the index database
<a href="#"><u>kermit</u></a>	Performs file transfer between systems
<a href="#"><u>killfns</u></a>	Terminates FileNet software
<a href="#"><u>ldap_exp</u></a>	Exports the LDAP-based security information to an XML formatted ASCII file.
<a href="#"><u>LDAP_import</u></a>	Use in conjunction with the ldap_exp tool and Web Services that makes the security centralization and authentication solution possible.
<a href="#"><u>LDAP_password</u></a>	Use to encrypt a password that will be used by the LDAP_import tool during the import process, so you can provide user name and password through the command line.
<a href="#"><u>less</u></a>	Displays a file, with capability to move backward or forward through the file contents
<a href="#"><u>log_create</u></a>	Creates a circular log file, or increase the size of an existing log file
<a href="#"><u>log_dir</u></a>	Displays the records within a circular log file

Tool	Description
<a href="#"><u>log_extract</u></a>	Extracts specific records, or a range of records, from a circular log file
<a href="#"><u>manifest</u></a>	Maintains the file that specifies the server software release partition contents (Use is restricted to support personnel)
<a href="#"><u>migrate_cdb_changes</u></a>	Updates the performance-related parameters in the clone system's configuration database (.cdb) from a specified file as part of the last step in the process of cloning the IS system configuration of multiple IS systems from a single master .cdb file
<a href="#"><u>MKF_ddl</u></a>	Initializes or updates the transient, permanent, or Network Clearinghouse databases
<a href="#"><u>MKF_debug</u></a>	Displays a list of all processes linked to the MKF shared library and the status of all known MKF databases on the server for use in diagnosing database hangs
<a href="#"><u>MKF_dump</u></a>	Displays data in MKF databases and after-image files for use in diagnosing and correcting inconsistencies
<a href="#"><u>MKF_export</u></a>	Copies data in an MKF database to a magnetic disk file or to a magnetic tape file
<a href="#"><u>MKF_fixup</u></a>	Performs an in-place regeneration of a corrupted MKF database
<a href="#"><u>MKF_import</u></a>	Inserts data gathered by MKF_export into an MKF database
<a href="#"><u>MKF_order</u></a>	Ensures that byte ordering of a database is appropriate for a given CPU
<a href="#"><u>MKF_rename</u></a>	Performs an in-place reconfiguration of the database and zeroes out the recovery logs
<a href="#"><u>MKF_shutdown</u></a>	Shuts down either the permanent or transient database prior to a tape backup operation
<a href="#"><u>MKF_startup</u></a>	Restarts a permanent or transient database that has been shut down
<a href="#"><u>MKF_stats</u></a>	Gathers statistical information about the use of index and data space by the permanent database
<a href="#"><u>MKF_tool</u></a>	Provides commands for viewing and editing tables in the permanent and transient databases
<a href="#"><u>MKF_verify</u></a>	Searches for and reports on corruption between the MKF btrees and the data

Tool	Description
<a href="#"><u>MKF_zeroaij</u></a>	Clears any data that exists in either the transient or permanent after-image recovery logs
<a href="#"><u>MKF_zerobj</u></a>	Clears any data that exists in either the transient or permanent before-image recovery logs
<a href="#"><u>move_disk</u></a>	Moves media from one Storage Library server to another
<a href="#"><u>msar_io_test</u></a>	Determines relative speed of I/O for a specific directory
<a href="#"><u>msar_sync_test</u></a>	Program has been enhanced and name has change to <a href="#"><u>sync_write_test</u></a>
<a href="#"><u>nch_check</u></a>	Verifies that the NCH_daemon program is running and that the NCH database is providing service for the server's default domain
<a href="#"><u>nch_flush</u></a>	Removes the in-memory version of the NCH database
<a href="#"><u>nch_tool</u></a>	Provides information on all system resources (hardware and software) defined in the Network Clearinghouse database
<a href="#"><u>NLT_build_maps</u></a>	Translates a user-defined character set to the system-defined character set
<a href="#"><u>ntdm_exp</u></a>	Exports the user names and group names from a Windows Server domain to an intermediate XML formatted file in an effort to support group/user mappings between Windows Server domain groups and IS groups (Unified Logon). Used in conjunction with the RSEC_imp or SEC_imp tool.
<a href="#"><u>oddump</u></a>	Reads and repairs media problems
<a href="#"><u>odrecover</u></a>	Imports documents from media for use in rebuilding damaged or corrupted permanent or index databases
<a href="#"><u>oraloglist</u></a>	Displays archive log mode
<a href="#"><u>osschk</u></a>	Provides summary information contained in FileNet error logs for optical storage libraries (OSARs)
<a href="#"><u>perf_mon</u></a>	Collects performance data
<a href="#"><u>perf_report</u></a>	Generates reports from the output of perf_mon
<a href="#"><u>ppm_log</u></a>	Writes the contents of a PPM log buffer to a file

Tool	Description
<b><u>PPMOI</u></b>	Serves as the operator interface to the Protocol Process Manager (PPM) abstract. You can use PPMOI to temporarily increase the allowable number of request handlers that can be started on a server.
<b><u>PRI_tool</u></b>	Provides commands to manage printers and print requests, and to obtain statistical information on FileNet printers
<b><u>QLG_dump</u></b>	Displays quick logging data from the log file
<b><u>QLG_flush</u></b>	Forces log records in QLG's in-memory buffer out to log files
<b><u>QLG_start</u></b>	Initiates quick logging of usage statistics
<b><u>remove_docs_ref</u></b>	Removes any and all references to the specified surfaces from the PermDB document locator (doc locator) database table.
<b><u>remove_surf</u></b>	Removes the specified surface from the surf_info Permanent MKF DB tables, family_disk write surfaces list (current and future write surfaces), Permanent MKF DB table, surf_locator Permanent MKF DB table, lib_surfaces Permanent MKF DB table, OSA shared memory, SRF shared memory and OSA checkpoint file.
<b><u>rollcall</u></b>	Lists systems running on the network  (Runs on Image Services for AIX/6000 systems only)
<b><u>RSEC_imp</u></b>	Imports the Windows Server domain security information into the IS Security Service in an effort to support group/user mappings between Windows Server domain groups and IS groups (Unified Logon). This tool must be run on a Windows Server machine with no Image Services software present, but that has either Web Service or IDM Desktop installed. Used in conjunction with the ntdm_exp tool.
<b><u>SEC_debug</u></b>	Enables support personnel to trace execution within the Security Services subsystem of the Image Services server.
<b><u>SEC_imp</u></b>	Imports the Windows Server domain security information into the IS Security Service in an effort to support group/user mappings between Windows Server domain groups and IS groups (Unified Logon). This tool must be run on an Image Services server (Windows Server or UNIX). Used in conjunction with the ntdm_exp tool.

Tool	Description
<a href="#"><u>SEC_init</u></a>	Rebuilds or initializes the security database
<a href="#"><u>SEC_tool</u></a>	Provides debugging information for the security services system
<a href="#"><u>sgs</u></a>	Sets the password for the f_maint user
<a href="#"><u>sgs</u></a>	Lists the system configuration settings on the Image Services server (Runs only on a UNIX-based Image Services server)
<a href="#"><u>SNT_update</u></a>	Updates the scalar_numbers table of the permanent database
<a href="#"><u>spacerpt</u></a>	Provides information about space use by FileNet tables in the index or WorkFlo databases
<a href="#"><u>sqlplus</u></a>	Provides commands to view tables in the index or WorkFlo databases
<a href="#"><u>ssn</u></a>	Displays the system serial number
<a href="#"><u>st_msar_convert</u></a>	Converts an optical surface to an MSAR surface in the background
<a href="#"><u>stamp</u></a>	Displays file characteristics (such as release number, developer, and SCR number)
<a href="#"><u>stampro</u></a>	Displays file characteristics (a read-only version of stamp)
<a href="#"><u>stdoccopy</u></a>	Copies information from one medium to another without first importing the medium
<a href="#"><u>stdocimp</u></a>	Starts an import process of documents stored on media
<a href="#"><u>stmigrate</u></a>	Moves documents from magnetic disk cache to storage media
<a href="#"><u>stsurfupdate</u></a>	Starts up a surface update background job that supports the updating of the short descriptors. It updates short descriptors (or user indexes) stored on optical surfaces by writing to a new location with index information taken from the Index Database
<a href="#"><u>sync_write_test</u></a>	Determines if a given storage management system directory supports synchronous writes
<a href="#"><u>sys_log</u></a>	Appends a message to the error log

Tool	Description
<a href="#"><u>tapeser</u></a>	Writes a volume label with serial number to a tape
<a href="#"><u>TAP_tool</u></a>	Provides commands to manage tapes and tape requests
<a href="#"><u>test_raw_partition</u></a>	Tests whether a UNIX platform supports raw partitions and synchronous writes to partitions and files
<a href="#"><u>TLIB_tool</u></a>	Controls Exabyte tape library functions
<a href="#"><u>uncompressdir</u></a>	Restores to their original sizes all files in a directory that were previously compressed with the <a href="#"><u>compressdir</u></a> tool  (Does not run on Image Services for Windows Server or Image Services for the Solaris Operating Environment platforms)
<a href="#"><u>vl</u></a>	Displays the FileNet event log
<a href="#"><u>WALSEC_debug</u></a>	Enables support personnel to trace execution within the Security Services subsystem of the FileNet IS Toolkit API.
<a href="#"><u>whatsup</u></a>	Displays a list of FileNet programs that are currently loaded into the memory of the server
<a href="#"><u>WQMpack</u></a>	Compresses WorkFlo database tables and queues
<a href="#"><u>WQS_move_q</u></a>	Copies WorkFlo Queue Services queues between servers
<a href="#"><u>WQS_tool</u></a>	Provides commands to view information on WorkFlo queues, including table IDs, queue names, and table field names
<a href="#"><u>WRT_clean</u></a>	Deletes outstanding write requests from the transient database or deletes documents from cache
<a href="#"><u>XPR_print</u></a>	Prints server-based files

## Subsection Descriptions

Tool descriptions can include several subsections. Each subsection has a unique purpose, as described below.

### Description

This subsection describes the actions performed by the tool during execution. The emphasis is on internal operation of the program.

### Use

Use describes how and when to use a tool.

### Syntax

Syntax provides the correct arrangement of tool commands, parameters, options, and flag settings.

### Flags and Options

Some tools have associated flags and/or options, which are described here. If a tool contains a number of flag settings, only the most commonly-used flags are described. Some menu-driven tools or those that require subcommands do not have flag settings.

### Commands

Several programs (CSM\_tool and MKF\_tool, for example) operate as entire subsystems and have their own set of commands, which are listed in special command sections.

### **Examples or Sample Output**

Output produced by execution of the tool is shown and described. Use the examples and/or sample output as a guide for interpreting your own output.

### **Checklist**

The checklist alerts you to important precautionary measures to take before you use a program. Read the checklist **before** you use a tool. Checklist items can involve shutting down FileNet software, contacting your service representative, or configuring particular files.

### **Procedure**

The recommended procedure describes the steps you perform to use a tool. The emphasis is on user action.

### **Related Topics**

Use these items as a cross-reference to related tools, manuals, or on-line help information.



# Cross Reference

This chapter provides a cross reference by task to the tools documented in [“Tool Descriptions” on page 69](#).

Use this cross reference when you know the task you need to perform and want to know the tool that can perform the task. For example, if you want to know how to import cache objects, look under the “Cache” heading and find the “Import objects” subheading. The tool associated with this task is CSM\_exim. Then refer to [“CSM\\_exim” on page 190](#) for details.

## Backup Preparation and Analysis

Analyze target tape drive	<a href="#">TAP_tool</a>
Display archive logging status	<a href="#">oraloglist</a>

## Batches

Analyze	<a href="#">bes_check</a> , <a href="#">bes_debug</a>
Delete records	<a href="#">bes_clean</a>
Diagnose inconsistencies	<a href="#">bes_check</a>
Update	<a href="#">bes_setid</a>

---

## Cache

Analyze fast batch objects	<a href="#"><u>DOC_dumpfb</u></a>
Analyze BES cache	<a href="#"><u>bes_check</u></a>
Back up logical cache	<a href="#"><u>CSM_exim</u></a>
Delete objects	<a href="#"><u>nch_flush</u></a> , <a href="#"><u>WRT_clean</u></a>
Display statistics	<a href="#"><u>CSM_tool</u></a>
Dump contents	<a href="#"><u>CSM_tool</u></a> , <a href="#"><u>DOC_dumpfb</u></a>
Export objects	<a href="#"><u>CSM_exim</u></a>
Identify partitions	<a href="#"><u>sgs</u></a>
Import objects	<a href="#"><u>CSM_exim</u></a>
Initialize	<a href="#"><u>CSM_tool</u></a>
Move documents from cache to storage media	<a href="#"><u>stmigrate</u></a>
Restore	<a href="#"><u>CSM_exim</u></a>
Verify pending write requests	<a href="#"><u>checkwrt</u></a> , <a href="#"><u>WRT_clean</u></a>

## Configuration

See online help for System Configuration Tools.

Copy/export/migrate the system configuration (.cdb) file for cloning purposes.	<a href="#"><u>export_cdb</u></a> , <a href="#"><u>import_cdb</u></a> , <a href="#"><u>migrate_cdb_changes</u></a>
Increases the precision and scale of a numeric user index	<a href="#"><u>enlarge_ncol</u></a>
Invoke a graphical user interface (GUI) utility called the FileNet System Configuration Editor	<a href="#"><u>fnddcfg</u></a>
Configure FnScsidd	<a href="#"><u>fnddcfg</u></a>
Start, stop, create, and update the MKF and RDBMS databases	<a href="#"><u>fn_util</u></a>
Sets up file and directory permissions on Windows servers	<a href="#"><u>fn_perm</u></a>
Collect statistics for FileNet datasets and MKF databases	<a href="#"><u>getstatus</u></a>
Collect statistics for FileNet datasets and RDBMS databases	<a href="#"><u>getstamps</u></a>

## Core Files

Analyze	<a href="#"><u>cordebug</u></a> , <a href="#"><u>common</u></a>
Display	<a href="#"><u>cstat</u></a>

---

## Databases

Analyze	<a href="#"><u>GDBdebug</u></a> , <a href="#"><u>ixdb_stat</u></a>
Copy databases	<a href="#"><u>GDB_exim</u></a>
Collect performance and configuration statistics for MKF databases	<a href="#"><u>getstatus</u></a>
Create table views	<a href="#"><u>dclview</u></a>
Increase the precision and/or scale of a numeric user index	<a href="#"><u>export_cdb</u></a>
Initialize	<a href="#"><u>MKF_ddl</u></a> , <a href="#"><u>SEC_init</u></a>
Starts, stops, creates, and updates the MKF and RDBMS databases	<a href="#"><u>fn_util</u></a>
Update	<a href="#"><u>MKF_ddl</u></a> , <a href="#"><u>MKF_tool</u></a>
Verify tables	<a href="#"><u>dbverify</u></a>

See also:

- [“Index Database” on page 49](#)
- [“MKF Databases” on page 53](#)
- [“NCH Database” on page 56](#)
- [“Permanent Database” on page 58](#)
- [“Security Database” on page 62](#)
- [“Transient Database” on page 65](#)
- [“WorkFlo Database” on page 68](#)

## Data Dictionary

Backup	<a href="#"><u>ddexim</u></a>
Export and import	<a href="#"><u>ddexim</u></a>

## Document Committal

Analyze batches	<a href="#"><u>bes_check</u></a> , <a href="#"><u>bes_debug</u></a>
View page cache	<a href="#"><u>CSM_tool</u></a>

## Document Deletion

Delete documents	<a href="#"><u>deldocs</u></a>
Remove references to surfaces	<a href="#"><u>remove_docs_ref</u></a> , <a href="#"><u>remove_surf</u></a>

## Document Services

Count documents in database tables	<a href="#"><u>doccnt</u></a>
Delete documents	<a href="#"><u>deldocs</u></a>
Diagnose problems	<a href="#"><u>dbp</u></a>
Move documents from cache to storage media	<a href="#"><u>stmigrate</u></a>
Verify document format on media	<a href="#"><u>docchk</u></a>

## Document Retrieval

View cache	<a href="#"><u>CSM_tool</u></a>
------------	---------------------------------

## Enterprise Backup/Restore (EBR) tools

Reclaim shared memory and interlocks for a failed EBR backup or restore	<a href="#"><u>EBR_clean</u></a>
Generate EBR dataset definition files, device specification files, and builds EBR backup and restore scripts	<a href="#"><u>EBR_genscript</u></a>
Label and relabel EBR backup tapes in a stand-alone tape drive or tape library and disk files used for backup	<a href="#"><u>EBR_label</u></a>
Resets an Oracle database, which has terminated abnormally, to a working state	<a href="#"><u>EBR_orreset</u></a>
Display EBR tape label on a tape in a stand-alone tape drive or tape library, or on a disk file	<a href="#"><u>EBR_tdir</u></a>
Unlock MKF databases locked for an offline EBR backup that did not complete	<a href="#"><u>EBR_ulmk</u></a>
Control Exabyte tape library functions.	<a href="#"><u>TLIB_tool</u></a>

## Event Log

Append message	<a href="#"><u>sys_log</u></a>
Display	<a href="#"><u>vl</u></a>

---

## File Manipulation

Compress	<a href="#"><u>compressdir</u></a>
Copy files between systems	<a href="#"><u>gcp</u></a>
Display file characteristics	<a href="#"><u>stamp</u></a> , <a href="#"><u>stampro</u></a>
Display file contents	<a href="#"><u>less</u></a>
Locate/list file identification	<a href="#"><u>ident</u></a>
Release maintenance	<a href="#"><u>manifest</u></a>
Uncompress files in a directory	<a href="#"><u>uncompressdir</u></a>

## FileNet Software

Diagnose	<a href="#"><u>ipc_tool</u></a>
Display ILKs/PCRs	<a href="#"><u>FNL_disp</u></a>
Display process status	<a href="#"><u>whatsup</u></a>
Restart	<a href="#"><u>initfnsw restart</u></a>
Sets file permissions	<a href="#"><u>fn_setup</u></a>
Start	<a href="#"><u>initfnsw start</u></a>
Starts, stops, creates, and updates the MKF and RDBMS databases	<a href="#"><u>fn_util</u></a>
Stop	<a href="#"><u>initfnsw stop</u></a>



## Index Database

Analyze	<a href="#"><u>cti</u></a> , <a href="#"><u>ixdb_stat</u></a> , <a href="#"><u>sqlplus</u></a> , <a href="#"><u>WQS_tool</u></a>
Analyze performance	<a href="#"><u>INXdebug</u></a>
Collect performance and configuration statistics for MKF databases	<a href="#"><u>getstatus</u></a>
Delete documents	<a href="#"><u>deldocs</u></a>
Display space use	<a href="#"><u>spacerpt</u></a>
Examine tables	<a href="#"><u>sqlplus</u></a>
Export and import indexes	<a href="#"><u>ddexim</u></a>
Gather statistics	<a href="#"><u>ixdb_stat</u></a> , <a href="#"><u>sqlplus</u></a> , <a href="#"><u>WQS_tool</u></a>
Increase the precision and/or scale of a numeric user index	<a href="#"><u>export_cdb</u></a>
Modify tables	<a href="#"><u>sqlplus</u></a> , <a href="#"><u>WQS_tool</u></a>
Modify WorkFlo queues and workspaces	<a href="#"><u>WQS_tool</u></a>
Rebuild	<a href="#"><u>odrecover</u></a>
Updates short descriptors (or user indexes) stored on optical surfaces by writing to a new location with index information taken from the Index Database	<a href="#"><u>stsurfupdate</u></a>

## Integral Single Document Storage (SDS)

For further information on these tools, see the *Integral SDS Procedures and Guidelines for Images Services*. To download this document from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

Converts documents from NLS-SDS to integral SDS.	<b>sds_convert_nls</b>
Copies documents between two SDS units.	<b>sds_copy</b>
Places a document on indefinite hold or releases a hold.	<b>sds_hold</b>
Imports SDS documents from another Image Services system.	<b>sds_import</b>
Migrates documents from MSAR optical media to SDS.	<b>sds_migr</b>
Updates the retention for EBR documents.	<b>sds_update_retention</b>

## Lightweight Directory Access Protocol (LDAP)

Supports ISRA/LDAP customers with anonymous logins.	<a href="#"><u>cranuser</u></a>
Gives customers the ability to use their LDAP tools to generate the LDIF files for use with FileNet software.	<a href="#"><u>fn_ldif_xfer</u></a>
Export the user names and group names from an LDAP directory service domain to an intermediate XML formatted file in an effort to support group/user mappings between the LDAP directory service and IS groups (Unified Logon). Used in conjunction with the LDAP_import tool.	<a href="#"><u>ldap_exp</u></a>
Import the LDAP security information into the IS Security Service in an effort to support group/user mappings between LDAP security service domain groups and IS groups. This tool must be run on an Image Services server (Windows Server or UNIX). Used in conjunction with the ldap_exp tool.	<a href="#"><u>LDAP_import</u></a>
Encrypt a password that will be used by the LDAP_import tool during the import process, so the user name and password can be provided through the command line.	<a href="#"><u>LDAP_password</u></a>

## Logging

Create a log file to receive performance log records	<a href="#"><u>log_create</u></a>
Displays quick logging data from the log file	<a href="#"><u>QLG_dump</u></a>
Write in-memory quick logging records to a log file	<a href="#"><u>QLG_flush</u></a>
Initiate quick logging	<a href="#"><u>QLG_start</u></a>

## Memory

Analyze corruption	<a href="#"><u>ipc_tool</u></a>
Display FileNet programs loaded into server memory	<a href="#"><u>whatsup</u></a>

## Messages

Append	<a href="#"><u>sys_log</u></a>
Interpret	<a href="#"><u>fn_msg</u></a>

## Microsoft SQL Server Databases

**WIN**

This database program is available for Windows servers only.

Analyze	<a href="#"><u>cti</u></a>
Display database statistics	<a href="#"><u>spacerpt</u></a>

See also [“Index Database” on page 49.](#)

## MKF Databases

Analyze content	<a href="#"><u>MKF_tool</u></a>
Analyze corruption	<a href="#"><u>MKF_dump</u></a>
Analyze hangs	<a href="#"><u>GDBdebug</u></a> , <a href="#"><u>MKF_debug</u></a>
Analyze performance	<a href="#"><u>GDBdebug</u></a> , <a href="#"><u>MKF_stats</u></a>
Byte ordering	<a href="#"><u>MKF_order</u></a>
Collect performance and configuration statistics for MKF databases	<a href="#"><u>getstatus</u></a>
Copy data	<a href="#"><u>GDB_exim</u></a> , <a href="#"><u>MKF_export</u></a>
Resolve corruption	<a href="#"><u>MKF_fixup</u></a> , <a href="#"><u>MKF_verify</u></a>
Export	<a href="#"><u>MKF_export</u></a>
Import	<a href="#"><u>MKF_import</u></a>
Initialize	<a href="#"><u>MKF_ddl</u></a>
Modify	<a href="#"><u>MKF_ddl</u></a> , <a href="#"><u>MKF_rename</u></a> , <a href="#"><u>MKF_tool</u></a>
Clear recovery logs	<a href="#"><u>MKF_zeroaij</u></a> , <a href="#"><u>MKF_zerobij</u></a>

Shut down	<a href="#"><u>MKF_shutdown</u></a>
Start up	<a href="#"><u>MKF_startup</u></a>
Statistics	<a href="#"><u>MKF_stats</u></a>
Verify integrity	<a href="#"><u>MKF_verify</u></a>
Write to databases	<a href="#"><u>MKF_tool</u></a>

See also:

- [“NCH Database” on page 56](#)
- [“Permanent Database” on page 58](#)
- [“Security Database” on page 62](#)
- [“Transient Database” on page 65](#)

## MSAR Tools

A DOC_tool command that ejects an MSAR surface based on a surface ID as input.	<a href="#"><u>EJmsar</u></a>
A DOC_tool command that inserts or incorporates an MSAR surface into an MSAR library based on the MSAR surface file input.	<a href="#"><u>INSmsar</u></a>
Determines relative speed of I/O for a specific directory	<a href="#"><u>msar_io_test</u></a>
Converts an optical surface to an MSAR surface in the background	<a href="#"><u>st_msar_convert</u></a>
A DOC_tool command that validates the checksum values in an MSAR label	<a href="#"><u>READMsarlabel</u></a>

## National Language Translation

Translate character set	<a href="#"><u>NLT_build_maps</u></a>
-------------------------	---------------------------------------

## NCH

Analyze hangs	<a href="#"><u>flat, nch_check, nch_tool</u></a>
Gather and interpret performance statistics	<a href="#"><u>nch_tool</u></a>
Initialize	<a href="#"><u>MKF_ddl</u></a>
List system resources	<a href="#"><u>nch_tool, remove_docs_ref</u></a>
Modify system resources	<a href="#"><u>MKF_ddl</u></a>
Update	<a href="#"><u>MKF_ddl</u></a>
Verify NCH_daemon and NCH service	<a href="#"><u>nch_check</u></a>

## NCH Database

Analyze hangs	<a href="#"><u>cordebug</u></a> , <a href="#"><u>ipc_tool</u></a> , <a href="#"><u>MKF_debug</u></a> , <a href="#"><u>MKF_tool</u></a> , <a href="#"><u>nch_check</u></a> , <a href="#"><u>nch_tool</u></a>
Delete in-memory version	<a href="#"><u>nch_flush</u></a>
Initialize	<a href="#"><u>MKF_ddl</u></a>
List system resources	<a href="#"><u>nch_tool</u></a>
List performance statistics	<a href="#"><u>MKF_stats</u></a>
Scavenge data	<a href="#"><u>MKF_fixup</u></a>
Shut down	<a href="#"><u>MKF_shutdown</u></a>
Start up	<a href="#"><u>MKF_startup</u></a>
Update	<a href="#"><u>MKF_ddl</u></a>
Verify integrity	<a href="#"><u>MKF_verify</u></a>

## Oracle Databases

Collect performance and configuration statistics for RDBMS databases	<a href="#"><u>getstamps</u></a>
Display archive log mode status	<a href="#"><u>oraloglist</u></a>
Display table spaces and rollback segments	<a href="#"><u>spacerpt</u></a>



## Performance

Collect performance data	<a href="#"><u>eladisp</u></a> , <a href="#"><u>FileNet System Manager</u></a> , <a href="#"><u>getstamps</u></a> , <a href="#"><u>getreports</u></a> , <a href="#"><u>getstatus</u></a> , <a href="#"><u>ipc_tool</u></a> , <a href="#"><u>perf_report</u></a> , <a href="#"><u>perf_mon</u></a>
Create log files	<a href="#"><u>log_create</u></a>
Display free cache	<a href="#"><u>CSM_tool</u></a>
Display ILKs/PCRs	<a href="#"><u>FNL_disp</u></a>
Display performance data	<a href="#"><u>getreports</u></a> , <a href="#"><u>log_dir</u></a> , <a href="#"><u>log_extract</u></a> , <a href="#"><u>perf_report</u></a>
Examine MKF databases	<a href="#"><u>MKF_stats</u></a>
Generate reports	<a href="#"><u>getreports</u></a> , <a href="#"><u>perf_report</u></a>
Determines if a platform supports raw partitions and is performing synchronous writes	<a href="#"><u>test_raw_partition</u></a>

## Permanent Database

Analyze hangs	<a href="#"><u>cordebug</u></a> , <a href="#"><u>ipc_tool</u></a> , <a href="#"><u>MKF_debug</u></a>
Clear recovery logs	<a href="#"><u>MKF_zeroaij</u></a> , <a href="#"><u>MKF_zerobij</u></a>
Collect performance and configuration statistics for MKF databases	<a href="#"><u>getstatus</u></a>
Count records	<a href="#"><u>MKF_tool</u></a>
Delete documents	<a href="#"><u>deldocs</u></a>
Delete records	<a href="#"><u>MKF_tool</u></a>
Delete tables	<a href="#"><u>MKF_tool</u></a>
Describe tables	<a href="#"><u>MKF_tool</u></a>
Describe tables	<a href="#"><u>MKF_tool</u></a>
Diagnose inconsistencies	<a href="#"><u>MKF_dump</u></a>
Display records	<a href="#"><u>MKF_tool</u></a>
Export tables	<a href="#"><u>MKF_export</u></a>
Gather and analyze performance statistics	<a href="#"><u>MKF_stats</u></a>
Import tables	<a href="#"><u>MKF_import</u></a>
Initialize	<a href="#"><u>MKF_ddl</u></a>
Join tables	<a href="#"><u>MKF_tool</u></a>
Modify the base data file	<a href="#"><u>MKF_rename</u></a>
Rebuild	<a href="#"><u>odrecover</u></a>
Update scalar numbers table	<a href="#"><u>SNT_update</u></a>
Scavenge data	<a href="#"><u>MKF_fixup</u></a>
Shut down	<a href="#"><u>MKF_shutdown</u></a>

Start up	<a href="#"><u>MKF_startup</u></a>
Update	<a href="#"><u>MKF_ddl</u></a>
Update scalar numbers table	<a href="#"><u>SNT_update</u></a>
Verify integrity	<a href="#"><u>MKF_verify</u></a>

## Printing

Disable printer	<a href="#"><u>PRI_tool</u></a>
Gather statistics	<a href="#"><u>PRI_tool</u></a>
Print server-based file	<a href="#"><u>XPR_print</u></a>
Re-enable printer	<a href="#"><u>PRI_tool</u></a>
Remove print request	<a href="#"><u>PRI_tool</u></a>
Restart print services	<a href="#"><u>PRI_tool</u></a>
View print cache	<a href="#"><u>CSM_tool</u></a>

## Quick Logging

Initiate	<a href="#"><u>QLG_start</u></a>
Write in-memory records to log file	<a href="#"><u>QLG_flush</u></a>
Display log file contents	<a href="#"><u>QLG_dump</u></a>

## Remote Communications

Analyze connections	<a href="#"><u>flat</u></a> , <a href="#"><u>nch_check</u></a> , <a href="#"><u>nch_tool</u></a>
Dial out	<a href="#"><u>dialout</u></a>
Dial out/data transfer	<a href="#"><u>kermit</u></a>
Display available hosts	<a href="#"><u>gaddr</u></a>
Display remote system directories	<a href="#"><u>gls</u></a>
Run commands on remote systems	<a href="#"><u>gsh</u></a>

## Request Handlers

Configure and debug	<a href="#"><u>ppm_log</u></a> , <a href="#"><u>PPMOI</u></a> , <a href="#"><u>MKF_ddl</u></a>
Display request information	<a href="#"><u>cordebug</u></a> , <a href="#"><u>cormon</u></a>

## Storage Media

Analyze	<a href="#"><u>oddump</u></a>
Display summary error log information	<a href="#"><u>osschk</u></a>
Removes references to surfaces	<a href="#"><u>remove_docs_ref</u></a> , <a href="#"><u>remove_surf</u></a>
Delete documents	<a href="#"><u>deldocs</u></a>
Convert optical to MSAR in background	<a href="#"><u>st_msar_convert</u></a>
Copy media	<a href="#"><u>stdoccopy</u></a>
Import	<a href="#"><u>odrecover</u></a> , <a href="#"><u>stdoccopy</u></a>
Balance load	<a href="#"><u>move_disk</u></a>
Modify	<a href="#"><u>oddump</u></a>
Move documents from cache to storage media	<a href="#"><u>stmigrate</u></a>
Reassign to different Storage Library server	<a href="#"><u>del_osvr</u></a> , <a href="#"><u>move_disk</u></a>
Verify document format	<a href="#"><u>docchk</u></a>
Write	<a href="#"><u>oddump</u></a>
Storage management synchronous writes	<a href="#"><u>sync_write_test</u></a>
List SCSI optical disk drives	<a href="#"><u>fndev</u></a>

## Security

Analyze	<a href="#"><u>SEC_tool</u></a>
Debug security services during runtime	<a href="#"><u>SEC_debug</u></a> , <a href="#"><u>WALSEC_debug</u></a>
Logon to security service	<a href="#"><u>fnlogon</u></a>
Set up file and directory permissions on Windows servers	<a href="#"><u>fn_perm</u></a>
Convert all standard SQL scripts and system scripts into customized scripts	<a href="#"><u>fn_pso_driver</u></a>
Update the master Programmable Object Data File (PODF)	<a href="#"><u>fn_pso_pdf_admin</u></a>
Change the user name of the standard FileNet software user	<a href="#"><u>fn_pso_switch</u></a>
Modify security	<a href="#"><u>SEC_init</u></a> , <a href="#"><u>SEC_tool</u></a>

## Security Database

Analyze	<a href="#"><u>MKF_ddl</u></a> , <a href="#"><u>SEC_tool</u></a>
Initialize	<a href="#"><u>SEC_init</u></a>
Logon to security service	<a href="#"><u>fnlogon</u></a>
Modify	<a href="#"><u>SEC_tool</u></a>
Reset	<a href="#"><u>SEC_init</u></a>

---

## Space Use

Analyze	<a href="#"><u>spacerpt</u></a>
---------	---------------------------------

## Storage Library Server

Add Storage Library server	<a href="#"><u>add_osvr</u></a>
Delete Storage Library server	<a href="#"><u>del_osvr</u></a>
Convert optical to MSAR in background	<a href="#"><u>st_msar_convert</u></a>
Copy from media to media	<a href="#"><u>stdoccpy</u></a>
Display read/write requests	<a href="#"><u>DOC_tool</u></a>
Gather drive information	<a href="#"><u>DOC_tool</u></a>
Gather surface statistics	<a href="#"><u>DOC_tool</u></a> , <a href="#"><u>oddump</u></a>
Import documents	<a href="#"><u>odrecover</u></a> , <a href="#"><u>stdocimp</u></a>
Obtain slot information	<a href="#"><u>DOC_tool</u></a>

## System Information

Take snapshot of an Image Services system in crisis	<a href="#"><u>911</u></a>
Collect stamps for all subsystems on the Image Services server on which you run the program. You can also use this tool to compare the stamps with the current database of SCRs that identify the available system fixes for a specified platform and Image Services software release.	<a href="#"><u>getstamps</u></a>
List the system configuration settings on the Image Services server	<a href="#"><u>sgs</u></a>
Display server serial number	<a href="#"><u>ssn</u></a>

## Tape and Tape Drives

Analyze tape drives	<a href="#"><u>TAP_tool</u></a>
Write volume serial number	<a href="#"><u>tapeser</u></a>



## Transient Database

Analyze hangs	<a href="#"><u>cordebug</u></a> , <a href="#"><u>ipc_tool</u></a> , <a href="#"><u>MKF_debug</u></a>
Clear recovery logs	<a href="#"><u>MKF_zeroaij</u></a> , <a href="#"><u>MKF_zerobij</u></a>
Collect performance and configuration statistics for MKF databases	<a href="#"><u>getstatus</u></a>
Count records	<a href="#"><u>MKF_tool</u></a>
Delete records	<a href="#"><u>MKF_tool</u></a>
Delete write requests	<a href="#"><u>WRT_clean</u></a>
Describe tables	<a href="#"><u>WQS_tool</u></a>
Diagnose inconsistencies	<a href="#"><u>MKF_dump</u></a>
Display records	<a href="#"><u>MKF_tool</u></a>
Export tables	<a href="#"><u>MKF_export</u></a>
Gather and analyze performance statistics	<a href="#"><u>MKF_stats</u></a>
Import tables	<a href="#"><u>MKF_import</u></a>
Initialize	<a href="#"><u>MKF_ddl</u></a>
Modify the base data file	<a href="#"><u>MKF_rename</u></a>
Scavenge data	<a href="#"><u>MKF_fixup</u></a>
Shut down	<a href="#"><u>MKF_shutdown</u></a>
Start up	<a href="#"><u>MKF_startup</u></a>
Update	<a href="#"><u>bes_setid</u></a> , <a href="#"><u>MKF_ddl</u></a>
Verify integrity	<a href="#"><u>MKF_verify</u></a>

## Troubleshooting Tools

Take a snapshot of an Image Services system in crisis.	<a href="#"><u>911</u></a>
Collect stamps for all subsystems on the Image Services server on which you run the program. You can also use this tool to compare the stamps with the current database of SCRs that identify the available system fixes for a specified platform and Image Services software release.	<a href="#"><u>getstamps</u></a>
Display error log summary for optical storage libraries	<a href="#"><u>osschk</u></a>

## Unified Logon Tools

<p>Export the user names and group names from a Windows Server domain to an intermediate XML formatted file in an effort to support group/user mappings between Windows Server domain groups and IS groups (Unified Logon). Used in conjunction with the RSEC_imp or SEC_imp tool.</p>	<p><b><u>ntdm_exp</u></b></p>
<p>Import the Windows Server domain security information into the IS Security Service in an effort to support group/user mappings between Windows Server domain groups and IS groups (Unified Logon). This tool must be run on a Windows Server machine with no Image Services software present, but that has either Web Service or IDM Desktop installed. Used in conjunction with the ntdm_exp tool.</p>	<p><b><u>RSEC_imp</u></b></p>
<p>Import the Windows Server domain security information into the IS Security Service in an effort to support group/user mappings between Windows Server domain groups and IS groups (Unified Logon). This tool must be run on an Image Services server (Windows Server or UNIX). Used in conjunction with the ntdm_exp tool.</p>	<p><b><u>SEC_imp</u></b></p>

## WorkFlo Database

Analyze	<a href="#"><u>sqlplus</u></a> , <a href="#"><u>WQS_tool</u></a>
Collect performance and configuration statistics for RDBMS databases	<a href="#"><u>get_rdbms</u></a>
Compress	<a href="#"><u>WQMpack</u></a>
Delete tables	<a href="#"><u>sqlplus</u></a>
Examine queues	<a href="#"><u>WQS_tool</u></a>
Examine tables	<a href="#"><u>sqlplus</u></a> , <a href="#"><u>WQS_tool</u></a>
Modify	<a href="#"><u>sqlplus</u></a>
Obtain space use information	<a href="#"><u>spacerpt</u></a>
Obtain statistics	<a href="#"><u>WQS_tool</u></a>

# Tool Descriptions

## 911

### Description



UNIX



WIN

The 911 tool is available on both UNIX-based and Windows Server-based Image Services servers.

911 is a shell script that takes a quick snapshot of an Image Services system in crisis. It quickly takes a cross-section of some key system metrics.

A good snapshot from 911 can isolate an actual system problem most of the time. For example, it could diagnose the following types of system errors:

- Server has run out of service request handlers (server stubs).
- System operation is slow.
- System is congested.

After gathering vital information, 911 writes data from various IS and system commands to files (reports). Using an editor, such as vi, you can view each generated report to help you resolve any problems detected.

The 911 script collects log and configuration files and then tars and compresses or zips (depending on your platform) them into one file (911.YYYYMMDD.##.Z on UNIX/911.YYYYMMDD.##.zip on Windows Server) that can be easily downloaded.

This compressed file includes the following:

- All of the files shown in **“Sample Reports” on page 73.**
- Current Configuration Database (CDB) plus any CDB files modified in the last 30 days
- ServerConfig
- ServerConfig.custom (if it exists)
- \*.ddl
- as\_conf.i.g
- as\_conf.s
- init.ora (FileNet-controlled only)
- init.rs (FileNet-controlled only)
- rc.net (AIX only)
- Near Line Storage (NLS) log files (current day only)
- IS log files (below shows UNIX - use \fnsw\_loc\logs\...for Windows)
  - /fnsw/local/logs/elogs (current day's elog file)
  - /fnsw/local/logs/TM\_daemon (TM\_daemon.log, TM\_daemon.old)
  - /fnsw/local/logs/ims\_logs (last 7 days of logs)
  - /fnsw/local/logs/.logs (current days log)

- All of the .txt files listed in the perf\_report -a output located at /fnsw/local/logs/perf/. to view this output, see [“perf\\_report\\_rful.YYYYMMDD.##.wri” on page 103](#).

You can also write pre911 and post911 scripts (pre911.cmd/post911.cmd for Windows Server) that reside in /fnsw/support to add customized output to 911. The pre911 script, if it exists, is run on the first step of 911. The post911 script, if it exists, is run on the last. Output from these scripts is stored in pre911.YYYYMMDD.##.wri and post911.YYYYMMDD.##.wri.

## Use

Use 911 when your system is having an undefined problem. You might want to use this tool when the standard and Image Services error logs fail to provide sufficient diagnostic information.

## Syntax

911[-p] [nowinmsd]

**-p** print option (UNIX only)

See [“pri\\_diags.YYYYMMDD.##.wri” on page 106](#) for a sample of this output.

**nowinmsd** Skips the winmsd command (Windows only). Winmsd could take several minutes to gather Windows operating system information on some systems.

## Sample Output

After you enter the 911 command, it collects vital information, displaying its progress on the screen, as shown in the following example.

## GATHERING VITAL INFORMATION:

NOTE: For a comprehensive snapshot, be sure to run 911 on all servers

```
Starting 911 script on milo at Wed Jul 27 14:42:43 PDT 2005
Finding core files at Wed Jul 27 14:42:44 PDT 2005
Getting stack traces from core files at Wed Jul 27 14:42:44 PDT 2005
Gathering virtual memory statistics at Wed Jul 27 14:42:47 PDT 2005
Gathering network statistics at Wed Jul 27 14:43:28 PDT 2005
Checking disk usage at Wed Jul 27 14:43:29 PDT 2005
Gathering ps information at Wed Jul 27 14:43:29 PDT 2005
Gathering OS specific data at Wed Jul 27 14:43:29 PDT 2005
Gathering RPC data at Wed Jul 27 14:43:30 PDT 2005
Gathering Process Management data at Wed Jul 27 14:43:31 PDT 2005
Determining sys config at Wed Jul 27 14:43:32 PDT 2005
...
Taking second snapshot of key statistics at Wed Jul 27 14:43:53 PDT 2005
Gathering RPC data at Wed Jul 27 14:44:53 PDT 2005
Running fn_procs at Wed Jul 27 14:45:13 PDT 2005
Running getstamps.sh at Wed Jul 27 14:45:13 PDT 2005
WARNING: Erasing previous /fnsw/local/tmp/stamps.milo.lst...
Getting basic system info...
Fetching IMS stamp info, please be patient...
Fetching WAL/Unix stamp info, please be patient...
Done capturing FileNET stamp information:
See list file /fnsw/local/tmp/stamps.milo.lst.
Ending 911 script at Wed Jul 27 14:45:36 PDT 2005
```

```
=====
The following file has been created and is available to download:
```

```
/fnsw/local/logs/911/911.20050727.01.Z
```

```
PLEASE CALL FILENET TECHNICAL SUPPORT IMMEDIATELY:
1-800-624-4877
```

```
=====
```



## Sample Reports

The following sections show samples of reports generated on an Image Services server when you run the 911 tool. They also describe the type of information 911 collects for each report.

---

**Tip** The vi editor displays all files in a directory through its batch mode. To start displaying these files, enter the command:

```
vi *.yyyyddmm.*
```

Use the **<Esc>:n!** command to quickly go from one report file to the next.

---

### **911log.YYYYMMDD.##.wri**

This report tells us what is happening as the 911 tool is running.

The following illustration shows a sample 911log report on a UNIX system.

```
Starting 911 script on becks at Fri Jul 22 10:45:26 PDT 2005
Running pre911 script at Fri Jul 22 10:45:26 PDT 2005
Finding core files at Fri Jul 22 10:45:26 PDT 2005
Getting stack traces from core files at Fri Jul 22 10:45:26 PDT 2005
Gathering virtual memory statistics at Fri Jul 22 10:45:33 PDT 2005
Gathering network statistics at Fri Jul 22 10:46:14 PDT 2005
Checking disk usage at Fri Jul 22 10:46:14 PDT 2005
Gathering ps information at Fri Jul 22 10:46:14 PDT 2005
Gathering OS specific data at Fri Jul 22 10:46:15 PDT 2005
Gathering RPC data at Fri Jul 22 10:46:15 PDT 2005
Gathering Process Management data at Fri Jul 22 10:46:16 PDT 2005
Determining sys config at Fri Jul 22 10:46:16 PDT 2005
Gathering MKF statistics at Fri Jul 22 10:46:45 PDT 2005
Gathering CSM statistics at Fri Jul 22 10:46:46 PDT 2005
Gathering WQS statistics at Fri Jul 22 10:46:46 PDT 2005
Checking /fnsw/proc for orphaned PIDs at Fri Jul 22 10:46:47 PDT 2005
Gathering FNL_disp statistics at Fri Jul 22 10:46:47 PDT 2005
Gathering dbp statistics at Fri Jul 22 10:46:47 PDT 2005
Gathering SEC statistics at Fri Jul 22 10:46:47 PDT 2005
Running performance reports at Fri Jul 22 10:46:48 PDT 2005
Dumping root and fnsw environment variables at Fri Jul 22 10:46:57 PDT 2005
List conf_db directory contents at Fri Jul 22 10:47:00 PDT 2005
List config and trigger files at Fri Jul 22 10:47:00 PDT 2005
Taking second snapshot of key statistics at Fri Jul 22 10:47:00 PDT 2005
Gathering RPC data at Fri Jul 22 10:48:00 PDT 2005
Running fn_procs at Fri Jul 22 10:48:21 PDT 2005
Running getstamps.sh at Fri Jul 22 10:48:21 PDT 2005
Running post911 script at Fri Jul 22 10:48:46 PDT 2005
Ending 911 script at Fri Jul 22 10:48:46 PDT 2005
```

The following illustration shows a sample 911log report on a Windows Server system.

```
Starting 911 script at 11:07:15.46
Collecting system information at 11:07:15.48
Running fn_util whichfn at 11:07:18.79
Running nch_tool at 11:07:18.90
Running ipc_tool at 11:07:19.01
Running cormon at 11:07:19.31
Running mkf_debug at 11:07:19.37
Running mkf_stats at 11:07:19.46
Running fnl_disp at 11:07:20.57
Running dbp at 11:07:20.84
Running csm_tool at 11:07:20.93
Running wqs_tool at 11:07:21.14
Running perf_report -rful at 11:07:21.32
Running perf_report -a at 11:07:21.73
Running ppmoi at 11:07:22.42
Running mkf_tool at 11:07:22.51
Running pri_tool at 11:07:22.81
Running sec_tool at 11:07:23.03
List conf_db directory contents at 11:07:23.29
Reporting configuration and trigger files at 11:07:23.31
Calling fn_procs at 11:07:23.76
Getting registry entry at 11:07:23.93
Calling getstamps32 at 11:07:24.43
...
Zip up todays elog file at 11:08:23.54
Zip up TM_daemon log files at 11:08:23.65
Zip up todays .log file at 11:08:23.78
Zip up CDB files at 11:08:23.87
Zip up ims_log files at 11:08:24.06
Zip up perf_report -a files at 11:08:24.21
Zip up ddl files at 11:08:25.09
Zip up serverConfig file at 11:08:25.25
Zip up as_conf.g file at 11:08:25.35
Zip up as_conf.s file at 11:08:25.45
Zip up getstamps32.exe output at 11:08:25.56
Zip up data dumps at 11:08:25.67
```

**UNIX****allcorefiles.YYYYMMDD.##.wri**

This report tells us whether any programs system wide "dumped core." If so, that program could have precipitated the entire crisis. If a core file is found under /fnsw or under the user fnsw home directory, 911 uses cstat and a debugger to analyze the core dumps.

The following illustration shows a sample allcorefiles report.

```
Core files found on this system on Thu Jul 21 16:45:46 PDT 2005:
-rw-rw-r--  1 root    bin          1933 Dec  3  1999 /usr/openwin/share/include/
images/core_eye.icon
-rw-rw-r--  1 root    bin           589 Dec  3  1999 /usr/openwin/share/include/
images/coredoc.icon
-r--r--r--  1 bin     bin          181 Dec  9  1999 /usr/share/man/sman1f/
coreceive.lf
-r--r--r--  1 bin     bin       15514 Dec  9  1999 /usr/share/man/sman1m/
coreadm.lm
-r--r--r--  1 bin     bin       16220 Dec  9  1999 /usr/share/man/sman4/core.4
-rw-r--r--  1 root    bin          3478 Jan  5  2000 /usr/share/lib/sgml/locale/
C/entities/core.gml
-r-xr-xr-x  1 root    bin       12052 Jan  5  2000 /usr/bin/coreadm
total 4
drwxr-xr-x  2 oracle  dba          512 Nov  3  2003 lmx
drwxr-xr-x  2 oracle  dba          512 Nov  3  2003 lvf
```

**confdb.YYYYMMDD.##.wri**

The following illustration shows a directory listing of /fnsw/local/sd/conf\_db on a UNIX system.

```

COMMAND: ls -ltr /fnsw/local/sd/conf_db / DATE: Thu Jul 21 15:24:14 PDT 2005
total 2758
-rw-rw----  1 fnsw  fnusr    20766 Nov  5  2003 IMS_1.cdb
-rw-rw----  1 fnsw  fnusr    20766 Nov  5  2003 IMS_2.cdb
-rw-rw----  1 fnsw  fnusr    20772 Nov  6  2003 IMS_3.cdb
-rw-rw----  1 fnsw  fnusr    24508 Nov  6  2003 IMS_4.cdb
-rw-rw----  1 fnsw  fnusr    25309 Mar  2  2004 IMS_5.cdb
-rw-rw----  1 fnsw  fnusr    25309 Mar  2  2004 IMS_6.cdb
-rw-rw----  1 fnsw  fnusr    25309 Mar  2  2004 IMS_7.cdb
-rw-rw----  1 fnsw  fnusr    25557 Apr 16  2004 IMS_8.cdb

```

The following illustration shows a directory listing of \fnsw\_loc\sdc\conf\_db on a Windows Server system.

```

COMMAND: dir /od fnsw_loc\sdc\conf_db TIME: 11:07:23.29
Volume in drive C is Dell Server
Volume Serial Number is 1846-3F4B

Directory of C:\fnsw_loc\sdc\conf_db

09/21/2004  05:52 PM                21,281 IMS_1.cdb
09/21/2004  05:53 PM                21,671 IMS_2.cdb
09/22/2004  10:31 AM                22,503 IMS_3.cdb
09/22/2004  01:11 PM                22,767 IMS_4.cdb
04/26/2005  03:41 PM                21,935 IMS_5.cdb
04/28/2005  11:34 AM                21,936 IMS_6.cdb
07/14/2005  01:43 PM                21,936 IMS_7.cdb
07/15/2005  01:58 PM                22,131 IMS_8.cdb
07/21/2005  05:25 PM                <DIR>      .
07/21/2005  05:25 PM                <DIR>      ..
                8 File(s)                176,160 bytes
                2 Dir(s)    5,290,573,824 bytes free

```

**cormon.YYYYMMDD.##.wri**

Cormon reports when any server processes have halted operation and, if so, which client PC started that process. It can also detect when a subsystem has run out of service request handlers (server stubs). cormon is run twice to see if status has changed and both runs output to the same file.

**Note** On Windows Server systems, cormon is run twice to see if status has changed. As a consequence there are two different file names: cormon1.YYYYMMDD.##.wri and cormon2.YYYYMMDD.##.wri.

Use this tool in conjunction with ipc\_tool, ps, and PPMOI.

911 takes two cormon samples. Compare the report at the top with the one at the bottom; if they're different, things are probably just very, very slow — not completely stopped.

**Tip** Use the cormon file to find the longest Remote Procedure Call (RPC) time.

The following example shows the longest RPC as **2208** on a UNIX system. The report looks very similar on a Windows Server system

```
COMMAND: cormon -p / DATE: Thu Jul 21 13:23:20 PDT 2005
CORH_state  Srvr    PID    ChldPID  Prog      Time    LatestUser@Address
-----
RCVMSG     X      1555    7765     SQIs      2208    172.16.10.138 [1083]
RCVMSG     594    12147          SECS0     172.16.10.16 [1340]
RCVMSG     X      1648    12092     SQIs0     172.16.20.228 [2519]
RCVMSG     X      1413    9856     WQS48     172.16.20.16 [2989]
RCVMSG     X      1507    27057     WQSs      1022    172.18.10.28 [2576]
RCVMSG     686    12155          INXs0     172.17.10.8 [2758]
```

**csm\_tool.YYYYMMDD.##.wri**

The following illustration shows a sample CSM\_tool st (**in bold**) and st long (statistics) report on a UNIX system. The report looks very similar on a Windows Server system.

```

COMMAND: CSM_tool, st, st long / DATE: Thu Jul 21 15:24:06 PDT 2005

Type '?' for help

Cache Id  Name                                % locked % full % free
-----  -
      1  page_cache1:becks:FileNet                0       1    99
      3  bes_cache1:becks:FileNet                  0       0   100
      4  sys_print_cache1:becks:FileNet            0       0   100
      5  app_print_cache1:becks:FileNet            0       0   100
      *  Physical space summary                  0       1    99

Statistics for cache #1, name = 'page_cache1:becks:FileNet'
min_cache_sectors  4718592  locked_sectors    0  inuse_sectors    42691
max_cache_sectors  4718592  locked_objects    0  inuse_objects    13
free_sectors       4675901  ageable           T    refcnts          F
self_cleaning      F

-----
Statistics for cache #3, name = 'bes_cache1:becks:FileNet'
min_cache_sectors  314572  locked_sectors    0  inuse_sectors    0
max_cache_sectors  1447034  locked_objects    0  inuse_objects    0
free_sectors       1447034  ageable           F    refcnts          F
self_cleaning      F

-----
Statistics for cache #4, name = 'sys_print_cache1:becks:FileNet'
min_cache_sectors  62914   locked_sectors    0  inuse_sectors    0
max_cache_sectors  1195376  locked_objects    0  inuse_objects    0
free_sectors       1195376  ageable           F    refcnts          F
self_cleaning      F

-----
Statistics for cache #5, name = 'app_print_cache1:becks:FileNet'
min_cache_sectors  62914   locked_sectors    0  inuse_sectors    0
max_cache_sectors  1195376  locked_objects    0  inuse_objects    0
free_sectors       1195376  ageable           F    refcnts          F
self_cleaning      F

-----
...

```

**UNIX****cstat.YYYYMMDD.##.wri**

The following illustration shows a sample cstat report from core files found under /fnsw and fnsw user home directories.

```
SPARC machine type
Big Endian Data Encode format
===> SVR4 Version <===
core file..... /fnsw/local/logs/TM_daemon/core.nvaytl
command..... SDS_CSAR_reader 1 2
real user..... fnsw
start time..... Thu Jul 21 13:07:05 2005
model..... SUNW,Ultra-250
process status flags.. (0x0) current signal..... (-198)Unknown signal -198
pid = 9133
ppid = 5787
process image size: 4131 pages
resident set size: 724 pages
```

**dbp.YYYYMMDD.##.wri**

The following illustration shows a sample dbp report on a UNIX system. The report looks very similar on a Windows Server system.

```
COMMAND: dbp -s / DATE: Thu Jul 21 15:24:07 PDT 2005
Directory buffers:
  Avg_wait_msec:      0,  Max_wait_msec:      0
  Num_avl_bufs:      4,  Min_avl_bufs:      4
Page buffers:
  Avg_wait_msec:      0,  Max_wait_msec:      0
  Num_avl_bufs:     16,  Min_avl_bufs:     15
Descriptor buffers:
  Avg_wait_msec:      0,  Max_wait_msec:      0,  Cache hits:      0
  Num_avl_bufs:     16,  Min_avl_bufs:     15,  Cache allocs:    0
```



## UNIX

**disk.YYYYMMDD.##.wri**

To review this report:

- Look for any file systems that are nearly 100% full, especially /tmp or /var.
- Verify that there's at least 2 MB available on each file system.
- Pay special attention to the **avail** column, not just **%used**. Remember: 85% free of nothing is still nothing!

The following illustration shows a sample disk report.

```
COMMAND: df -k / DATE: Thu Jul 21 15:23:35 PDT 2005
Filesystem          kbytes   used   avail capacity  Mounted on
/dev/vx/dsk/rootvol 30928248 11657667 18961299   39%      /
/proc                0         0         0         0%      /proc
fd                   0         0         0         0%      /dev/fd
mnttab               0         0         0         0%      /etc/mnttab
swap                 2026624   16 2026608   1%      /var/run
swap                 2027000   392 2026608   1%      /tmp
/dev/vx/dsk/fndg/fnsw 1286378 1021041 136700    89%      /fnsw
/dev/vx/dsk/home     1984564   19 1925009   1%      /export/home
/dev/vx/dsk/fndg/local 1055979 415719 582662    42%      /fnsw/local
/dev/vx/dsk/fndg/ora920 2879972 2362320 229655    92%      /usr/ora/920
diskfarm01:/svrdev_wkspc/as 46080000 40242920 5837080   88%      /fn/as
```

NOTE: Pay special attention to the 'avail/free' column for key filesystems /, /tmp, /var and /fnsw/local



## dmesg.YYYYMMDD.##.wri

The following illustration shows a sample dmesg report.

```
COMMAND: dmesg / DATE: Thu Jul 21 15:23:35 PDT 2005
```

```
Thu Jul 21 15:23:35 PDT 2005
```

```
Jul  1 11:00:09 becks genunix: [ID 936769 kern.info] pcipsyl is /pci@1f,2000
Jul  1 11:00:09 becks scsi: [ID 365881 kern.info] /pci@1f,4000/scsi@3 (glm0):
Jul  1 11:00:09 becks   Rev. 5 Symbios 53c875 found.
Jul  1 11:00:09 becks scsi: [ID 365881 kern.info] /pci@1f,4000/scsi@3 (glm0):
Jul  1 11:00:09 becks   target1-scsi-options=0x5f8
Jul  1 11:00:09 becks scsi: [ID 365881 kern.info] /pci@1f,4000/scsi@3 (glm0):
Jul  1 11:00:09 becks   target2-scsi-options=0x5f8
Jul  1 11:00:09 becks scsi: [ID 365881 kern.info] /pci@1f,4000/scsi@3 (glm0):
Jul  1 11:00:09 becks   target3-scsi-options=0x5f8
Jul  1 11:00:09 becks scsi: [ID 365881 kern.info] /pci@1f,4000/scsi@3 (glm0):
Jul  1 11:00:09 becks   target4-scsi-options=0x5f8
Jul  1 11:00:09 becks scsi: [ID 365881 kern.info] /pci@1f,4000/scsi@3 (glm0):
Jul  1 11:00:09 becks   target5-scsi-options=0x5f8
Jul  1 11:00:09 becks scsi: [ID 365881 kern.info] /pci@1f,4000/scsi@3 (glm0):
Jul  1 11:00:09 becks   target6-scsi-options=0x5f8
Jul  1 11:00:09 becks pcipsy: [ID 370704 kern.info] PCI-device: scsi@3, glm0
```

**UNIX****env.fnsw.all.YYYYMMDD.##.wri**

The following illustration shows a sample env.fnsw.all report. This is all the environment variables defined for the fnsw user.

```
COMMAND: su - fnsw -c env / DATE: Thu Jul 21 15:24:11 PDT 2005
Sun Microsystems Inc.   SunOS 5.8           Generic February 2000
ORACLE_TERM = (vt100)  _=/usr/bin/env
MANPATH=/usr/man:/usr/local/slate/doc:/usr/local/newsprint/man:/usr/dt/man:/usr/
motif/docs/man:/usr/guide/man:/usr/local/man:/usr2/xpilot:/usr/local/openwin/man
NLSPATH=/fnsw/lib/nls/msg/%L/%N:/fnsw/lib/nls/msg/%N
HZ=100
LC_TIME=C
VISUAL=vi
XVTPATH=/fnsw/xvt/print
PATH=/fnsw/client/bin:/fnsw/local/bin:/fnsw/bin:/fnsw/etc:/fnsw/lib/perf:/fnsw/s
upport:/usr/dt/bin:/opt/SUNWspro/bin:/opt/SUNWmotif/bin:/usr/bin:/etc:/usr/sbin:
/usr/ucb:/usr/local/bin:/usr/openwin/bin:/usr/ora/920/bin:/bin:/usr/ccs/bin
MOTIFHOME=/usr/dt
UIDPATH=/fnsw/lib/X11/%L/uid/%U:/fnsw/lib/X11/uid/%U:/usr/dt/lib/uid/%U:%U
OPENWINHOME=/usr/openwin
EDITOR=vi
LOGNAME=fnsw
ORACLE_UID=oracle
ORACLE_SID=IDB
SHELL=/bin/ksh
ORACLE_TERM=vt100
XFILESEARCHPATH=/usr/dt/lib/%T/%N%S
HOME=/usr/fnsw
LD_LIBRARY_PATH=/usr/dt/lib:/usr/openwin/lib:/:usr/lib:/opt/SUNWmotif/lib:/usr/o
ra/920/lib32:/usr/ora/920/lib
TERM=vt100
XAPPLRESDIR=/fnsw/lib/X11/%L:/fnsw/lib/X11
ORACLE_HOME=/usr/ora/920
PWD=/usr/fnsw
TERMINFO=/usr/share/lib/terminfo
TZ=PST8PDT
ENV=/usr/fnsw/.env
XMBINDDIR=/usr/dt/etc/key_bindings
```

**UNIX****env.fnsw.fn.YYYYMMDD.##.wri**

The following illustration shows a sample env.fnsw.fn report. This is the IS-specific environment variables defined for the fnsw user.

```
HOME=/usr/fnsw
NLSPATH=/fnsw/lib/nls/msg/%L/%N:/fnsw/lib/nls/msg/%N
ORACLE_HOME=/usr/ora/920
ORACLE_SID=IDB
PATH=/fnsw/client/bin:/fnsw/local/bin:/fnsw/bin:/fnsw/etc:/fnsw/lib/perf:/fnsw/s
upport:/usr/dt/bin:/opt/SUNWspro/bin:/opt/SUNWmotif/bin:/usr/bin:/etc:/usr/sbin:
/usr/ucb:/usr/local/bin:/usr/openwin/bin:/usr/ora/920/bin:/bin:/usr/ccs/bin
TERM=vt100
TZ=PST8PDT
```



## env.root.all.YYYYMMDD.##.wri

The following illustration shows a sample env.root.all report. This is all the environment variables defined for the root user.

```
COMMAND: su - root -c env / DATE: Thu Jul 21 15:24:10 PDT 2005
Sun Microsystems Inc.   SunOS 5.8           Generic February 2000
ORACLE_TERM = (vt100) EDITOR=vi
ENV=~/.env
HOME=/
HZ=100
LC_TIME=C
LOGNAME=root
NLSPATH=/fnsw/lib/nls/msg/%L/%N:/fnsw/lib/nls/msg/%N
ORACLE_HOME=/usr/ora/920
ORACLE_SID=IDB
ORACLE_TERM=vt100
ORACLE_UID=oracle
PATH=/fnsw/bin:/fnsw/etc:/fnsw/lib/perf:/fnsw/support:/usr/local/bin:/usr/dt/bin
:/opt/SUNWspro/bin:/opt/SUNWmotif/bin:/usr/ora/920/bin:/usr/bin:/etc:/usr/sbin:/
usr/ucb:/usr/openwin/bin:/bin:/usr/ccs/bin
SHELL=/sbin/sh
TERM=vt100
TERMINFO=/usr/share/lib/terminfo
TZ=PST8PDT
UIDPATH=/fnsw/lib/X11/%L/uid/%U:/fnsw/lib/X11/uid/%U:/usr/dt/lib/uid/%U:%U
VISUAL=vi
XAPPLRESDIR=/fnsw/lib/X11/%L:/fnsw/lib/X11
XFILESEARCHPATH=/usr/dt/lib/%T/%N%S
XMBINDDIR=/usr/dt/etc/key_bindings
XVTPATH=/fnsw/xvt/print
```

**UNIX****env.root.fn.YYYYMMDD.##.wri**

The following illustration shows a sample env.root.fn report. This is the IS-specific environment variables defined for the root user.

```
HOME=/
NLSPATH=/fnsw/lib/nls/msg/%L/%N:/fnsw/lib/nls/msg/%N
ORACLE_HOME=/usr/ora/920
ORACLE_SID=IDB
PATH=/fnsw/bin:/fnsw/etc:/fnsw/lib/perf:/fnsw/support:/usr/local/bin:/usr/dt/bin
:/opt/SUNWspro/bin:/opt/SUNWmotif/bin:/usr/ora/920/bin:/usr/bin:/etc:/usr/sbin:/
usr/ucb:/usr/openwin/bin:/bin:/usr/ccs/bin
TERM=vt100
TZ=PST8PDT
```

**fn\_procs.YYYYMMDD.##.wri**

The following shows a sample fn\_procs report on a UNIX system. This program reports on files in /fnsw/procs and notes if any program listed has died.

```
00023885: TM_daemon -s (fnsw) 2005/07/22 10:00:00
00023998: OCOR_Listen -pt -s32769 -t3600 -d20 (fnsw) 2005/07/22 10:00:25
00023888: /fnsw/bin/ilk_daemon (fnsw) 2005/07/22 10:00:04
00023889: MKF_clean (fnsw) 2005/07/22 10:00:04
00023890: MKF_writer 0 (fnsw) 2005/07/22 10:00:04
00023974: MKF_writer 3 (fnsw) 2005/07/22 10:00:23
00024002: OCOR_Listen -pt -s32769 -t3600 -d20 (fnsw) 2005/07/22 10:00:25
00023968: NCH_daemon -pt (fnsw) 2005/07/22 10:00:22
00023987: CSM_daemon (fnsw) 2005/07/22 10:00:24 **dead**
```

The following shows a sample `fn_procs` report on a Windows Server system.

```
COMMAND: fn_procs TIME: 11:07:23.76
00001024: C:\FNSW\bin\tm_daemon.exe (fnsw) 2005/07/22 11:05:57
00002788: MKF_writer 3 (fnsw) 2005/07/22 11:07:01
00002860: ipc_daemon (fnsw) 2005/07/22 11:06:57
00002868: MKF_writer 0 (fnsw) 2005/07/22 11:06:57
00002888: INXu (fnsw) 2005/07/22 11:07:00
00002908: C:\fnsw\bin\fn_procs (fnsw) 2005/07/22 11:07:23
00002932: MKF_clean (fnsw) 2005/07/22 11:06:57
00003040: bes_commit 2 Bes1:cassini:FileNet (fnsw) 2005/07/22 11:07:00
00003300: perf_mon -f/fnsw/local/sd/perf_mon.script -f/fnsw/lib/perf/perf_
mon.script (fnsw) 2005/07/22 11:07:00
00003312: NCH_daemon -pt (fnsw) 2005/07/22 11:07:00
00003340: SEC_daemon (fnsw) 2005/07/22 11:07:00
00003356: INXbg -s IndexServer (fnsw) 2005/07/22 11:07:00
00003364: ds_init (fnsw) 2005/07/22 11:07:00
00003372: CSM_daemon (fnsw) 2005/07/22 11:07:00
00003384: PRI_daemon (fnsw) 2005/07/22 11:07:00
00003392: fn_snmpd.exe (fnsw) 2005/07/22 11:07:00
00003412: OCOR_Listen -pt -s32769 -t3600 -d20 (fnsw) 2005/07/22 11:07:02
00003420: fn_trapd.exe (fnsw) 2005/07/22 11:07:00
00003432: MKF_writer 2 (fnsw) 2005/07/22 11:07:01
00003436: MKF_writer 1 (fnsw) 2005/07/22 11:07:01
00003468: PRI_notify (fnsw) 2005/07/22 11:07:02
00003476: PRI_check (fnsw) 2005/07/22 11:07:02
00003488: PRI_worker (fnsw) 2005/07/22 11:07:02
00003496: dtp_tran (fnsw) 2005/07/22 11:07:03
00003504: ds_notify (fnsw) 2005/07/22 11:07:03
00003512: ds_notify (fnsw) 2005/07/22 11:07:03
00003520: rmt_commit (fnsw) 2005/07/22 11:07:03
00003556: fbc_commit (fnsw) 2005/07/22 11:07:03
00003564: osi_migrate (fnsw) 2005/07/22 11:07:03
00003608: bes_commit 1 Bes1:cassini:FileNet (fnsw) 2005/07/22 11:07:05
Number of FileNet processes: 30
```

The UNIX logo consists of the word "UNIX" in white, bold, sans-serif capital letters, centered on a red, parallelogram-shaped background.**fnl.YYYYMMDD.##.wri**

The following illustration shows a sample FNL display report.

```
COMMAND: FNL_disp -p / DATE: Fri Jul 22 10:46:47 PDT 2005
Dumping all process records
pros_rec[0] @ 0x5E002B60:
  myinx          0
  pid            0
  generation     0
  next_p         0x00000000
  prev_p         0x00000000
  wait_ilk_p     0x00000000
  wait_type      0 (WAIT_NONE)
  trying         0 (FALSE)
  pros_owned_ilk_p: 0x00000000
  msgsqid        0
  waitcount      0
  pause_spinlock 0x00000000
  pause          0
  write_in_prog  0
  cs_ilk [0]:    0x00000000
  cs_ilk [1]:    0x00000000
  cs_ilk [2]:    0x00000000
  args
  pr_status      0 (AVAILABLE RECORD) OK
  start_time     NULL
```



**WIN****fnl\_disp.YYYYMMDD.##.wri**

The following illustration shows a sample fnl\_disp report.

```
COMMAND: fnl_disp -i -p [-b] TIME: 11:07:20.57
Running fnl_disp -i
```

```
Dumping all interlock records
interlock_rec[0] @ 0x46008570:
  lockint                0
  claim_count            0
  claimq_p               0x00000000
  cs_count               0
  shareq_p              0x00000000
  validation             0x00000000
  last_claim_pid         0 (0x0)
  last_claim_tid         0 (0x0)
  last_claim_tui         0 (0x0)
  handle_addr            0x00000000
  next_pros_ilk_p        0x00000000
...
```

**UNIX****fns\_w\_procs.YYYYMMDD.##.wri**

The following illustration shows a sample fns\_w\_procs report.

```
/fns_w_procs: Fri Jul 22 10:46:47 PDT 2005
No orphaned PIDs found
```

**HPUX****getinfo.YYYYMMDD.##.wri**

The following illustration shows a sample getinfo report.

```

handle 11 {
tag Drivers: list KC_DRIVER 224 {
KC_DRIVER {
KC_DRIVER_NAME = "CentIf";
KC_DRIVER_STATUS = 1;
KC_DRIVER_PLANNED = NULL;
KC_DRIVER_TEMPLATE = NULL;
KC_DRIVER_DEFAULT = 0;
...

```

**HPUX****ioscan.YYYYMMDD.##.wri**

The following illustration shows a sample ioscan report.

```

COMMAND: ioscan -fn / DATE: Fri Jul 15 15:12:07 PDT 2005
Class      I  H/W Path      Driver      S/W State H/W Type  Description
=====
bc          0                root        CLAIMED   BUS_NEXUS
bc          1  8             ccio        CLAIMED   BUS_NEXUS I/O Adapter
bc          2  8/0           bc          CLAIMED   BUS_NEXUS Bus Converter
tty         0  8/0/0         mux2        CLAIMED   INTERFACE MUX
           /dev/diag/mux0  /dev/diag/tty0p7 /dev/tty0p1
           /dev/diag/tty0p0 /dev/mux0         /dev/tty0p7
           /dev/diag/tty0p1 /dev/tty0p0
ext_bus     0  8/4           c720        CLAIMED   INTERFACE GSC add-on Fast/Wide
SCSI Interface
target      0  8/4.5         tgt         CLAIMED   DEVICE
disk        0  8/4.5.0       sdisk       CLAIMED   DEVICE     SEAGATE ST39236LC
           /dev/dsk/c0t5d0  /dev/rdisk/c0t5d0

```

**WIN****ipc\_tool1.YYYYMMDD.##.wri and ipc\_tool2.YYYYMMDD.##.wri**

ipc\_tool is basically the same as the UNIX ipci tool below. It is run twice to see if status has changed. As a consequence there are two different file names: ipc\_tool1.YYYYMMDD.##.wri and ipc\_tool2.YYYYMMDD.##.wri.

The following illustration shows a sample ipc\_tool report.

```
COMMAND: ipc_tool -i -s -t -a TIME: 11:07:19.01
Running ipc_tool -i

Interlock [38] created by MKF "MKF 0 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0 max: 0 min: -1)
    Claim queue (oldest first)
        [1] MKF_writer 0 (0xb34.b70) (2868.2928.3) [alive]

Interlock [53] created by MKF "MKF 1 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0 max: 0 min: -1)
    Claim queue (oldest first)
        [1] MKF_writer 1 (0xd6c.d74) (3436.3444.32) [alive]

Interlock [66] created by MKF "MKF 2 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0 max: 0 min: -1)
    Claim queue (oldest first)
        [1] MKF_writer 2 (0xd68.d70) (3432.3440.33) [alive]

...
```

**UNIX****ipci.YYYYMMDD.##.wri**

ipc\_tool is your most important tool for debugging halted processes or errors caused by insufficient service request handlers. This tool generates two ipc\_tool reports. ipc\_tool is run twice to see if status has changed and both runs output to the same file. To review these reports:

- Look for any processes marked **dead**.
- Look for any long queues (3 or more processes) hanging on an interlock.
- Look at the bottom of each report to see if Image Services software is at or near its maximum quota of interlocks.
- Compare the top and bottom reports to see if things are moving, or if they're completely halted.
- Refer back to cormon to correlate any suspicious stubs with the client PC that might have started them.

The following illustration shows a portion of a sample ipc report.

```
COMMAND: ipc_tool -i / DATE: Thu Jul 21 15:23:35 PDT 2005

Interlock [8] created by OCOR ""
      Claim counts (initial: 0 current: -1 shared: 0 max: 1 min: -1)
      Last claimer: OCOR_Listen -pt -s32769 -t3600 -d20 (5731) [alive]
      Claim queue (oldest first)
          [1] OCOR_Listen -pt -s32769 -t3600 -d20 (5731) [alive]

Interlock [9] created by OCOR ""
      Claim counts (initial: 0 current: 0 shared: 0 max: 0 min: -1)
      Last claimer: (5730) [dead]

Interlock [12] created by MKF "MKF 0 writeahead ilk"
      Claim counts (initial: 0 current: -1 shared: 0 max: 0 min: -1)
      Claim queue (oldest first)
          [1] MKF_writer 0 (5626) [alive]

...
```

WIN**ipconfig.YYYYMMDD.##.log**

ipconfig is the only 911 report file with a .log suffix.

The following illustration shows a sample ipconfig report.

```
COMMAND: ipconfig /all TIME: 11:07:18.29

Windows IP Configuration

    Host Name . . . . . : cassini
    Primary Dns Suffix . . . . . :
    Node Type . . . . . : Hybrid
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No
    DNS Suffix Search List. . . . . : filenet.com

Ethernet adapter Intel Pro 1000 MT Gigabit Ethernet Adapter - onboard:

    Connection-specific DNS Suffix . : filenet.com
    Description . . . . . : Intel(R) PRO/1000 MT Network Connection
    Physical Address. . . . . : 00-C0-9F-19-DA-3F
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IP Address. . . . . : 10.14.51.129
    Subnet Mask . . . . . : 255.255.252.0
    Default Gateway . . . . . : 10.14.48.250
    DHCP Server . . . . . : 10.1.0.48
    DNS Servers . . . . . : 10.1.0.151
                           10.1.0.152
    Primary WINS Server . . . . . : 10.1.0.47
    Secondary WINS Server . . . . . : 10.1.0.48
    Lease Obtained. . . . . : Friday, July 22, 2005 11:05:50 AM
    Lease Expires . . . . . : Friday, July 29, 2005 11:05:50 AM
```

**UNIX****ipcs.YYYYMMDD.##.wri**

The following illustration shows a sample ipcs report.

```
IPC status from <running system> as of Fri Jul 22 10:46:16 PDT 2005
T          ID          KEY          MODE          OWNER          GROUP
Message Queues:
q           0           0           --rw-rw-rw-   root           fnusr
q        36865           0           -Rrw-rw-rw-   root           fnusr
q        36866           0           --rw-rw-rw-   root           fnusr
q        43011           0           --rw-rw-rw-   root           fnusr
q        36868           0           --rw-rw-rw-   fnsw           fnusr
q        36869           0           -Rrw-rw-rw-   root           fnusr
q        40966           0           --rw-rw-rw-   fnsw           fnusr
q        40967           0           --rw-rw-rw-   fnsw           fnusr
```

**UNIX****mkf.YYYYMMDD.##.wri**

Use this report to check out all the MKF databases. Look for:

- Dead processes
- Database fatal error
- Number of **virgin buffers** equal to number of **buffers** and number of **available buffers** equal 0 (database full)

The following illustration shows a portion of a sample mkf report.

```
MCOMMAND: MKF_stats -q /fnsw/dev/1/permanent_db* / DATE: Thu Jul 21 15:24:05 PDT
2005
WARNING: /fnsw/dev/1/permanent_db0 is in normal state.
Statistics may not be completely up to date.

Thu Jul 21 15:24:05 2005

MKF statistics for /fnsw/dev/1/permanent_db0:

Max number of blocks in data base =                25,600
Virgin blocks in data base         =                23,207
Nonvirgin blocks in data base      =                 2,393    9%

Elapsed time:  0 seconds

COMMAND: MKF_stats -q /fnsw/dev/1/transient_db* / DATE: Thu Jul 21 15:24:05 PDT
2005
WARNING: /fnsw/dev/1/transient_db0 is in normal state.
Statistics may not be completely up to date.

Thu Jul 21 15:24:05 2005
...
```



WIN**mkf\_debug.YYYYMMDD.##.wri**

The following illustration shows a sample mkf\_debug report.

```
COMMAND: mkf_debug TIME: 11:07:19.37
Fri Jul 22 11:07:19 2005
All processes using an MKF database on this station:

* if  + in
dead MKF  pid.tid      program name
-----
*      1836.1792    nch_tool nch.dat
*      660.2064     fn_util  whichfn
      3608.3604    bes_commit 1 Bes1:cassini:FileNet
      3556.3524    fbc_commit
      3496.3560    dtp_tran
      3564.3568    osi_migrate
      3520.3516    rmt_commit
      3488.3484    PRI_worker
      3420.3424    fn_trapd.exe
      3476.3492    PRI_check
      3356.3360    INXbg -s IndexServer
...

```

**WIN****mkf\_stats.YYYYMMDD.##.wri**

The following illustration shows a sample mkf\_stats report.

```
COMMAND: mkf_stats -q file TIME: 11:07:19.46
Checking permanent_db0

WARNING: C:\fns\dev\1\permanent_db0 is in normal state.
Statistics may not be completely up to date.

Fri Jul 22 11:07:19 2005

MKF statistics for C:\FNSW\DEV\1\PERMANENT_DB0:

Max number of blocks in data base =                12,800
Virgin blocks in data base         =                10,527
Nonvirgin blocks in data base     =                 2,273   17%

Elapsed time:  0 seconds

Checking transient_db0

WARNING: C:\fns\dev\1\transient_db0 is in normal state.
Statistics may not be completely up to date.
...
```

**WIN****mkf\_tool\_print.YYYYMMDD.##.wri**

The following illustration shows a sample mkf\_tool print report.

```
Outputting to file 'mkf_tool_print.20050722.01.wri'
<MKF_tool>count print_requests*

Total of 0 records counted
<MKF_tool>sel print_requests*
<MKF_tool>sel print_options*
<MKF_tool>q
```

**WIN****nch\_tool.YYYYMMDD.##.wri**

The following illustration shows a sample nch\_tool report. On UNIX, this information is found in the sysconfig.YYYYMMDD.##.wri file

```
COMMAND: nch_tool defaultdomain/netaddresses TIME: 11:07:18.92
nch_tool> defaultdomain
The default domain is cassini:FileNet
nch_tool> netaddresses

[10.14.51.129,32769]
```

**WIN****netstat.YYYYMMDD.##.wri**

The following illustration shows a sample netstat report.

```
COMMAND: netstat -a TIME: 11:07:18.54

Active Connections

Proto Local Address           Foreign Address         State
TCP   cassini:epmap           cassini:0               LISTENING
TCP   cassini:microsoft-ds   cassini:0               LISTENING
TCP   cassini:1027           cassini:0               LISTENING
TCP   cassini:1029           cassini:0               LISTENING
TCP   cassini:1032           cassini:0               LISTENING
TCP   cassini:1311           cassini:0               LISTENING
TCP   cassini:5631           cassini:0               LISTENING
TCP   cassini:8000           cassini:0               LISTENING
TCP   cassini:tms            cassini:0               LISTENING
TCP   cassini:cor            cassini:0               LISTENING
TCP   cassini:netbios-ssn    cassini:0               LISTENING
TCP   cassini:1037           nfsdepot01.eng.filenet.com:netbios-ssn ESTABLISHED
TCP   cassini:1040           hq-eng1.filenet.fn.com:netbios-ssn TIME_WAIT
TCP   cassini:1042           nfsdepot03.eng.filenet.com:netbios-ssn ESTABLISHED
TCP   cassini:1044           fireant.eng.filenet.com:netbios-ssn ESTABLISHED
...
```

**UNIX****netstats.YYYYMMDD.##.wri**

When reviewing this report, look at:

- Any **requests for memory denied**
- Any excessive number of collisions (**Coll**) or number of errors (**Ierrs**, **Oerrs**)
- Unusually small network buffer (on AIX, theWall: should be at least 2048)

The following illustration shows a sample netstats report.

```

COMMAND: netstat -m / DATE: Thu Jul 21 15:23:34 PDT 2005 streams allocation:
                                     cumulative  allocation
                                     current   maximum   total     failures
streams                            405      440      97523     0
queues                             912      962     233978     0
mblk                                891     2413     284791     0
dblk                                886     2451     21248795  0
linkblk                             14      169       103       0
strevent                            9       169     1347140   0
syncq                               29       50       4447     0
qband                               2       127        2       0

1188 Kbytes allocated for streams data

COMMAND: netstat -in / DATE: Thu Jul 21 15:23:34 PDT 2005
Name Mtu Net/Dest Address Ipkts Ierrs Opkts Oerrs Collis Queue
lo0  8232 127.0.0.0 127.0.0.1 1524671 0 1524671 0 0 0
hme0 1500 10.14.48.0 10.14.48.28 531886 4 277027 0 25861 0

#/TIME_WAIT connections: 10
#/CLOSE_WAIT connections: 1

```

**AIX****no.YYYYMMDD.##.wri**

The following illustration shows a sample no report.

```
COMMAND: no -a / DATE: Wed Jul 27 14:43:29 PDT 2005
      arpgsize = 12
      arpt_killc = 20
      arptab_bsiz = 7
      arptab_nb = 73
      bcastping = 0
      clean_partial_conns = 0
      delayack = 0
      delayackports = {}
      dgd_packets_lost = 3
      dgd_ping_time = 5
      dgd_retry_time = 5
      directed_broadcast = 0
      extendednetstats = 0
      fasttimo = 200
      icmp6_errmsg_rate = 10
      icmpaddressmask = 0
      ie5_old_multicast_mapping = 0
```

**AIX****oslevel.YYYYMMDD.##.wri**

The following illustration shows a sample oslevel report.

```
COMMAND: oslevel -r / DATE: Wed Jul 27 14:43:29 PDT 2005
5200-02
```

**perf\_report\_a.YYYYMMDD.##.wri**

The following illustration shows a sample perf\_report -a report on a UNIX system. The report looks very similar on a Windows Server system.

```
COMMAND: perf_report -a / DATE: Thu Jul 21 16:47:09 PDT 2005
```

```
ReportsDir=/fnsw/local/logs/perf/
```

```
REPORTS GENERATED
```

Name	Title	File
sol_sum	Solaris Processor Summary Report	cmb1.sol_sum.Jul21.txt
sol_vmm	Solaris Virtual Memory Activity	cmb1.sol_vmm.Jul21.txt
css	CSS Summary Report	cmb1.css.Jul21.txt
sv	Service activity	cmb1.sv.Jul21.txt
bs	Batch Entry Report	cmb1.bs.Jul21.txt
rbs	Batch Entry Server Report	cmb1.rbs.Jul21.txt
transdb_io	Transient DB I/O Tuning Report	cmb1.transdb_io.Jul21.txt
transdb_trans	Transient DB Transaction Tuning	cmb1.transdb_trans.Jul21.txt
csm	Cache Services Report	cmb1.csm.Jul21.txt
doc	Document Services Report	cmb1.doc.Jul21.txt
fb	Fast Batch Throughput Report	cmb1.fb.Jul21.txt
inx	Index Services Report	cmb1.inx.Jul21.txt
nch	Network Clearinghouse Report	cmb1.nch.Jul21.txt
ds	Document Services Summary	cmb1.ds.Jul21.txt
pf	Image Prefetch Activity Report	cmb1.pf.Jul21.txt
cpr	Client Page Request Report	cmb1.cpr.Jul21.txt
ol	OSAR Load Summary	cmb1.ol.Jul21.txt
permdb_io	Permanent DB I/O Tuning Report	cmb1.permdb_io.Jul21.txt
permdb_trans	Permanent DB Transaction Tuning	cmb1.permdb_trans.Jul21.txt
prt	Print Activity Report	cmb1.prt.Jul21.txt
wfl	WorkFlo Activity Report	cmb1.wfl.Jul21.txt
sq	SQL Services Report	cmb1.sq.Jul21.txt
sec	Security Services Report	cmb1.sec.Jul21.txt
secdb_io	Security DB I/O Tuning Report	cmb1.secdb_io.Jul21.txt
secdb_trans	Security DB Transaction Tuning	cmb1.secdb_trans.Jul21.txt

**perf\_report\_rful.YYYYMMDD.##.wri**

The following illustration shows a sample perf\_report -rful report on a UNIX system. The report looks very similar on a Windows Server system.

```

COMMAND: perf_report -rful / DATE: Thu Jul 21 16:47:06 PDT 2005
New_Day (00:00-00:00) (1) becks

Base file : /fnsw/local/sd/1/perflog
Date-time : Thu Jul 21 16:30:00 2005

Date-time : Thu Jul 21 16:45:00 2005

Services:      INDEX DOC BATCH CACHE PRI OSAR
              WorkFlo NCH SQL RmtFile Security

Operating System Identification

SYSNAME      NODENAME      RELEASE      VERSION      MACHINE
SunOS        becks         5.8          Generic_108528-12sun4u

priority     % total      % n-idl      ticks      tot-cpu-time
-----
idle         95.291      -----
kernel      3.969       84.287      7145       71.450
user        0.473       10.039      851        8.510
wait        0.267       5.674       481        4.810
-----
              100.000      100.000      180002     1800.020

System Activity Information

```

**ppmoi.YYYYMMDD.##.wri**

When reviewing this report, look for any service request handlers (stubs) where **Total** equals **Max**. This indicates congestion.

The following illustration shows a sample ppmoi report on a UNIX system.

```

COMMAND: PPMOI, rp / DATE: Thu Jul 21 15:23:36 PDT 2005
# Name On Max Alloc OnQ DQ RPipe WPipe CWH Pid(s)
=== =====
0 NCHs 1 64 64 0 0 4 5 8611
1 CSMS 1 64 64 0 0 6 7 6066
2 DOCs 1 64 64 0 0 8 9 6068
3 INXs 1 64 64 0 0 10 11 6070
4 PRIs 1 64 0 0 0 12 13

COMMAND: PPMOI, sh / DATE: Thu Jul 21 15:23:37 PDT 2005
Entry Program Name Program # Ver # Max Total Idle NoPrc NoCul MaxQsz
=====
0: NCHs 2 12 1 1 0 0 64
1: CSMS 134231040 1 12 1 1 0 0 64
2: DOCs 134231041 1 12 1 1 0 0 64
3: INXs 134231042 1 12 1 1 0 0 64
4: PRIs 134231043 1 12 0 0 0 0 64

COMMAND: PPMOI, st / DATE: Thu Jul 21 15:23:37 PDT 2005
# Name Number Vers MaxProc No Proc No Culs
=====
0 NCHs 2 2 12 0 0
Pid Status Conns Avg Time Min Time Max Time
=====
8612 available/idle 44 0.0024 0.0017 0.0107
1 CSMS 134231040 1 12 0 0
Pid Status Conns Avg Time Min Time Max Time
=====
6067 available/idle 7 0.0334 0.0019 0.1402
...
Grand Totals Conns Avg Time Min Time Max Time
=====
157 0.0278 0.0013 0.9906

```



The following illustration shows a sample ppmoi report on a Windows Server system.

```
COMMAND: ppmoi rp, qp, sh, st TIME: 11:07:22.43
```

```
FileNet PPM Operator Interface COMMANDS
```

```
Cleanup - Initiates PPM's process cleanup routine.[cleanup|c]
Debug    - Turns PPM_DEBUG on and off.[debug|d]
debugger- Allows server stub start by itself.[debugger|db]
Help     - Gets you this list.[help|h]
Modify   - Change max or opt value of one entry in table.[modify|m]
Resume   - Restores all max values to original setting.[resume|r]
SHow     - Displays all the entries in the PPM table.[show|sh]
SStatus  - Displays server stub process status.[status|st]
CULstatus - Displays CUL status.[culstatus|cul]
SUSpend  - Changes all max values in the PPM table to 0.[suspend|su]
RPCqueue- Display entries in RPC queues.[rpcqueue|rp]
QPerf    - Display RPC queue performance statistics.[qperf|qp]
Quit     - Terminates the Operator Interface session.[quit|q]
```

```
PPMOI> # Name On Max Alloc OnQ DQ RPipe WPipe CWH Pid(s)
```

```
==== ===== == ===== ===== == ===== ===== =====
 0 NCHs 0 0 0 0 0 0 0
 1 CSMS 0 0 0 0 0 0 0
 2 DOCs 0 0 0 0 0 0 0
 3 INXs 0 0 0 0 0 0 0
 4 PRIs 0 0 0 0 0 0 0
 5 BESS 0 0 0 0 0 0 0
 6 PSMS 0 0 0 0 0 0 0
```

```
...
```

```
PPMOI> # Name Number Vers MaxProc No Proc No Culs
```

```
==== ===== ===== ===== ===== =====
No active server stub processes
```

**UNIX****pri\_diags.YYYYMMDD.##.wri**

This report prints out a report as a result of running the 911 command with the -p option.

The following illustration shows a sample pri\_diags report.

```

MKF_tool, print_requests, print_options: Fri Jul 29 11:18:51 PDT 2005
Outputting to file '/tmp/20970.tmp'
<MKF_tool>count print_requests*
Total of 0 records counted
<MKF_tool>sel print_requests*
<MKF_tool>sel print_options*
<MKF_tool>quit

-----
PRI_tool, print requests: Fri Jul 29 11:18:51 PDT 2005
Outputting to file '/tmp/20970.tmp' and standard out
pr * long
p
#   printer                print    fetch requests    pages    idle
   name                    request request  queued    queued    time
-----
r
request total    pages status    printer
   id pages  printed    name
-----
quit

```

**WIN****pri\_tool.YYYYMMDD.##.wri**

The following illustration shows a sample pri\_tool report.

```

Outputting to file 'pri_tool.20050722.01.wri' and standard out
pr * long
p
# printer                print    fetch requests  pages      idle
  name                   request request  queued      queued     time
-----                -
r
request total    pages status    printer
   id  pages  printed      name
-----
q

```

**UNIX****ps.YYYYMMDD.##.wri**

This report prints out comprehensive information about each process currently known to UNIX:

- Use the ps report to isolate a suspicious process you might have discovered in cormon or ipc\_tool.
- Check the **SZ** (size) field to identify processes consuming excessive memory.
- Check the sheer number of processes. (Over 150-200 processes can adversely affect a moderate-size system.)
- Pay special attention to Image Services and Oracle processes; look for anything unusual (such as excessive size or default status).

The following illustration shows a sample ps report.

```

COMMAND: ps -eaf1 / DATE: Thu Jul 21 10:13:59 PDT 2005
  F S  USER   PID  PPID  C  PRI  NI ADDR  SZ   WCHAN  STIME  TTY  TIME CMD
 203803 S  root     1    0    0  60  20 1405  232          Aug 12  - 20:48 /
etc/init
 260801 S  root  1384    1    0  60  20 751d  188          Aug 12  - 0:00 /
etc/srcmstr
 261801 S  root  2421  1384    0  60  20 6539  260          Aug 12  - 0:03 /
usr/lib/sendmail -bd -q30m 240801 S  root  2834    1  0  60  20 58d6  64
5ac3e98 Aug 12  - 7:25 /etc/syncd 60
 241801 S  root  3202    1    0  60  20 5154  152 594c924 Aug 12  - 1:03 /
etc/cron
 240801 S  root  3463    1    0  60  20 3d6f   36  46f10 Aug 12  - 0:00 /
etc/uprintfd
 42801 S  root  3604    1    0  60  20 74dd  260    d4f8 Aug 12  - 0:02 /
usr/lib/errdemon
 240801 S  fnsw 12748 12234    0  60  20 65f9  512 5a12a24 Aug 14  - 0:00
oracleIDB P:4096,7,10,
 261801 S  fnsw 13462    1    0  60  20 41f0  164          Aug 14  - 1:47
NCH_daemon -pt

```

**HPUX****sar.YYYYMMDD.##.wri**

The following illustration shows a sample sar report.

```
COMMAND: /usr/bin/sar -v 5 3 / DATE: Fri Jul 15 15:11:02 PDT 2005
```

```
HP-UX panda B.11.11 U 9000/820 07/15/05
```

```
15:11:02 text-sz ov proc-sz ov inod-sz ov file-sz ov
15:11:07 N/A N/A 181/1005 0 933/1085 0 722/5010 0
15:11:12 N/A N/A 181/1005 0 933/1085 0 722/5010 0
15:11:17 N/A N/A 181/1005 0 933/1085 0 722/5010 0
```

```
COMMAND: /usr/bin/sar -u 5 3 / DATE: Fri Jul 15 15:11:18 PDT 2005
```

```
HP-UX panda B.11.11 U 9000/820 07/15/05
```

```
15:11:18 %usr %sys %wio %idle
15:11:23 0 0 0 100
15:11:28 0 0 1 99
15:11:33 0 0 0 100
Average 0 0 0 100
```

```
COMMAND: /usr/bin/sar -d 5 3 / DATE: Fri Jul 15 15:11:33 PDT 2005
```

```
HP-UX panda B.11.11 U 9000/820 07/15/05
```

```
15:11:33 device %busy avque r+w/s blks/s await avserv
15:11:38 c0t5d0 10.00 2.39 14 112 17.36 41.47
c0t8d0 0.80 0.50 1 10 3.07 12.70
15:11:43 c0t5d0 9.38 0.95 12 96 10.93 46.14
c0t8d0 1.20 0.50 1 16 2.57 13.76
15:11:48 c0t5d0 12.83 1.00 20 115 8.51 28.57
c0t8d0 1.60 0.50 2 20 2.96 13.02
...
```

**sec\_tool.YYYYMMDD.##.wri**

The following illustration shows a sample sec\_tool report on a UNIX system. The report looks very similar on a Windows Server system.

```

COMMAND: SEC_tool / DATE: Fri Jul 22 10:46:48 PDT 2005
=====
2 sessions found on Fri Jul 22 10:46:48 2005
=====

user_name                               endpoint_name
-----                               -
SysAdmin:becks:FileNet                  WS001.0000@10.14.39.1
ServiceProcess:System:System            ServProcTerm@0.0.0.0

1 who

```

**WIN****set.YYYYMMDD.##.wri**

The following illustration shows a sample set report. This shows all the environment variables currently set.

```

COMMAND: set TIME: 11:07:15.48
ALLUSERSPROFILE=C:\Documents and Settings\All Users
APPDATA=C:\Documents and Settings\fnsw\Application Data
CLIENTNAME=Console
ClusterLog=C:\WINDOWS\Cluster\cluster.log
CommonProgramFiles=C:\Program Files\Common Files
COMPUTERNAME=CASSINI
ComSpec=C:\WINDOWS\system32\cmd.exe
datestring=20050722
DellAgentPath=C:\Program Files\Dell\OpenManage\OMSA\bin
dirdate=07/22/2005
fnswlibdrive=C
fnswlocaldrive=C
...

```

**WIN****set.fn.YYYYMMDD.##.wri**

The following illustration shows a sample set.fn report. This shows all the IS-specific environment variables currently set.

```
COMMAND: Get FileNet environment variables TIME: 11:07:15.50
```

**WIN****sharedSection.YYYYMMDD.##.wri**

The following illustration shows a sample sharedSection report. This shows the registry entry containing values for desktop heap sizes. The SharedSection entry will have 3 or four values.

1. Max size of system wide heap (in KB).
2. Size of per desktop heap.
3. Non-interactive desktop heap size (FileNet requires min. 512 KB).
4. Windows Terminal Server heap size.

```
COMMAND: Registry entry HKLM\System\CurrentControlSet\Control\Session
Manager\Subsystems\Windows TIME: 11:07:23.93
%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
SharedSection=1024,3072,512 Windows=On SubSystemType=Windows ServerDll=basesrv,1
ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2 ProfileControl=Off
MaxRequestThreads=16
```

**UNIX****stack.YYYYMMDD.##.wri**

The following illustration shows a sample stack report of stack traces of files found under /fnsw or the fnsw user home directory. The following debuggers are used:

AIX	-	dbx
HP	-	gdb
Solaris	-	dbx

If these debuggers are not found, adb is used.



```
COMMAND: Getting stack traces / DATE: Fri Jul 22 10:45:27 PDT 2005

Running dbx /fnsw/bin/SDS_CSAR_reader /fnsw/local/logs/TM_daemon/core.GsaG.U
changes (topic)
The major new features of this release relative to 6.1 are:

o The dbx environment variable "rtc_use_traps" has been removed.
  Traps are used automatically as long as the hardware is UltraSparc.
o There is now some basic support for looking at core files from
  different versions of Solaris. See `help core mismatch' for details.
o Stepping to a specific function support. See `help step to' for details.
o Fortran intrinsics support on Intel has been removed.
o Dbx automatically translates some old dbxenv variables to their new
  names. See `help dbxenv' for details.
o The default command-line editing mode is now emacs. See "How do I enable
  command-line editing?" in FAQ for details (`help FAQ').
o Two new collector commands have been implemented: `pause' and `resume'.
  See `help collector' for details.
o The collector enable_once command has been removed.
o The collector store command has been extended to include experiment groups.

See also `help changes61'

To suppress this message, add the following line to your .dbxrc file:

    dbxenv suppress_startup_message 6.2
Reading SDS_CSAR_reader
core file header read successfully
Reading ld.so.1
Reading libFPLibrary.so
Reading libPAI_module32.so
Reading libFPParser32.so
Reading libSDS.so
Reading libCSM.so
Reading libCKS.so
...
```

**UNIX****sysconfig.YYYYMMDD.##.wri**

This report helps you start troubleshooting any Image Services software problem. It displays system information such as:

- System platform and OS version (**fn\_util whichfn**)
- Image Services version (**fn\_util whichfn**)
- Number of Image Services servers
- OS-specific error reporting and logical volume commands
- Output from sgs
- Output from nch\_tool

The following illustration shows a portion of a sample sysconfig report. This report lists all trigger and configuration files found and their contents.

```

COMMAND: uname -a / DATE: Thu Jul 21 15:23:37 PDT 2005
UNAME: SunOS becks 5.8 Generic_108528-12 sun4u sparc SUNW,Ultra-250

Solaris Base OS Release:

    SunOS 5.8 Generic_108528-12

Oracle RDBMS Release:

    FileNet Controlled, 9.2.0.1.0

FileNet Image Services Release:

    4.0 GA Service Pack 3 Hot Fix Pack 1 (4.0.30)

The number of Servers configured on this system:          1

Sta #   Station_Type           Server Name       Address(es)
-----  -
    1   Root/Index/Document/Osar  becks            10.14.48.28
The number of Servers configured on this system:          1

Server_Name   Partition_Name           Mbytes
-----
becks:
           /fnsw/dev/1/cache1      3072
           /fnsw/dev/1/transient_db1    40
           /fnsw/dev/1/oracle_tr0      400
...

```

### trigger.YYYYMMDD.##.wri

The following illustration shows a sample trigger report on a UNIX system. This report lists all IS trigger and configuration files found and their contents.

```
/fnsw/local/cold/config/fonts exists - 0 bytes

/fnsw/local/sd/ims_start exists - 202 bytes
echo "Starting cold_3770 daemon:" >> $OFIL
/fnsw/bin/sys_log "Starting COLD_3770 daemon"
nohup /fnsw/bin/cold_3770 -d /fnsw/local/cold/data/test > /dev/null &
echo "Initialization complete." >> $OFIL

/fnsw/local/sd/inx_conf exists - 58 bytes
no_catalog = 0
ixwfq_ctl_file = 1
multi_byte_security = 1

/fnsw/local/sd/root_station exists - 0 bytes

/fnsw/local/sd/l/sds_conf exists - 148 bytes
SDS 1 {
sds_name CSAR_reader
worker SDS_CSAR_reader 4
info "WorkingDirectory=/csar emcdomain=10.10.16.9,10.10.16.1"
sds_priority high
debug max
}

/fnsw/local/sd/snt.chkpt exists - 26 bytes
100763 3020 15 1121967986

/fnsw/local/ssn exists - 7 bytes
101755
```

The following illustration shows a sample trigger report on a Windows Server system.

```
C:\fnsw_loc\sd\inx_conf exists - 61 bytes
no_catalog = 0
ixwfq_ctl_file = 1
multi_byte_security = 1

C:\fnsw_loc\sd\snt.chkpt exists - 26 bytes
101003 3000 1 1114722120
```

**UNIX****uname.YYYYMMDD.##.wri**

The following illustration shows a sample uname report.

```
COMMAND: uname -a / DATE: Fri Jul 22 10:46:15 PDT 2005
SunOS becks 5.8 Generic_108528-12 sun4u sparc SUNW,Ultra-250
```

**WIN****ver.YYYYMMDD.##.wri**

The following illustration shows a sample ver report.

```
COMMAND: ver TIME: 11:07:15.48

Microsoft Windows [Version 5.2.3790]
```

**UNIX****vmstats.YYYYMMDD.\***

ipc\_tool, netstat, and vmstats are the three most important reports in 911. Vmstats gives you a thumbnail sketch as to how much real and virtual memory you have, what kind of load the system was under at the moment of crises, and where most of the disk I/O was going.

Check the following:

- uptime: See how long the system has been up. If the system has been running for days (or weeks), the problem might be a memory leak. Try recycling the operating system.
- uptime: If the load average is much higher than 2, the system performance is probably degrading.
- Physical RAM: For example, if a system has only 32 MB RAM and frequently uses the swap file, you should recommend the customer have more memory installed.
- vmstat: Pay particular attention to the **po** (page out) field. If it's consistently greater than 0, the system probably has memory problems.
- iostat: Indicates where most disk activity is occurring: **swap** (insufficient memory), **Oracle datasets** (database access), or **other** (normal user I/O).

**Note**

---

Interpreting virtual memory and I/O performance is a complex task. To help you analyze and diagnose the problem, you should consult with your service representative or performance tuning expert.

---

```

COMMAND: uptime / DATE: Thu Jul 21 15:22:54 PDT 2005
 3:22pm up 13 day(s), 23:05, 5 users, load average: 0.26, 0.08, 0.05

COMMAND: swap -l / DATE: Thu Jul 21 15:22:54 PDT 2005
swapfile          dev  swaplo blocks  free
/dev/vx/dsk/swapvol 176,6      16 4191920 4191920

COMMAND: swap -s / DATE: Thu Jul 21 15:22:54 PDT 2005
total: 529936k bytes allocated + 313904k reserved = 843840k used, 2028704k available

COMMAND: prtconf | grep Memory / DATE: Thu Jul 21 15:22:54 PDT 2005
1024Megabytes

COMMAND: ndd -get /dev/tcp tcp_keepalive_interval / DATE: Thu Jul 21 15:22:54 PDT 2005
7200000

COMMAND: vmstat 5 5 / DATE: Thu Jul 21 15:22:54 PDT 2005
procs      memory          page          disk          faults      cpu
 r  b  w   swap  free  re  mf  pi  po  fr  de  sr  s0  s6  s7  s8   in  sy   cs  us  sy  id
0  0  0  2272664 478856 1   3   4   0   0   0   0   1   0   0   0   416  116   78  0   1  99
0  0  0  2028696 282280 2   1  601  0   0   0   0  212  0   0   0  1763  898  246  0   3  96
0  0  0  2028704 283000 5  30 1155  0   0   0   0  174  0   0   0   941 3856  445  1  13  86
0  0  0  2028704 290016 15 100 995  0   0   0   0  167  0   0   0   922 4756  436  2  14  84
0  1  0  2028704 290320 0   0 1262  0   0   0   0   62  0   0  141 1044 1968 478  0   6  93

COMMAND: iostat 5 5 / DATE: Thu Jul 21 15:23:14 PDT 2005
tty          sd0          sd6          sd7          sd8          cpu
tin tout kps tps serv kps tps serv kps tps serv kps tps serv us sy wt id
 0   1   7   1   9   0   0   0   0   0   0   4   6   0   16   0   1   1  98
 0   0  29   4  10   0   0   0   0   0   0   438 223   4   0   4  47  48
 0   0 136  17   9   0   0   0   0   0   0   330 232   4   0   7  51  41
 0   0 1123 140 1307   0   0   0   0   0   0  1379 296  435   1   7  60  31
 0   0   0   0   0   0   0   0   0   0   0   517 181   4   1  17  34  48
...

```

WIN**whichfn.YYYYMMDD.##.wri**

The following illustration shows a sample whichfn report. On UNIX, this information is found in the sysconfig.YYYYMMDD.##.wri file.

```
COMMAND: fn_util whichfn TIME: 11:07:18.79

Windows Server Release:

Microsoft Windows Server 2003 Service Pack 1

MSSQL Server Release:

FileNet Controlled, SQL Server 2000

FileNet Image Services Release:

4.0 GA Service Pack 3 (4.0.30)
```



**WIN****winmsd.YYYYMMDD.##.wri**

The following illustration shows a sample winmsd report. You can exclude this report by using the **nowinmsd** command line option.

```
System Information report written at: 07/22/05 11:08:21
System Name: CASSINI
[System Summary]

ItemValue
OS NameMicrosoft(R) Windows(R) Server 2003, Enterprise Edition
Version5.2.3790 Service Pack 1 Build 3790
Other OS Description Not Available
OS ManufacturerMicrosoft Corporation
System NameCASSINI
System ManufacturerDell Computer Corporation
System ModelPowerEdge 1600SC
System TypeX86-based PC
Processorx86 Family 15 Model 2 Stepping 7 GenuineIntel ~2791 Mhz
Processorx86 Family 15 Model 2 Stepping 7 GenuineIntel ~2791 Mhz
Processorx86 Family 15 Model 2 Stepping 7 GenuineIntel ~2791 Mhz
Processorx86 Family 15 Model 2 Stepping 7 GenuineIntel ~2791 Mhz
BIOS Version/DateDell Computer Corporation A09, 10/8/2003
SMBIOS Version2.3
Windows DirectoryC:\WINDOWS
System DirectoryC:\WINDOWS\system32
Boot Device\Device\HarddiskVolume2
...
```

**wqs\_tool.YYYYMMDD.##.wri**

The following illustration shows a sample wqs\_tool report on a UNIX system.

```

COMMAND: WQS_tool, dumpqs * *, qstats / DATE: Fri Jul 22 10:46:46 PDT 2005

Type '?' for help

GENERAL COUNTS - Fri Jul 22 10:46:47 2005

-----
              general                dbms                lock
              count    avg time      count    avg time      count avg time

Total Open: 0    Critical Lock: NOT LOCKED (0)
=====

```

The following illustration shows a sample wqs\_tool report on a Windows Server system.

```

COMMAND: wqs_tool dumpqs * *, qstats TIME: 11:07:21.14

Type '?' for help

<WQS_tool><WQS_tool>Total Open: 0    Critical Lock: NOT LOCKED (0)
=====

```

## Checklist

Before you use 911, be aware of the following:

- Image Services software must be running.
- Whenever your system is having a problem, run 911 to capture the data for later analysis.
- You can run 911 immediately when the Image Services system is having undefinable problems, such as when the system runs slowly for no apparent reason.
- The operation could take approximately two minutes to complete.
- You should run 911 on **all** Image Services servers in the FileNet system. One server could cause problems on another one.
- 911 does not require any special setup or syntax. It should already be installed in the /fns/support directory where it can enable execution from any directory through the system's \$PATH).
- 911 does **not** write anything besides log files or alter the customer's system in any way. It just captures diagnostic information.
- You can run 911 more than one time. The log files all have an incremental **version** number, so you don't have to worry about overwriting your last log.
- You can write pre911 and post911 scripts, if desired.

## Procedure

To diagnose system problems using 911, follow these steps:

- 1 At the system prompt, enter:

**911**

- 2 Send the compressed file to your service representative for diagnostics (911.YYYYMMDD.##.Z on UNIX or 911.YYYYMMDD.##.zip on Windows Server).
- 3 Copy the file to the desired location on your system for unpacking.
- 4 Unpack the compressed file by entering:

**uncompress 911.YYYYMMDD.##.Z** (UNIX)  
**tar xvf 911.YYYYMMDD.##**

**unzip 911.YYYYMMDD.##.zip** (Windows Server)  
(or use the standard Windows method of extracting .zip files)

where:

YYYYYY = 4-digit year, such as 1998  
MM = Month, such as 01 for January  
DD = Day, such as 01 for the first of the month  
## = Run number for the day (e.g., 01, 02, etc.)

- 5 Interpret the reports, as shown in **[“Sample Reports” on page 73.](#)**

## Related Topics

Most of the output reports listed in this 911 section have a corresponding tool section in this manual. Refer to these sections for more information on the tools.

## add\_osvr

### Description

The add\_osvr tool adds a new Storage Library server to an existing system, updating surface locator tables in the MKF permanent database to reflect the Storage Library servers you are adding. In addition, add\_osvr adds records to the MKF permanent database media family table of each Storage Library server being added.

The add\_osvr tool does not explicitly assign media or families to a new Storage Library server. Families set up to send write requests to **all** Storage Library servers send write requests to the new Storage Library server, but families that explicitly send write requests to a **specific** Storage Library server do not send write requests to the new Storage Library server. Previously created media remain on the original Storage Library server but you can format new media on the new Storage Library server.

To balance the load on a system, consider running the move\_disk tool to logically move media from one Storage Library server to another. Use the FileNet Database Maintenance application to redirect a family's write request to a new Storage Library server.

If Image Services fails or the system is rebooted before add\_osvr completes, you can rerun add\_osvr.

If add\_osvr fails during execution, correct the problem and rerun add\_osvr until it completes successfully.

**CAUTION** Do not run any other program (such as del\_osvr) after an unsuccessful run of add\_osvr. Doing so leaves the databases in a partially converted state, producing unpredictable results.

---

## Use

Use the add\_osvr tool to add a new Storage Library server to an existing system.

You typically use add\_osvr as part of a procedure that includes use of the del\_osvr and move\_disk tools. You should not use add\_osvr independently. For an example of a typical scenario in which you might use these tools, see [“Typical Scenario” on page 290](#), under the description of the del\_osvr tool.

When add\_osvr completes successfully, the following message displays:

Program terminated successfully.

## Syntax

```
add_osvr <svrid1> <svrid2> ... <svridn>
```

**<svrid1> ...  
<svridn>**

One or more identification numbers for the Storage Library servers to be added to the system.

---

**Note**

Server IDs are assigned during system configuration. You can find the ID assigned to each Storage Library server in the system.ascii file or the NCH database. In the system.ascii file, the server ID for the Storage Library server appears in the unitID field. Alternatively, use the nch\_tool listproperties command to locate the Storage Library server identification number from the NCH database.

---

## Checklist

Before you use add\_osvr, be aware of the following:

- add\_osvr does not explicitly assign media or families to a new Storage Library server.
- Shut down document services before you run add\_osvr.
- Before you run add\_osvr, ensure that NCH is started and MKF permanent and transient databases are accessible on all Storage Library servers.



## Procedure

Perform the following steps to add a Storage Library server to the system:

- 1 Back up the system to tape.

If a failure occurs (for example, the new server does not boot) before `add_osvr` completes successfully, restoring the backup is the only way to return the system to its original state.

- 2 Configure the new Storage Library server.

### UNIX

- For UNIX systems, enter **fn\_edit** on the command line to start the System Configuration Editor.

### WIN

- For Windows Server systems, click on the System Configuration Editor icon.

- 3 Verify system readiness.

- Verify all Storage Library servers are in single-user mode.
- Verify Document Services is shut down.
- Verify that NCH is started.
- Verify MKF permanent and transient databases are accessible on all Storage Library servers.

**4** Start add\_osvr.

At the command line prompt of any Storage Library server on the system, enter **add\_osvr** with appropriate server identification numbers.

The tool has completed successfully when the following message displays:

```
program terminated successfully
```

**5** Restart FileNet software.

When add\_osvr completes, shut down the Storage Library server and reboot the system. Reboot all Storage Library servers to multi-user mode.

As an option, you can also run the move\_disk tool and/or use the FileNet Database Maintenance application to balance the system load.

## Related Topics

[\*\*“del\\_osvr” on page 288\*\*](#)

[\*\*“move\\_disk” on page 826\*\*](#)

See the “Database Maintenance” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21\*\*](#).

## bes\_check

### Description

To ensure consistency between batch components and corresponding MKF transient database table entries, `bes_check` verifies that each batch component has a corresponding entry in its associated batch entry services (BES) tables. However, `bes_check` does not check batch integrity.

If Image Services Batch Services is working correctly, you should never see an “orphaned” batch table row in the transient database BES tables. An orphaned row has no corresponding “parent” row in an associated BES table. An example of an orphaned row is a row in the `batch_image` or `batch_ixval` table whose corresponding row in the `batch_doc` table is missing. The `bes_check` tool also finds objects in BES cache which do not have corresponding rows in the `bes_image` table, or vice versa. In addition, `bes_check` checks the batch header counts against the `batch_doc`, `batch_ixval`, and `batch_ixdir` tables.

At the start of execution, `bes_check` attempts to obtain exclusive use of the batch tables. If batch entry services (BES) is in use and the exclusive lock cannot be obtained, `bes_check` terminates.

The `bes_check` tool verifies that:

- Each image in the cache has an entry in `batch_image` table with the same object ID (a one-to-one relationship).
- Each image whose `doc_id` is nonzero has a corresponding entry in the `batch_doc` table with the same `doc_id` and `batch_id` (a one-to-one relationship). (These numbers start with one; do not confuse them with an object ID.)

- Each index value has a corresponding record in the batch\_doc table (a many-to-one relationship). For each entry in the batch\_docs table, zero or more entries exist in the batch\_ixval table. The num\_indices field in the batch\_doc table determines the number of index values.
- Each of the following tables maintains a one-to-one relationship between the batch\_dyn\_hdr and batch\_stat\_hdr for each document: batch\_ixval, batch\_data, batch\_ixdir, batch\_doc.

If inconsistencies are found, bes\_check writes descriptive messages to a standard output file but makes no attempt to correct the inconsistency.

During the checking process, bes\_check creates two files, .bes\_pre\_sort<pid> and .bes\_sorted<pid>, in which to store a sorted list of batch images. The files are created in the /tmp and /fnsw/local/tmp directories. If bes\_check cannot create these files, a message similar to the following is issued when bes\_check attempts to open the sort files:

```
Cannot open /fnsw/local/tmp/.bes_pre_sort7771 errno = <err>
```

After the message displays, bes\_check terminates. The message could indicate that bes\_check does not have the proper permission level to create the sort files.

If the following message displays, the file system (/fnsw/local/tmp or /tmp) is too small to accommodate the sort lists:

```
Failed the lseek errno = <err>
```

Correct the problem described in the error message and rerun bes\_check.

## Use

Use `bes_check` to perform a consistency check between objects in BES cache and records in transient database batch tables.

Output from `bes_check` normally displays to the standard output device. However, you can redirect the output to a file as shown in the following example for a Windows Server environment:

```
bes_check > \fnsw_loc\tmp\beschk.out
```

Then use the editor of your choice to view the file contents.

## Syntax

```
bes_check
```

## Sample Output

---

```
corona(root)> bes_check
*** BATCH CONSISTENCY CHECK ***
This program checks for orphaned batch components. It does not attempt to clean up
inconsistencies.
This program must be run local to the target batch entry service.
kill -USR1 7771 will cause this program to abort.
The MKF database must be up.
DO NOT run the program if the system is in operation!!!
Start checking Bes1:hpbalboa:FileNet
Verify consistency between cache objects and image headers.
Each object in the cache associated with the Batch Entry Service must have a
corresponding entry in the batch_image table.
Verify consistency between images and documents.
Each entry in batch_image whose doc_id <> 0 must be referenced in batch_doc.
**** Cache/batch_image & batch_image/batch_doc verification complete.****
Verify consistency between documents and images.
Each entry in batch_doc must be exist in batch_image.
**** batch_doc/batch_image verification complete. ****
Verify consistency between index values and documents.
Each entry in batch_ixval must be referenced in batch_doc.
**** batch_ixval/batch_doc verification complete. ****
Verify consistency between documents and batch headers.
Each entry in batch_doc must be referenced in batch_stat_hdr and batch_dyn_hdr.
**** batch_doc/batch_hdr verification complete. ****
Verify consistency between batch headers counts against batch_doc, batch_ixval, and
batch_ixdir tables:
    1) batch_stat_hdr.act_docs == batch_doc (entries)
    2) batch_stat_hdr.num_indices == batch_ixdir (entries)
    3) batch_stat_hdr.act_docs*num_reqd_indcs <= batch_ixval (entries)
**** batch_hdr verification complete. ****
Verify consistency between batch object data, batch_id and object_id for object
types batch, document, image and phase.
**** batch object data verification complete. ****
Program terminated normally.
```

---

In the following sample, bes\_check located orphan rows. The introductory comments and list of image IDs is truncated for readability:

---

```
*** BATCH CONSISTENCY CHECK ***
```

```
This program checks for orphaned batch components. It does not attempt to clean up inconsistencies.
```

```
This program must be run local to the target batch entry service.
```

```
kill -USR1 3958 will cause this program to abort.
```

```
...
```

```
Each entry in batch_image whose doc_id <> 0 must be referenced in batch_doc.
```

```
Image id      111303 of doc          1 in batch_image (batch_id = 184); not in bes  
cache *
```

```
Image id      111304 of doc          1 in batch_image (batch_id = 184); not in bes  
cache *
```

```
...
```

```
Image id      111361 of doc          1 in batch_image (batch_id = 211); not in bes  
cache *
```

```
Image id      111362 of doc          1 in batch_image (batch_id = 211); not in bes  
cache *
```

```
* The batch of the image is in the process of committal, or is committed with errors.  
However, the image may have already been successfully committed.
```

```
**** Cache/batch_image & batch_image/batch_doc verification complete.****
```

```
Verify consistency between documents and images.
```

```
Each entry in batch_doc must be exist in batch_image.
```

```
**** batch_doc/batch_image verification complete. ****
```

```
Verify consistency between index values and documents.
```

```
Each entry in batch_ixval must be referenced in batch_doc.
```

```
**** batch_ixval/batch_doc verification complete. ****
```

```
Verify consistency between documents and batch headers.
```

```
Each entry in batch_doc must be referenced in batch_stat_hdr and batch_dyn_hdr.
```

```
**** batch_doc/batch_hdr verification complete. ****
```

```
Verify consistency between batch headers counts against batch_doc, batch_ixval, and  
batch_ixdir tables:
```

```
1) batch_stat_hdr.act_docs == batch_doc (entries)
```

```
2) batch_stat_hdr.num_indices == batch_ixdir (entries)
```

```
3) batch_stat_hdr.act_docs*num_reqd_indcs <= batch_ixval (entries)
```

```
**** batch_hdr verification complete. ****
```

```
Program terminated normally.
```

---

## Checklist

Before you use bes\_check, be aware of the following:

- Do not run the program if document entry is in progress.
- You must run bes\_check on the document server.
- If you must terminate bes\_check before it completes, use the following command:

**kill -USR1 <pid>**for UNIX platforms

**killfnsw -p <pid>**for Windows Server platforms

where <pid> is the process ID of bes\_check. You can locate the process ID in the output as shown in the following example:

```
This program must be run local to the target batch entry service.  
kill -USR1 3958 will cause this program to abort.
```

**WIN**

In the Windows Server environment, open another MS-DOS window from which to issue the killfnsw -p <pid> command.



## Procedure

The following procedure shuts down the FileNet software before starting bes\_check. You can run bes\_check while FileNet software is active. However, batch entry services (BES) cannot be in use while bes\_check is running.

- 1 Log off Doc Entry at the client.

At this point, Batch Entry Services is still running on the server.

- 2 Click the Task Manager Stop button to shut down FileNet software on the server.
- 3 Start FileNet databases.

At a UNIX server command line prompt or a Windows Server MS-DOS prompt, enter the following command:

```
fn_util startdb
```

- 4 Run bes\_check.
- 5 Restart FileNet software.

When bes\_check completes, click the Task Manager Restart button to restart FileNet software.

## Related Topics

See [“MKF\\_tool” on page 801](#) to view the transient database BES tables.

## bes\_clean

### Description

The bes\_clean tool removes all or selected uncommitted batches.

#### **CAUTION**

---

This tool deletes all data stored in bes\_cache, including the repository used for Capture Professional. If your server supports Capture Professional, you should back up its repository before running bes\_clean. For details, see the backup procedures provided in the Capture Professional documentation.

---

The tool deletes the following records from transient database batch tables:

- Dynamic headers
- Static headers
- Index directory records
- Image records
- Cache objects (images and companion images)
- Index value records
- Document records
- Batch data records
- Batch folder records
- Capture Professional repository

**Note** Folders could have associated batch IDs. However, if you remove all batches by running `bes_clean` with no options, it also removes all folders.

---

At the start of execution, `bes_clean` attempts to obtain exclusive use of the batch tables. If batch entry services (BES) is in use and `bes_clean` cannot obtain the exclusive lock, `bes_clean` terminates.

To delete selected batches, specify the batch ID or the batch name. You can also have `bes_clean` prompt you before it performs the deletion. If you specify a batch ID or batch name, `bes_clean` deletes the specified batch as soon as you respond to the confirmation prompt with a **y**.

At successful completion of the program, `bes_clean` displays a report of deleted records.

## Use

When instructed to do so by your service representative, use `bes_clean` to delete records in the transient database batch tables. You can delete all records or all entries corresponding to a specified batch identifier or batch name. For example, if you have had a magnetic disk failure, your service representative could instruct you to use `bes_clean` to initialize the batch tables in the MKF transient database to match an empty cache.

**CAUTION** Do not use `bes_clean` unless instructed to do so by your service representative.

---

Depending on which options you specify when you enter the `bes_clean` command, the tool either starts immediately or presents a confirmation message. Respond to the confirmation message to continue or terminate the process.

Output from `bes_clean` normally displays to the standard output device. However, you can redirect the output to a file as shown in the following example for a Windows Server environment:

```
bes_clean > \fnsw_loc\tmp\bescIn.out
```

Use the editor of your choice to view the output file contents.

## Syntax

To display the usage statement, enter the tool name followed by a hyphen as shown below:

```
bes_clean -
```

The following syntax statement displays:

```
bes_clean [-y] {-b <batch number> | -B <batch name>} [-v]
```

**-y** Do not display a confirmation message.

If you specify this option, `bes_clean` automatically starts processing.

If you do **not** specify the `-y` option, `bes_clean` issues a confirmation message at startup. Continue or terminate the process by responding to the message.

**-b <batch\_number>**

Batch identifier of the batch for which records are to be deleted. Mutually exclusive with `-B <batch name>`.

- B <batch\_name>** Batch name of the batch for which records are to be deleted. Mutually exclusive with -b <batch number>.
- v** Verbose mode. Displays detailed information during processing.

---

**CAUTION** If you enter the bes\_clean command with no options, a confirmation prompt displays as shown in the example below:

```
corona(fnsw)/usr/fnsw> bes_clean  
Start cleaning Bes1:corona:FileNet  
Enter 'y' to continue:
```

If you reply **y** to the prompt, bes\_clean deletes **all** batches from the transient database. If you press any other key, bes\_clean terminates.

---

## Examples

If bes\_clean is unable to obtain exclusive use of batch tables, the following message displays:

```
corona> bes_clean
bes_clean : Exclusive use is denied.
```

If bes\_clean successfully obtains exclusive use, processing of the batch tables starts immediately. If you do not specify the -y option, a confirmation message displays first:

```
corona> bes_clean
Start cleaning Bes1:corona:FileNet
Enter 'y' to continue : y
```

Enter **y** to continue. Press any other key to terminate bes\_clean.

At successful completion, bes\_clean displays a report indicating the actions taken:

```
3 dynamic headers deleted
3 static headers deleted
3 index directory records deleted
21 image records deleted
21 cache objects deleted
15 index value records deleted
15 document records deleted
```

The following examples delete a specific batch. The first example deletes a batch by its batch ID of 91476, the second deletes a batch by its batch name of b91434.

```
costa5> bes_clean -b 91476
Start cleaning batch with batch_id = '91476' from Bes1:costa5:FileNet
Enter 'y' to continue : y
1 dynamic headers deleted
1 static headers deleted
13 index directory records deleted
4 image records deleted
4 cache objects deleted
18 index value records deleted (ixval may be packed)
2 document records deleted
0 batch object data records deleted
0 folder records deleted
costa5> bes_clean -B b91434
Start cleaning batch = 'b91434' from Bes1:costa5:FileNet
Enter 'y' to continue : y
1 dynamic headers deleted
1 static headers deleted
13 index directory records deleted
4 image records deleted
4 cache objects deleted
18 index value records deleted (ixval may be packed)
2 document records deleted
0 batch object data records deleted
0 folder records deleted
```

The following example also deletes a batch by its batch ID of 91478. However, in this example, the user does not want to be prompted with a confirmation message:

```
costa5> bes_clean -y -b 91478
Start cleaning batch with batch_id = '91478' from
Bes1:costa5:FileNet
1 dynamic headers deleted
1 static headers deleted
13 index directory records deleted
4 image records deleted
4 cache objects deleted
18 index value records deleted (ixval may be packed)
2 document records deleted
0 batch object data records deleted
0 folder records deleted
```



## Checklist

Before you use `bes_clean`, be aware of the following:

- Do not run `bes_clean` unless instructed to do so by your service representative.
- If your Image Services server supports Capture Professional, running `bes_clean` deletes the Capture Professional repository. Before running `bes_clean`, you must back up the Capture Professional repository to your local PC as described in the Capture Professional documentation. After running `bes_clean`, copy the repository back to the Image Services `bes_cache`.
- You must run `bes_clean` on the server that is running Batch Services.
- If you enter the `bes_clean` command with no options, a confirmation prompt displays. If you reply **y** to the prompt, `bes_clean` deletes **all** batches from the transient database. If you reply **n**, `bes_clean` terminates.
- You should not terminate `bes_clean` prior to its completion. However, if you are directed to abort `bes_clean` execution by your service representative, use the Control+c key sequence.

## Procedure

The following procedure shuts down the FileNet software before starting bes\_clean. You can run bes\_clean while FileNet software is active. However, batch entry services (BES) cannot be in use while bes\_clean is running.

- 1 Log off Doc Entry at the client.

At this point, Batch Entry Services is still running on the server.

- 2 Click the Task Manager Stop button to shut down FileNet software.
- 3 Start FileNet databases:

At a UNIX server command line prompt or a Windows Server MS-DOS prompt, enter the following command:

```
fn_util startdb
```

- 4 Run bes\_clean with appropriate options.

If you do not specify any options, bes\_clean displays a confirmation prompt. Reply **y** to delete **all** batches in the transient database. Press any other key to terminate bes\_clean.

- 5 Restart FileNet software.

When bes\_clean completes, click the Task Manager Restart button to restart FileNet software.

## Related Topics

[“bes\\_check” on page 131](#)

[“bes\\_debug” on page 148](#)

[“WRT\\_clean” on page 1341](#)

## bes\_debug

### Description

The bes\_debug program reads the value of the queue column in the batch\_stat\_hdr table of the transient database and displays the status of the batches in a committal or cataloging phase. Batch status can include the following:

- Committing/cataloging
- Waiting Index Services
- Waiting target Index Services
- Waiting Document Services
- Waiting target Document Services

### Use

Use bes\_debug to detect problems with batches that are in the process of committal or cataloging.

### Syntax

```
bes_debug
```

## Sample Output

In the examples below, the “inprogress queue” message refers to the queue column of the batch\_stat\_hdr table.

If no committals are in progress, bes\_debug displays a summary report similar to the following.

```
corona(root)> bes_debug  
Batch Services Bes:Corona:FileNet  
bes_debug completed: 0 batch found in inprogress queue.  
bes_debug completed successfully.
```

If a normal committal is in progress, the report shows the batch\_name, batch\_id, and status information with the summary report.

```
corona(root)> bes_debug  
Batch:  b25,      id=146  
status:  committing  
Batch Services Bes:Corona:FileNet  
bes_debug completed: 1 batch found in inprogress queue.  
bes_debug completed successfully.>
```

If a committal is waiting for a service (or possibly hung), the report displays the service that is needed but has not yet responded.

```
corona(root)> bes_debug
Batch:      b25,      id=146
status:    committing
           waiting Document Services - DocServer:andy:FileNet
           wait_time = 3 seconds
Batch Services Bes:Corona:FileNet
bes_debug completed: 1 batch found in inprogress queue.
bes_debug completed successfully.
```

## Checklist

Before you use `bes_debug`, be aware of the following:

- You must run this tool on the server that is running Batch Services.
- To abort `bes_debug` prior to normal completion, use the Control+c key sequence.

## Procedure

No specific procedure is required.

## bes\_setid

### Description

The `bes_setid` program sets the `batch_id` or the `batch_name_id` in the `batch_ctl_table` to new values. The `batch_ctl_table` keeps track of the next available `batch_id` number and `batch_name_id` number that is assigned by the system during batch creation.

The new value must be greater than any value in the `batch_dyn_hdr` table.

### Use

Use `bes_setid` whenever the transient database is reset and you want to reset the `batch_id` **or** the `batch_name_id` to continue a previous numbering sequence.

### Syntax

```
bes_setid <Batch Service name> [-b | -n] <new_id_value>
```

**<Batch Service name>**

NCH name in the format object:domain:organization

**-b**

Sets the batch ID to the value specified in `<new_id_value>`. Mutually exclusive with `-n`.

**-n**

Sets the name ID to the value specified in `<new_id_value>`. Mutually exclusive with `-b`.

**<new\_id\_value>**

New value of `batch_id` or `batch_name_id`

**Note** To set batch\_id to a new value, the batch\_dyn\_hdr table cannot contain an existing batch with a batch\_id greater than or equal to <new\_id\_value>.

To set batch\_name\_id to a new value, the batch\_dyn\_hdr table cannot contain an existing batch with a batch\_name\_id greater than or equal to <new\_id\_value>.

---

## Sample Output

In the first example below, the batch\_id number in the batch\_ctl\_table is updated to 76. In the second example, the batch\_name\_id is updated to 77.

```
corona(root)/> bes_setid Bes1:corona:FileNet -b 76  
Batch Services Bes1:corona:FileNet...  
bes_setid batch id initialized to 76  
bes_setid completed successfully  
  
corona(root)/> bes_setid Bes1:corona:FileNet -n 77  
Batch Services Bes1:corona:FileNet...  
bes_setid batch name id initialized to 77  
bes_setid completed successfully
```



## Checklist

You must run bes\_setid on the server that is running Batch Services.

## Procedure

No specific procedure is required.

## Related Topics

See [“MKF\\_tool” on page 801](#) to view the transient database BES tables.

## check\_page

### Description

The check\_page tool provides diagnostic information about an image. The tool checks the validity of a FileNet page format that exists in a file. The page must already exist as a file in the file system—check\_page does not copy the page from cache.

If a corrupted image is tiled, check\_page provides only limited information. More detailed information is available for corrupted banded images. If the tiled or banded image is uncorrupted, check\_page displays limited information about the image and a message indicating the image is “OK.”

---

**Note** TIFF images cannot be validated by check\_page.

---

### Use

Before you invoke check\_page, you must use the CSM\_tool object-tofile command to convert the image in cache to a file. Then run check\_page to determine if the image is corrupted or to examine an image that you know is corrupted.

## Syntax

check\_page [options] <filename>

**<filename>** Full path name of the file containing the converted image to be checked

### Options

- v** Displays more (verbose) information
- V** Displays the most detailed information
- d** Dumps the contents of the file specified by <filename>
- D** Dumps everything
- q** Does not display progress messages (quiet mode)

## Example

In the first example below, the `dump` option displays information about the banded image contained in the file `/tmp/chkpgsmp1`. The second example uses the `verbose` option to display more information about the image. In both cases, the information provided is minimal since the image is a good (uncorrupted) image.

```
corona(root)/> check_page -d /tmp/chkpgsmp1
===== /tmp/chkpgsmp1 =====
page_type [FN_IMAGE_PAGE_TYPE]
OK /tmp/chkpgsmp1
corona(root)/>

corona(root)/> check_page -v /tmp/chkpgsmp1
===== /tmp/chkpgsmp1 =====
Got page header
Got band list
OK /tmp/chkpgsmp1
corona(root)/>
```

## Checklist

Before you use `check_page`, be aware of the following:

- You should use the `CSM_tool objecttofile` command to convert the image to a file before you run `check_page`.
- Information about a tiled image is limited.
- If the image is not corrupted (that is, a “good” image), `check_page` provides limited information.
- TIFF images cannot be validated by `check_page`.

## Procedure

- 1 Convert the image to a file.
  - a Locate the image.
  - b Use the CSM\_tool listobjects command to retrieve the image information (such as file name, cache identifier, system serial number, object identifier, and page number) that CSM\_tool objecttofile needs.
  - c Run CSM\_tool objecttofile.

- 2 Save the converted image.

When conversion is complete, save the converted image as a file, specifying a name of your choice, in a directory of your choice as shown in the example below from a UNIX system:

```
/tmp/chkpgsmp1
```

- 3 Run check\_page against the file you created in Step 2.

See [“Example” on page 156](#).

## Related Topics

[“CSM\\_tool” on page 204](#)

# checkwrt

## Description

Each pending write request should have a corresponding page in cache. The checkwrt tool compares pending write requests against the images in cache and verifies that each page in the cache has an associated pending write request. If they do, checkwrt displays the message “All write requests have pages in cache.” However, if a write request exists with no corresponding image in the cache, checkwrt displays a discrepancy report.

## Use

Although rarely used, checkwrt can verify that each page in cache corresponds to a pending write request.

---

**Note** The checkwrt tool is used primarily as a diagnostic tool for your service representative. The function provided by checkwrt is more fully implemented in the WRT\_clean tool. (See [“WRT\\_clean” on page 1341](#).)

---

## Syntax

```
checkwrt
```

## Sample Output

In the example below, the output of checkwrt indicates that all pending write requests have a corresponding image in the cache.

```
corona(root)/> checkwrt  
All write requests have pages in cache.
```

## Checklist

Before you use checkwrt, be aware of the following:

- The preferred tool to perform the cache page-to-pending write request correlation is the WRT\_clean tool.
- You must run checkwrt on the Storage Library server.

## Procedure

No specific procedure is required.

## Related Topics

[“WRT\\_clean” on page 1341](#)

# compressdir

## Description

**HPUX**

The compressdir tool is available only on an Image Services for AIX/6000 or Image Services for HP-UX server.

**AIX**

It does not run on Image Services for the Solaris Operating Environment or Image Services for Windows Server.

The compressdir tool recursively descends each specified directory and calls the compress program to compress each file encountered. The compressdir tool passes flag values as compress options.

If a file is very small, compressdir does not compress the file and the file extension remains the same. However, if the file is large enough to compress, compressdir calls the compress program to compress the data and replace the file with a file having a .Z extension.

You can force compression of very small files by specifying the compressdir force flag.

## Use

Use compressdir to reclaim space by reducing the size of the files within the directory.

**CAUTION**

---

If you do not specify a directory, compression is applied to **all** files starting with the **current directory**.

---

Restore compressed files to their original form by running the uncompressdir tool against the affected directories. See [“uncompressdir” on page 1294](#).



## Syntax

compressdir [ flags ] [ directory ... ]

- flags** See [“Flags” on page 161](#).
- directory** One or more directories that contain the files to be compressed. If you do not specify a directory, **all files starting with the current directory** are compressed.

## Flags

The following flags are passed to the compress program as options:

- b<maxbits>** Number of bits to use in determining the bit codes for substring compression
- <maxbits> value must be between 9 and 16 (default is 16)
- c** Concatenate option causes the compress program to write to the standard output without compressing any files
- f** Force compression of small files that are not normally compressed
- q** Do not display progress messages (quiet mode)
- v** Display progress messages that indicate the percent reduction of each file being compressed
- V** Print the version and all selected options

## Example

The following command compresses all files in the /tmp/dumps directory:

```
corona(root)/> compressdir /tmp/dumps
```

## Checklist

Before you use compressdir, be aware of the following:

- Very small files are not compressed. However, you can specify the -f flag to force compression of small files.
- The compressdir tool is not supported in Image Services for Windows Server or Image Services for the Solaris Operating Environment.

## Procedure

No specific procedure is required.

## Related Topics

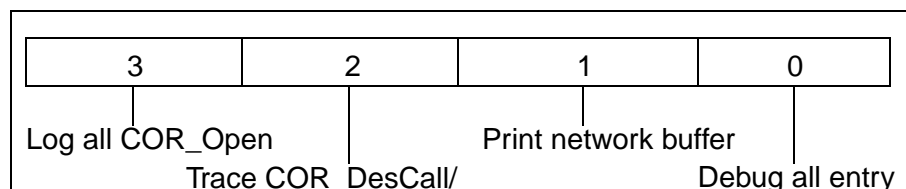
**[“uncompressdir” on page 1294](#)**

# cordebug

## Description

The cordebug tool displays COR handler request information based on your specification of the <debuglevel> option. The value of <debuglevel> is a bit mask that sets one or more flags. Simply combine the options into a decimal number that represents the functions you want.

In the following illustration, bit position number 0 corresponds to the low order bit in the mask, position 1 is the second to the lowest, and so on.



Bit Position Mapping

Bit Position	Description
0	Debug all entry points (stderr)
1	Print network buffer contents (stderr)
2	Trace COR_DesCall/COR_Delete (sys_log)
3	Log all COR_Open failures (sys_log)

Hexadecimal value 0x00000010 sets the correct bit (position 1) to print network buffer contents. The corresponding decimal debug value is 2. Hexadecimal value 0x00000011 sets the correct bit positions (0 and 1)

to print network buffer contents and debug all entry points. The corresponding decimal debug value is 3.

## Use

Only when explicitly instructed to do so by your service representative, use cordebug to diagnose problems in the COR handler. Running cordebug causes **severe** performance degradation.

## Syntax

```
cordebug <debuglevel>
```

**<debuglevel>** A bit mask that sets one or more flags. Bit position 0 corresponds to the low order bit in the mask, position 1 is the second to the lowest, and so on.

The flags set by the mask are as follows:

- 0      Debug all entry points
- 1      Print network buffer contents
- 2      Trace COR\_DesCall/Cor\_Delete
- 3      Sys\_log COR\_Open failures

## Example

```
corona(root)/> cordebug 1

***WARNING***

DO NOT USE UNLESS YOU ARE AN EXPERT
SEVERE PERFORMANCE DEGRADATION WILL OCCUR

Do you want to continue [y/n]? y
COR Debug mode is ON (debug value = 0x00000001)
```

## Checklist

Before you use cordebug, be aware of the following:

- You should use cordebug **only** when your service representative instructs you to do so.
- Running cordebug causes severe performance degradation. When invoked, cordebug issues a message warning of performance degradation. You must respond to a confirmation prompt to continue, or terminate the program.

## Procedure

No specific procedure is required.

## Related Topics

[“cormon” on page 166](#)

# cormon

## Description

UNIXWIN

The cormon tool is available on both UNIX-based and Windows Server-based Image Services servers. Note that on Windows servers, your DOS window might turn purple.

The cormon tool displays the Courier (COR) handle table. Each row in the table represents a client or server connection and its current state (CORH state). The RPC Time value represents the seconds since the last Courier call message was received on the server side or the seconds since the last call was sent by the client application with the IssueMsg call.

The ChIdPID column represents the OCOR\_Listen process that is paired with a request handler for that connection. Client connections have no associated ChIdPID; this condition is indicated by zero (0) in the ChIdPID column.

The cormon tool displays handles according to their slot in the COR handle table. A blank line represents a free handle entry. Processes are randomly assigned to the table and could fill empty slots.

## Use

The cormon program produces a dynamic display of the COR handle table. Use cormon to monitor the current state of the connections by clients or servers on the FileNet system.

Because cormon displays only as many handles as your output screen permits, at any given time, some handles are not visible. To view more handles, press the space bar on the keyboard to page forward in the

display. To obtain a complete listing, run `cordebug`, redirect the output to a file, and use a text editor to view the file contents.

## Syntax

`cormon -p`

- p** Prints the Courier handle table to the standard output device.

To save a snapshot of activity to a file, use this option with the redirect command. For example, on a UNIX system, the following command saves the output to a file named `cormon.out` in the `/tmp` directory:

**`cormon -p > /tmp/cormon.out`**

## Commands

Use the following commands at the `CMD (?-help)>` prompt to display different parts of the Courier handle table:

**s**

Delay screen update for **n** seconds. The program prompts you to enter a value for **n**.

**p | f**

Display next page (forward). Enter **p** or **f** or press the space bar on your keyboard to advance to the next page.

**b | -**

Display previous page (backward). Enter **b** or **-** to return to the previous page.

**q | x**

Terminate cormon. Enter **q** or **x**.

**j**

Move forward one line.

**k**

Move backward one line.

**h | ?**

Display help information. Enter **h** or **?**.

## Sample Output

If you enter **cormon** with no options at the command line, a dynamic display of activity, similar to the following, appears. Press the space bar on the keyboard to cycle through the slots in 20-slot increments. Enter command selections at the `CMD (?-help)>` prompt:

```
cormon      Courier Handle Display  Slots 1 - 19      Fri Jun  6 13:16:31
1997

CORH_state  Srvr  PID  ChldPID  Prog  Time  LatestUser@Address
-----
      CRAPC X   39736  41783  PRIs      0  SysAdmin@135.0.75.58 [11238]
      CRAPC X   20811  39498  PRIs      0  SysAdmin@135.0.75.58 [35695]
      DESER X   10961  21458  BESS      0  Roy Timothy@135.16.12.23 [8401]
      RCV   X   21507  35266  INXs    1345  Jane Smith@127.61.22.45 [1278]

cmd (?-help)>
```



You could enter options with the cormon command, as shown in the following example:

```
corona(root)/> cormon -p
```

CORH_state	Srvr	PID	ChldPID	Prog	Time	LatestUser@Address
RCVMSG	X	19726	9542	INXs	1187	SysAdmin@172.16.12.45 [11238]
RCVMSG	X	25680	25806	WQSS	3923	SysAdmin@172.16.12.46 [3621]
DESER	X	21588	22650	DOCs	123	Roy Timothy@172.16.12.46 [1406]
RCVMSG	X	20749	23886	INXs	7182	SysAdmin@172.16.12.44 [1043]
DESER	X	19954	22670	WQSS	2	SysAdmin@172.16.12.46 [1419]
RCVMSG	X	21313	14512	WQSS	6614	SysAdmin@172.16.12.46 [3845]
RCV	X	21258	1548	INXs	232	Jane Smith@172.16.12.45 [7248]

The following paragraphs describe each output field of the report.

### **CORH state**

Current state of a particular network connection. (See [“CORH States” on page 170](#) for a description of each CORH state.)

### **Srvr**

Connection type (server side or client side). An X in the column denotes a server connection; a blank denotes a client connection.

### **PID**

Process identifier of either the request handler (server) or a client program.

### ChIdPID

Process identifier of the OCOR\_Listen process handler that is linked to the request handler. The value in this field is 0 for clients.

### Prog

Name of the request handler for the connection. For clients, the name represents the request handler that the client is communicating with on the remote machine.

### Time

Number of seconds since a Remote Procedure Call (RPC) was transmitted or received. On the client side, the Time is reset each time an RPC is transmitted. On the server side, the time is reset each time an RPC is received.

### LatestUser@Address

Name of user that logged on to the Image Services server from the client and the network address of the client workstation or PC.

### CORH States

The following table describes the COR handler states. An S indicates a server-only state, C a client-only state, and \* either a server or client state. **Serialized** refers to network packets being sent; **deserialized** refers to packets being received.

CORH State	Designator	State Description
ATTNMSG	S	Send an out-of-band attention on the network.
BLKABRT	*	Abort bulk data transfer

CORH State	Designator	State Description
BLKGET	*	Receiving bulk data
BLKPUT	*	Sending bulk data
CHLDAL	S	OCOR_Listen child received an ALARM signal
CHLDEX	S	OCOR_Listen child has exited
CLOSE	C	Close the connection
CLOSEMSG	S	Close the connection
CONN	C	Received an open reply
CONNE	C	Error occurred waiting for an open reply
CONNW	C	Wait for the open reply
CRAPC	S	Request handler is getting the connection
CRPIPE	S	Create a named pipe
DABORT	C	Deserialize a Courier Abort message
DBYTES	*	Deserialize bytes from the Courier network buffer
DCALL	S	Deserialize a Courier Call message
DCSS	*	Deserialize a sequence from the Courier network buffer
DELETE	S	Delete the handle and close the connection
DELMSG	S	Delete the handle and close the connection
DESER	*	Internal Courier deserialization
DLWORD	*	Deserialize a long word from the Courier network buffer
DLWORDS	*	Deserialize long words from the Courier network buffer
DMOVE	*	Move the Courier network buffer pointer
DREJ	C	Deserialize a Courier Reject message
DRET	C	Deserialize a Courier Return message
DSTRUCT	*	Deserialize a structure from the Courier network buffer
DWORD	*	Deserialize a word from the Courier network buffer

CORH State	Designator	State Description
DWORDS	*	Deserialize words into the Courier network buffer
FLUSH	C	Flush the Courier network buffer
FLUSHMSG	S	Flush the Courier network buffer
GETPPM	S	Looking for an available request handler
GETRPC	S	Blocked, waiting to get an RPC over the network
IDLEMSG	S	Obsolete - no longer used
INITH	S	Initialize handle
ISSUE	*	Send a message on the network
NOSTATE	*	Unknown; probably a connection being established
OPEN	C	Establish a connection
PEEK	*	Check network for out-of-band attention
PEEKMSG	S	Check network for out-of-band attention
QUEUED	S	Connection is queued
RCV	*	Blocked, waiting for network data
RCVFD	S	Request handler is receiving the connection file descriptor
RCVFDE	S	Error occurred while getting the connection file descriptor
RCVFDN	S	Notify sender of connection file descriptor
RCVMSG	S	Blocked, waiting for network data
RCVTO	*	Same as RCV with a timeout
RELMSG	S	Connection termination sequence has started
RXATTNMSG	S	Out-of-band attention has been received
RXDATAMSG	S	Network data has been received
SABORT	S	Serialize a Courier Abort message
SBYTES	*	Serialize bytes into the Courier network buffer
SCALL	C	Serialize a Courier Call message

CORH State	Designator	State Description
SERIAL	*	Internal Courier serialization
SLWORD	*	Serialize a long word into the Courier network buffer
SLWORDS	*	Serialize long words into the Courier network buffer
SNDATT	*	Send an out-of-band attention on the network
SNDBLKMSG	S	Sending bulk data
SNDFFD	S	Send a connection file descriptor
SNDFFDW	S	Wait for completion of the SNDFFD state
SNDMSG	S	Send a message on the network
SREJ	S	Serialize a Courier Reject message
SRET	S	Serialize a Courier Return message
SSTRNG	*	Serialize a string into the Courier network buffer
SSTRUCT	*	Serialize a structure from the Courier network buffer
SVER	*	Serialize the Courier version into the Courier network buffer
SWORD	*	Serialize a word into the Courier network buffer
SWORDS	*	Serialize words into the Courier network buffer
TXDATAMSG	S	Network data has been sent
TXEXDATA- MSG	S	Out-of-band attention has been sent
WREQH	S	OCOR_Listen child is waiting for the request handler
ZOMBIEMSG	S	Connection is in an unstable state

## Checklist

Before you use cormon, be aware of the following:

- cormon dynamically monitors system activity.
- cormon displays as many handles as there are lines on your screen. During times of intensive system activity, some handles are not visible in the display.
- You can create a snapshot of the cormon table by running cordebug and redirecting the output to a file.
- cormon displays handles according to their slot in the COR handle table. Blank lines represent free handle entries. Processes are randomly assigned to the table and could fill empty slots.
- If the client uses TCP/IP protocol, the address is an Internet address in the following format:

135.0.75.11[1031]

## Procedure

- 1 Enter the following command to start cormon:

**cormon**

The CMD (?-help)> prompt displays at the end of the output.

- 2 At the CMD (?-help)> prompt, enter commands to display desired parts of the COR handle table.

## Related Topics

[\*\*“cordebug” on page 163\*\*](#)

# CPT\_test

## Description

UNIXWIN

The CPT\_test tool is available on both UNIX-based and Windows Server-based Image Services servers.

The Courier Performance Testing tool (CPT\_test) presents Courier timing measurements. CPT\_test uses COR bulk data transfers across a network to a specified host (the sever that the data is sent to or received from) while taking timing measurements. CPT\_test presents the following timing measurements:

- Transfer rate in kilobytes per second (KB/sec)
- Transfer times in milliseconds (with and without connection establishment time)
- CPU time in milliseconds per kilobyte (cpu-msec/KB) spent during the transfer

CPT\_test has four test types. Each test consists of two components: CPT\_test, the client program, and CPTs, the server stub. CPT\_test is your interface to specify test parameters. It initiates the data transfers and makes timing measurements. CPT\_test starts CPTs, which receives incoming data, sends data, and makes timing measurements.

## Transfer Direction

Transfer direction is determined from the client perspective. Sending data is a transfer of data from the client to the server. Receiving data is a transfer of data from the server to the client. Some platforms perform better in one direction than the other. The CPT\_test automatic test type tests data transfer in both directions. Your service representative can examine the results of the automatic test for indications of implementation or configuration problems that could affect the transfer rate in one direction or the other.

## Transfer Size

Transfer size is the number of bytes you wish to send for each data transfer during a test. You could want to measure throughput rates for a particular image size or try to obtain the best throughput possible by making the transfer size very large. See [“Test Type Descriptions” on page 177](#) for specific transfer size recommendations for a particular test.

## Transfer Delay

The transfer delay is a period of time you can specify in your tests that CPT\_test waits between iterations of the test. You can specify the delay as a number of centiseconds (1/100ths of a second) or as an exponential delay with a specified mean. The ability to set a transfer delay in centiseconds is useful if you need to emulate a uniform distribution of workload or an exponential delay to emulate a Poisson distribution of workload. You could also want to set a transfer delay to prevent depletion of system resources when you are running many iterations in an environment in which connections are being made very rapidly.



### Infinite Iteration Mode

You can run iterations of CPT\_test forever or until you issue a command key sequence (Control+c) to stop the tests. This infinite iteration mode is useful when performing online debugging of a network problem or to see if resources run low.

### Test Type Descriptions

The four CPT\_test test types are:

- Throughput Test

The throughput test (THR) reports the rate of transferring data to and from the specified host and the CPU time spent during the transfer. Both data transfer and CPU time are measured from the server side. Specifically, the throughput test times the bulk data transfer call. Multiple iterations of the throughput test produce a report consisting of one line of transfer rate and CPU time information per iteration.

Transfer size affects the accuracy of the test output. A transfer size that is too small creates a large variance in the stated transfer rates and CPU times because transfer time is small enough to cause a significant increase in the time spent collecting statistics. For the most accurate report, specify the largest transfer size you can (at least one megabyte) without causing a long wait for the transfer to take place.

- RPC Latency Test

The latency test (LAT) reports the time spent transferring data. This test is measured from the client side and does not include the time required to establish the connection. Specifically, the latency test includes the request serialization, the bulk data transfer, and the return deserialization. Multiple iterations of the test produce a report which has **one** output line containing the average time over the number of iterations. The more iterations performed, the more accurate the reported time.

You can measure the time spent sending a null RPC (an RPC that contains no data) by setting the transfer size to zero (0) bytes.

- RPC Latency with Connection Test

The latency with connection test (LWC) reports the time spent transferring data. This test is measured from the client side and includes the time required to establish the connection. Specifically, the latency with connection test includes connection establishment, request serialization, the bulk data transfer, the return deserialization, and connection closure. Multiple iterations of the test produce a report which has **one** output line containing the average time over the number of iterations. The more iterations performed, the more accurate the reported time.

You can measure the time spent sending a null RPC (an RPC that contains no data) by setting the transfer size to zero (0) bytes. From the results of this test, you can also measure the time spent establishing a connection.

- Automatic Test

The automatic test runs a series of each of the other three CPT\_test types, varying the transfer direction, transfer size, and packet size. The automatic test provides the easiest way to collect a range of data points.

## Use

Use CPT\_test to determine the network state and your platform's efficiency in handling network data transfers. This requires assistance from your service representative to interpret the output of CPT\_test.

CPT\_test is typically run to a remote host so that the transfer is across the network. However, to obtain information about loopback performance, you can run CPT\_test from the same server on which CPTs is running.

If you have a multi-server system and want to run CPT\_test to one of the servers, select a service that resides on that server by specifying the server's internet address, host name, or NCH name, and include the service, domain, and organization name.

## Syntax

```
CPT_test -t<test type> [-c#] [-b#] [-r] [-s] [-d#] [-R] [-i] [-n#] \  
[-h<hostaddress | NCH name>[,<port number>]] [-e[#]]
```

**-t<test type>** Run the test specified by <test type> where <test type> is one of the following:

- a** Automatic. If test type is automatic and you specify neither send (-s) nor receive (-r), the test is set to use both send and receive.
- t** Throughput test
- l** RPC latency test
- c** RPC latency (with connection) test

**-c#** Maximum amount of data, in bytes, to send in a Courier message at one time  
The value of # can be any value in the range of 512 to 102400 bytes. Default is 1458 bytes.

---

**Tip** The larger the chunk size, the more efficient the data transmission. Multiples of packet sizes are usually specified, but that is not required.

---

**-b#** Transfer the number of bytes specified by #  
The value of # must be greater than or equal to zero (0).

**-d#** Sets the delay to the number of centiseconds specified by #  
The value of # must be greater than or equal to zero (0).

- R** Exponential random delay  
The mean exponential random delay is the number of centiseconds set in -d#.
- r** Set up the test to receive data  
Receive is the default if neither receive nor send is specified. However, if <test type> is automatic, and neither -r nor -s is specified, the test is set to both receive and send.
- s** Set up the test to send data  
If <test type> is automatic, and neither -s nor -r is specified, the test is set to both send and receive.
- n#** Perform the number of iterations specified by #  
The value of # must be greater than or equal to 1.
- i** Perform an infinite number of iterations

---

**Tip** Use Control+c key sequence to terminate infinite iteration mode.

---

- h<hostaddress | NCHname> [,<portnumber>]** RPC destination specified as a host address or NCH name  
<hostaddress> can be specified in one of the following formats:  
<NCHname> can be specified in one of following three formats:
- |                             |   |
|-----------------------------|---|
| domain                      | Example: -hIndexServer                  |
| domain:organization         | Example: -hIndexServer:magic            |
| domain:organization:company | Example:<br>-hIndexServer:magic:FileNet |
- e[#]** Number (#) of Courier errors that must occur before CPT\_test exits

## Test Output Format

The first two columns of test output identify the test type (THR for throughput, LAT for RPC latency, LWC for RPC latency with connection) and the send/receive mode (S means client sends the data; R means client receives the data). The remainder of the output differs between the throughput test and the two RPC latency tests, as described in the following table:

Column	Throughput Test	RPC Latency Tests
Pkt	Chunk size in bytes	Chunk size in bytes
Wnd	IO Size (set internally)	IO Size (set internally)
Bytes	Data transfer size	Data transfer size
KB/s	Data transfer rate or throughput measured in kilobytes per second	N/A
msecs	N/A	Number of milliseconds required to perform the data transfer, averaged over the number of iterations
count	N/A	Number of iterations
cpu-msecs/KB	CPU time spent during data transfer, measured in milliseconds per kilobyte	N/A
idle	Percentage of time the server was completely idle (not even waiting for IO) during a data transfer	N/A
kern	Percentage of time the server spent running kernel code during the data transfer	N/A

Column	Throughput Test	RPC Latency Tests
user	Percentage of time the server spent running user code during the data transfer	N/A
wait	Percentage of time the server spent waiting for IO devices during the data transfer	N/A

## Sample Output

The following sample shows throughput test output:

```

cpu-msecs
Test  Pkt  Wnd  Bytes  KB/s      /KB  idle  kern  user  wait
THR  R 1024 0  1048576 17476.27  0.06  0%   83%  17%  0%
THR  R 1024 0  1048576 17476.27  0.06  0%   67%  33%  0%
THR  R 1024 0  1048576 17476.27  0.06  0%   50%  50%  0%
THR  R 4096 0  1048576 17476.27  0.06  0%   83%  17%  0%
THR  R 4096 0  1048576 17476.27  0.06  0%   83%  17%  0%
THR  R 4096 0  1048576 17476.27  0.06  0%   83%  17%  0%
THR  R 1024 0  1048576 2139.95   0.47  0%   18%  82%  0%
THR  R 1024 0  1048576 2056.03   0.49  0%   14%  86%  0%
THR  R 1024 0  1048576 1839.61   0.49  11%  28%  61%  0%
THR  R 4096 0  1048576 6168.09   0.16  0%   41%  59%  0%
THR  R 4096 0  1048576 5825.42   0.14  17%  17%  67%  0%
THR  R 4096 0  1048576 5518.82   0.15  16%  21%  63%  0%

```

The following sample shows latency test output:

Test	Pkt	Wnd	Bytes	msecs	count
LAT R	16384	0	0	1	100
LAT R	16384	0	25000	3	75
LAT R	16384	0	50000	4	50
LAT R	16384	0	75000	6	25
LAT R	16384	0	100000	7	10
LAT S	16384	0	0	1	100
LAT S	16384	0	25000	3	75
LAT S	16384	0	50000	7	50
LAT S	16384	0	75000	7	25
LAT S	16384	0	100000	9	10
LWC R	16384	0	0	25	100
LWC R	16384	0	25000	18	75
LWC R	16384	0	50000	15	50
LWC R	16384	0	75000	16	25
LWC R	16384	0	100000	19	10
LWC S	16384	0	0	11	100
LWC S	16384	0	25000	13	75
LWC S	16384	0	50000	15	50
LWC S	16384	0	75000	16	25
LWC S	16384	0	100000	18	10



## Checklist

Before you use CPT\_test, be aware of the following:

- The server stub, /fnswh/bin/CPTs, on the server you specify in the -h option must have permission to read /dev/kmem.
- If CPT\_test reveals a problem, you could need to collect local area network (lan) traces to further investigate the problem.

## Procedure

No specific procedure is required.

## cranuser

### Description

The cranuser tool's sole purpose is supporting ISRA/LDAP customers with anonymous logins. When ISRA is configured with LDAP authentication, the user may be logged in as an 'anonymous' user. If this is the case, all subsequent IS logins from ISRA will be identified as "LDAP/ISRA\_ANONYMOUS\_USER". This tool is for ISRA/LDAP customers only.

### Use

This tool, located in the /fns/bin directory, generates the Image Services user **LDAP/ISRA\_ANONYMOUS\_USER** on the specified IS domain and its associated hashed password in the IS security database. This user has minimal IS privileges and is not initially associated with any IS security groups. The System Administrator should properly adjust the **maximum logons** property.

Normally it is only possible to add an LDAP user into the IS security database through the LDAP\_import tool that reads directly from an imported LDAP XML file (see [“LDAP\\_import” on page 671](#)). This tool circumvents that requirement for ISRA/LDAP customers.

## Syntax

```
cranuser /h<host>[:organization] /s<server_name>  
/a<ip_address> [ /u<id> /p"<pwd>" ]
```

? Help. This will bring up the help screen.

**h<host>[:organization]** Image Services host or domain name, organization name (the organization name is optional)

**s<server name>** LDAP server where the anonymous user resides

**a<ip address>** IP address of the server specified by the **s** option

**u<id>** Authorized IS user name (use with the **p** option)

**p"<pwd>"** Encrypted IS user password (use with the **u** option)

Be aware that you need to create this encrypted password using the LDAP\_password tool and enter the encrypted password with the **p** option (see [“LDAP\\_password” on page 690](#)).

---

**Note** Options are **not** case sensitive. The command switches must be prefixed by a '-' on UNIX platforms and "/" on Windows Server platforms.

---

## Example

In the example below, the **h** option with the IS Host name (idm1) is entered, as well as the **s** option with the LDAP server name and the **a** option with the IP address of the LDAP server. The LDAP/ISRA\_

ANONYMOUS\_USER is then generated on the host after the user name and password prompts are answered.

```
J:\3.6.10\user\sec\nt>cranuser /hidm1 /sldaphost /a10.55.14.25
Locating IS server...please wait
Authorized IS user name: SysAdmin
Authorized IS user password:
User LDAP/ISRA_ANONYMOUS_USER has been added.
J:\3.6.10\user\sec\nt>
```

## Procedures

### Example 1

Create anonymous ISRA LDAP user on an Image Services system:

```
cranuser /hidm1 /sldaphost /a10.55.14.25
```

All information or error messages are logged to system log file or **cranuseryyyymmddlog.txt** under the LDAP directory of the common FileNet logging directory.

### Example 2

Create anonymous ISRA LDAP user on an Image Services system with a specified user name and a specified encrypted IS user password (obtained via LDAP\_password):

```
cranuser /hidm1 /sldaphost /a10.55.14.25 /filenet /p"sr8#%ke"
```

### Note

On Windows systems, if the encrypted password contains a % or a \, you need to add a second % or \. For example, **/p":015K+]%"** needs

to be entered as `/p":015K+]%%"`. A % becomes %% and a \ becomes \\. 

---

## Related Topics

["ldap\\_exp" on page 662](#)

["LDAP\\_import" on page 671](#)

["LDAP\\_password" on page 690](#)

# CSM\_exim

## Description

The CSM\_exim tool exports and imports objects in the cache. Objects are usually pages of a scanned document that are ready to be indexed or committed to storage media. When you export objects, CSM\_exim saves the objects on tape or magnetic media. When you import objects, CSM\_exim restores the objects from tape or magnetic media files to the current cache.

During import, CSM\_exim extracts selected objects from the named file, loads the retrieved objects into cache, and creates a report file.

During export, CSM\_exim locates the objects in cache and creates a set of files to receive the data. The files created by CSM\_exim during export are:

- Directory of selected objects

The directory file includes the cache ID, object ssn, object ID, and page ID of each selected cache object.

- One or more files containing the selected objects
- Report file

After creating the set of files, CSM\_exim writes the selected objects to tape or magnetic media.

The name of the CSM\_exim report file is always **csm\_ei\_rpt.<pid>** where pid is the identifier of the CSM\_exim process that created the file. However, the file names of the directory file and the receiving file differ depending on the external data location (magnetic disk or magnetic tape) you specify in your CSM\_exim command.

CSM\_exim requires ten bytes of memory for each object to be exported. If the number of cache objects to be exported is very large (for example, millions of objects), CSM\_exim could fail with an out-of-memory error during export attempts. In this case, consider using the FileNet Cache Export/Import Program, which does not have this memory constraint.

Since the CSM\_exim tool does not synchronize the cache with the transient database, it is not the best tool to use for backing up cache to restore for disaster recovery on the same machine.

If you want to back up your cache to store for disaster recovery on the same Image Services server, you should use the FileNet Enterprise Backup and Restore (EBR) program instead. EBR synchronizes cache with the transient database, thus ensuring the restored objects have the same IDs they had when you performed cache backup. (For details on using EBR, see the *Enterprise Backup/Restore User's Guide*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#))

## Use

Use CSM\_exim to export or import cache objects. You should use this tool to export locked objects from one Image Services server and import them on a different server.

During an export or import operation, CSM\_exim automatically displays information about the objects being exported or imported. You can optionally suppress the display of this information.

Use the `-d<storage spec>` option to specify the external location (the local tape drive or a magnetic disk) of the data to be exported or imported. See [“Options” on page 193](#) for details of the `-d` option.

If you use the `-d` option to specify a file name prefix, CSM\_exim uses the prefix to build its file names. If you do not specify a file name prefix with the `-d` option or you specify `-d tape`, CSM\_exim creates a default file prefix of `CSM_EI_DAT` for each of its automatically-created files.

For example, in the following table, the commands shown in the left column result in files similar to those shown in the right column (actual files would reflect the cache identifier and process identifier):

Export Command	Automatically-Created Files
CSM_exim -e -d ABC	ABC.DIR ABC.<cache_id> csm_ei_rpt.<pid>
CSM_exim -e -d tape	CSM_EI_DAT.DIR CSM_EI_DAT.<cache_id> csm_ei_rpt.<pid>
CSM_exim -e	CSM_EI_DAT.DIR CSM_EI_DAT.<cache_id> csm_ei_rpt.<pid>



## Syntax

CSM\_exim -e | -i <options> <selections>

- e Exports the specified caches or cache objects to the specified magnetic media (disk file or tape). Mutually exclusive with -i.
- i Imports the specified caches or cache objects from the specified file to the current cache. Mutually exclusive with -e.

## Options

Valid options are:

- r Replaces existing objects on import. If cache objects in the import file have the same name as those currently in cache, the objects in the import file replace those in cache. CSM\_exim does not replace cache objects with the same name unless you specify -r.
- n Displays selections, such as cache and object IDs from the named file, and cache information. It does not export or import cache. Use this option only to obtain information.
- k Retains checksum of the object, if any, even if the system is not configured for checksum. This option is applicable only to import operations.
- v Verify mode displays the location of cache objects (in the format **cachename:domain:organization**) during the export process. This option is applicable only to export operations.
- q Turns off the display of cache object attributes (cache ID, ssn, object ID, page) during the export or import process.

**-l** Exports locked objects only

**-d<storage spec>**

Specifies the external location of the data. If not specified, the default storage medium is magnetic disk and the file name prefix defaults to CSM\_EI\_DAT.

If the data is to be imported from or exported to magnetic disk, <storage spec> is the file name prefix of each file in a set of files containing the data.

To export to or import from magnetic tape, specify the word **tape** to use the default tape drive or specify the Network Clearinghouse (NCH) name of a tape drive. You can specify the NCH name in one of the following formats:

```
tape_drive_name:domain_name:organization_name  
tape_drive_name:domain_name  
tape_drive_name:
```

If you do not specify the domain and organization, CSM\_exim uses the default domain and organization.

---

**Note** If a specified tape drive is not local to the server, CSM\_exim produces an error message.

If you use the -d tape option, be aware that CSM\_exim uses the local tape drive, if one exists, and does not use the default tape drive.

---

The names of files automatically created by CSM\_exim differ depending on the specification of the -d option. See [\*\*“Use” on page 192\*\*](#) for more information.

**-t<time\_spec>**

Exports only those objects with a creation time stamp or the last update time stamp later than the specified time. The specified time, <time spec>, must be in the format “**mm/dd/yyyy hh:mm:ss am | pm**”.  
(Requires double quotes)

---

**Note** The format “am | pm” means you must select a designation of either am for morning hours or pm for afternoon hours.

---

## Selections

You can use selections to import objects to the same or a different cache and object ID. For example, the following command imports all `page_cache1` objects found on the tape into `page_cache2` on magnetic disk and all `page_cache3` objects found on the tape into `page_cache3` on magnetic disk:

```
CSM_exim -i -d tape -c page_cache1,page_cache2  
-c page_cache3
```

Valid selections are:

```
-c <cache name>[,<cache name>] [ -o <objects>[,<objects>]]
```

Name of the cache, and optionally the objects, that you want to export or import

where

**<cache name>** is one of the following formats:

`cache_name:domain:organization`

`cache_name:domain` in the default organization

`cache_name` in the default domain and organization

If you do not specify domain and organization, CSM\_exim uses the default domain and organization.

**<objects>** is one specified in the following table:

Object Specification	Description	Example
'ssn id page'	Selects all objects in <cache name> with system serial number = 'ssn', object ID = id, and page number = page	'10000 101019 1'
'ssn id *'	Selects all objects in <cache name> with system serial number = 'ssn' and object ID = id	'10000 101019 *'
'ssn *'	Selects all objects in <cache name> with ssn = 'ssn'	'10000 *'
'*'	Selects all objects in <cache name>. This is the default.	'*'

---

**Note** Specifying <objects> is optional, but if specified, you must enclose them in single quotes.

---

#### **-f <filename>**

Specifies the name of the file that contains the caches and/or cache objects to be selected for import or export

The file is line-oriented. Precede cache specifications by a line containing the word "cache" and object specifications by a line containing the word "object." CSM\_exim ignores leading blanks and tab characters and the pair of quotes enclosing an object specification is optional. Place each cache or object specification on a separate line. Add comments by inserting a # sign as the first character of the text line.

As an example, the following command,

```
CSM_exim -c cache1 -o '123 23456 1' -o '123 45678 *'  
-c cache2 -o '*'
```

can be specified as

**-f sel\_file**

with sel\_file containing the following lines:

```
# cache specification  
cache  
    cache1  
object  
# an object specification enclosed in quotes  
    '123 23456 1'  
# another set of objects specified without quotes  
    123 45678 *  
cache  
    cache2  
# the following 2 lines can be omitted without changing  
the meaning  
object  
    '*'
```

## Sample Output

The following sample output shows an export of page\_cache1 to a labeled tape. CSM\_exim prompts the user to confirm overwriting the tape:

```
>CSM_exim -e -d tape -c page_cache1
CSM_exim: initializing
CSM_exim: initialized.
CSM_exim -e -d tape -c page_cache1
CSM_exim: expanding list of objects to be exported...
CSM_exim: creating directory of selected objects...

For CSM_exim reel 1, labeled tape requested-overwrite tape?
CSM_exim: the selected objects are:
```

<u>Cache</u>	<u>Object SSN</u>	<u>Object ID</u>	<u>Page</u>
1	10000	101019	0
	10000	101019	1

```
CSM_exim: number of objects selected from page_cache1:corona:FileNet=2
Exporting objects to CSM_EI_DAT
CSM_exim: completed with no error.
```

The following output shows an import from tape to page\_cache1:

```
>CSM_exim -i -d tape -c page_cache1
CSM_exim: initializing
CSM_exim: initialized.
CSM_exim -i -d tape -c page_cache1
Extracting selected objects from CSM_EI_DAT.DIR
CSM_exim: the objects to be imported are...
```

<u>Cache</u>	<u>Object_SSN</u>	<u>Object_ID</u>	<u>Page</u>
1	10000	101019	0
	10000	101019	1

```
CSM_exim: number of objects selected from page_cache1:corona:FileNet=2
Importing objects...
CSM_exim: completed with no error.
```



## Checklist

Before you use CSM\_exim, be aware of the following:

- You must have a valid fnlogon session before invoking CSM\_exim.
- During an import, CSM\_exim does not overwrite objects in cache with imported objects of the same name. Use the -r flag to update current cache objects with those of the same name from the imported file.
- The FileNet Cache Export/Import Program provides an alternate method of backing up (exporting) and restoring (importing) cache.
- If you want to back up your cache to store for disaster recovery on the same Image Services server, you should use the FileNet Enterprise Backup and Restore (EBR) program instead. EBR synchronizes cache with the transient database, thus ensuring the restored objects have the same IDs they had when you performed cache backup.

## Procedure

- 1 Collect information about caches and cache objects.

Use the CSM\_tool statistics command to obtain the names of caches you can export. Use the CSM\_tool listobjects command to obtain cache object names and identifier information. This information becomes input to your CSM\_exim command.

- 2 Use one of the following CSM\_exim command procedures with appropriate options and selections:

- a To export, use the -e option.

If exporting to magnetic tape, you could receive the following prompt to overwrite a labeled tape:

For CSM\_exim reel 1, labeled tape requested-overwrite tape?

If you do not want to overwrite the tape, respond **No** and eject the tape. Insert a new tape into the drive and rerun the command. Respond **Yes** to overwrite the tape and continue.

If you are exporting to magnetic disk, write down the file name associated with the saved (exported) cache. (This information is not written to the report file so you need to make a note of it.) You need the file name when importing the cache.

- b To import, use the -i option.

Use the -r flag to update existing cache objects with those from the imported file.

## Related Topics

[\*\*“CSM tool” on page 204\*\*](#)

[\*\*“fnlogon” on page 489\*\*](#)

See the “Backup” chapter of your *System Administrator’s Companion for UNIX*, *System Administrator’s Companion for Windows Server* for FileNet Cache Export/Import Program description and procedures.

See the *Enterprise Backup/Restore User’s Guide* for cache backup procedures.

To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21\*\*](#).

## CSM\_tool

### Description

CSM\_tool manipulates cache and retrieves cache statistics. You can create cache objects from data in files, delete cache objects, add and initialize cache partitions, and dump cache objects to readable files. You can also display the contents of cache objects and display information on cache names and sizes.

Progress indicators for the initialize, scavenge, and initpartition commands give you feedback on the status of the process and the number of objects processed. A sample of progress indicators for the initpartition command are shown below:

```
<CSM_tool>initpartition /fnsw/dev/1/cache0 out
Deleting cache used space in the partition. Number of objects scanned:
511 scanned (510 found in the partition and deleted).
Deleting csm_free_space. Number of rows deleted:
4
Refreshing cache used space. Number of objects refreshed:
0
Current stats shows a total of 0 objects in cache.
Scanning all cache used space. Number of objects scanned:
0
New stats determined from offline scan:
cache  inuse_objs  inuse_secs  locked_objs  locked_secs
-----
  0          0          0           0           0
  1          0          0           0           0
  2          0          0           0           0
  3          0          0           0           0
  4          0          0           0           0
Partition '/fnsw/dev/1/cache0' initialized, 510 objects deleted
<CSM_tool> q
```

## Use

Use CSM\_tool to:

- Create a specified number of cache testing objects.
- Examine document services conditions. If you cannot retrieve a document you think was committed, use CSM\_tool to verify that it has been committed.
- Provide information to your service representative. Some problems require support personnel to view the contents of cache. Use CSM\_tool to display the cache contents.
- Examine cache full and document committal problems. If you cannot commit documents, the cache could be full. Use CSM\_tool to view cache usage statistics.
- Locate your documents. Given a cache object ID, CSM\_tool can determine if the object exists in a cache.
- Scan cache objects after FileNet software has started to rebuild cache statistics.
- Analyze printing problems. Look at the contents of the print cache to locate problems.

## Syntax

CSM\_tool

After entering CSM\_tool, the program prompts:

<CSM\_tool>

Enter your CSM\_tool commands at this prompt.

---

**Note** Because some CSM\_tool commands, if improperly used, can damage or corrupt data, you must supply a password before you can use them. You can obtain the password from your service representative.

---

In the command names in this section, uppercase characters indicate the minimum number of characters of the command name that you must enter at the command line to run the command. You can always enter all characters of a command to run it but you cannot enter fewer characters than those shown in uppercase. For example, to run the delageable command (shown as DELAGEable), you must enter at least the following characters:

**delage**

The command fails if you enter **delag**, **dela**, **del**, **de**, or **d**. At the command line, you do not have to enter any character in uppercase letters. Uppercase is used only in this section to indicate required syntax.

CSM\_tool provides many commands. The following table identifies all available commands, in alphabetical order, with a brief description of each. (“**Commands**” on page 208 describes each command in detail.) An asterisk (\*) in the Indicator column identifies a repeatable command (the function is repeated when you press the carriage return (<cr> key); a plus sign (+) indicates a password-protected command (one that requires you to first use the ALLOWUPDATES command and enter a password).

Indicator	Command	Description
+	ADDpartition	Adds a partition when used in conjunction with specific configuration procedures
	CHECKSum	Displays checksum saved in database and checksum computer on the object
	CREATEobjects	Creates and populates a number of cache test objects with a specified id and size
* +	Debug	Lists lock holders
+	DELAGEable	Deletes all ageable objects in cache
+	DELETEobjects	Deletes one or more cache objects
	DUmpobject	Produces a hexadecimal dump of a cache object
+	Filetoobject	Writes a file to cache as an object
	HARdcopy	Places a copy of all output into a file
+	INITialize	Initializes cache
+	INITPartition	Initializes a partition
*	LForward	Lists objects from a specified object ID forward
*	LBackward	Lists objects from a specified object ID backward
*	Listobjects	Lists specified objects
+	LOCKdoc	Locks a document in ageable cache
	OBjecttofile	Writes a specified object to a file

---

Indicator	Command	Description
	PAging OFF	Starts paging mode for list commands
	PAging ON	Stops paging mode for list commands
	Quit	Exits CSM_tool
+	REBUILDstats	Rebuilds in-use and locked objects/sectors statistics
+	SCavenge	Rebuilds the free space chain for all caches on a server
*	Statistics	Prints cache statistics
	TERMOFF	Turns off terminal output for list commands
	TERMON	Turns on terminal output for list commands
+	TRACE ON	Starts tracing
+	TRACE OFF	Stops tracing

## Commands

This section describes each CSM\_tool command in detail.

---

**Tip** Some commands allow you to enter an abbreviation for the command. This section uses uppercase only to indicate the required minimum number of characters that you must enter at the command line to run the command. You can enter characters in uppercase letters or lowercase.

---



**ADDpartition <directory\_path><partition\_name>**

(Requires ALLOWUPDATES.)

Adds a cache partition when used in conjunction with specific configuration procedures. You should run these procedures when you want to quickly add a cache partition without destroying data stored in existing page cache.

The following examples show the procedures required to add cache partitions for the specified Image Services server platform.

**On a Windows Server Image Services server:**

```
initfnsw stop
fn_edit -> Procedures -> Add an additional dataset -> Cache
fn_build -a
fn_util updatecache
fn_util starttrans
fn_util startsec
csm_tool
<CSM_tool>allowupdates
<CSM_tool>addpart C:\FNSW\dev\1\cache1
<CSM_tool>quit
initfnsw restart
```

**On a SUN Solaris Image Services server:**

```
initfnsw stop
fn_edit -> Procedures -> Add an additional dataset -> Cache
fn_build -a
vxassist -g fndg make fn_cache1 100m
(Replace 100m with your desired cache size)
In -s /dev/vx/rdisk/fndg/fn_cache1 /fnsw/dev/1/cache1
fn_util starttrans
fn_util startsec
csm_tool
<CSM_tool>allowupdates
<CSM_tool>addpart /fnsw/dev/1/cache1
<CSM_tool>quit
initfnsw restart
```

**On an HPUX or AIX Image Services server:**

```
initfnsw stop
fn_edit -> Procedures -> Add an additional dataset -> Cache
(Use su root and set DISPLAY to run fn_dataset_config)
fn_dataset_config
fn_util starttrans
fn_util startsec
csm_tool
<CSM_tool>allowupdates
<CSM_tool>addpart /fnsw/dev/1/cache1
<CSM_tool>quit
initfnsw restart
```

## ALLOWUPDATES

Some CSM\_tool commands change the cache (initialize, scavenge, deleteobjects, filetoobject, lockdoc) and two commands are used for debugging (debug and trace). You cannot use these commands until you run ALLOWUPDATES. The ALLOWUPDATES command requires a password, which you must obtain from your service representative. Command descriptions in this section indicate whether a command requires ALLOWUPDATES.

---

**Note** When listing CSM\_tool commands, some might not appear if you have not run the ALLOWUPDATES command. To list password-protected commands, you must first run ALLOWUPDATES.

---

### Checksum <cache\_id> <ssn> <object\_id> <page>

Calculates and displays the checksum for the specified object. It also lists the checksum saved in the database for this object, if any.

The following is a sample of checksum output when the checksum is not found in the database:

```
<CSM_tool> checksum 1 1717 100100 1
Computed checksum for object is 0xfcc98d2c
No checksum was saved in database for this object
```

The following is a sample of checksum output when the checksum is found in the database:

```
<CSM_tool> checksum 1 6000 531568 1
Computed checksum for object is 0x920466bd
Checksum saved in database for this object is 0x920466bd
<CSM_tool>
```

### **CREATEobjects <cache\_id> <ssn> <start\_object\_id> [<number\_objects> [<bytes>]]**

Creates and populates a number of cache test objects with a specified id and size.

For example, the following command creates 100 cache objects of 1000 bytes each in cache ID of 1, ssn of 6000, starting with object id 200000:

```
<CSM_tool> create 1 6000 200000 100 1000
```

### **Debug [long]**

(Requires **ALLOWUPDATES**.)

Lists open objects in cache.

Specifying the long option displays the open object attributes, the critical section lock owner, and retry counts.

The following table describes each of the debug information categories:

Category	Attributes	Description
Open Object Attributes	obj_id pg ssn cache_id	Object identifier, page, system serial number, and cache identifier of the open object.
	q_rdrs q_wrtr	Number of processes waiting to read (q_rdrs) or write (q_wrtrs) the object.
	start	Hexadecimal starting address of the object in the form 0xAABBBBBB where AA is the page cache partition number and BBBBBB is the offset in the partition.
	max_length	Maximum length of the object. This value indicates the amount of space allocated for the object.
	cur_length	Current length of the object. This value indicates how much space the object has been written to.
	active_rdrs active_wrtr	Number of processes currently reading or writing the object. Several readers can be active at one time, but only one writer can be active at a time. If readers are active, no writers can be active, and if a writer is active, no readers can be active.
	closers	Number of processes in the process of closing the object. This count should be nonzero for no more than a fraction of a second.
	delete_flag	Indicates if the object is in the process of being deleted. This flag should be true for an object no more than a fraction of a second.
open_pids	Indicates which processes currently have the object open. Up to four process IDs are saved. More than four processes can have the object open, but those process IDs are not shown. If a process ID is present in the open_pids field, you can be sure that the process has the object open. However, the absence of a process ID in this field does not guarantee that a process does not have the object open.	

Category	Attributes	Description
Critical Section Lock Owner	Not applicable	Process ID that owns the critical section lock of CSM. An individual process should hold this lock no more than a fraction of a second. If a process holds this lock longer, either the process has died or a deadlock exists, either of which could be the cause of a system hang.
Retry Count Types	major busy retries	Number of attempts to obtain a lock for an object. This type of retry occurs when a process is waiting for an object that is open. Each wait for the open lock of the object lasts four seconds before the attempt is aborted and retried. Five retries (over a twenty-second period) occur before a busy error is returned to the caller.
	minor busy retries	Number of attempts to obtain a lock for an object. This type of retry occurs when a process is waiting for an object that is in the process of being opened, deleted, or closed. The process waits for 100 milliseconds before aborting and re-executing the transaction, then repeating the wait for the completed process.
	temporary busy retries	Same as a minor busy retry except that the caller is not in transaction state; therefore, a 100-millisecond wait occurs then the operation is retried. It does not abort or re-execute transactions.

### DELAGEable

(Requires **ALLOWUPDATES.**)

Deletes all ageable objects in cache. Ageable objects are documents that also exist on storage media.

**DELETEobjects [[<object-range>] | [<start\_object\_id> FOR <count>]] [desc]**

(Requires **ALLOWUPDATES**.)

Deletes a specified range of objects from cache. Before using this command, use the listobjects command with the same parameters to list the objects to be deleted.

You can start deletion with a particular object or you can delete objects within a range from a specified beginning object. You can limit the number of objects to delete by specifying a count value in the FOR phrase. You could also delete objects in ascending order (default) or descending order (by adding the **desc** option).

---

**CAUTION**

This command could corrupt your system. Do not use deleteobjects under normal system operating conditions.

---

To avoid accidental deletions, the program displays a prompt, requiring you to verify every object deletion, unless you select the 'a' option to delete all objects. For example:

```
Delete object 1:6000:200000:1 ?  
'y' / 'CR' = yes, 'a' = yes to all, 'x' / 'q' = exit, all others = no
```

As the program deletes objects, it records them in a log file (named log\_) under the following directory path:

```
/fnsw/local/logs/csmlog           for UNIX platforms  
  
<drive>:\fnsw_log\logs\csmlog     for Windows Server platforms
```

The following command deletes cache objects in cache id 1 with an ssn of 6000, having object ids within the range of 200000 and 200999:

```
<CSM_tool> delete 1 6000 200000 - 200999
```

The following command deletes the first 3 objects in cache id 1:

```
<CSM_tool> delete 1 for 3
```

### **DUmpobject <cache\_id> <ssn> <object\_id> <page>**

Displays the contents of an object in hexadecimal and ASCII. The following is a sample of dumpobject output:

```
Sector: 0  Quadrant: 0  <==>  Offset: 0x00000000
Offset      0          4          8          C          0123456789ABCDEF
00000000: 02100023 06c00898 0240021c 02400180  ...#. ....@...@..
00000010: 02400150 02400140 02400140 02400140  .@.P.@.@.@.@.@
00000020: 02400140 02400140 02400140 02400140  .@.@.@.@.@.@.@
00000030: 02400140 02400144 02400180 02400180  .@.@.@.D.@...@..
00000040: 024001d4 02400488 0240083c 0240093c  .@...@...@.<.@.<
00000050: 0240092c 024009ec 02400a44 02400a30  .@.,.@...@.D.@.0
00000060: 02400ae8 02400b4c 02400d00 02400da8  .@...@.L.@...@..
00000070: 02400d44 02400b00 02400ad8 024007bc  .@.D.@...@...@..
00000080: 02400564 0240034c 02400324 02400204  .@.d.@.L.@.$.@..
00000090: 021800b0 00135159 a3a1c0e4 06e00135  .....QY.....5
000000a0: 159b581c 835c0001 351b346c 0e406000  ..X..\..5.4l.@`.
000000b0: 01351b34 b40e4080 01351b35 2c09a0b4  .5.4..@..5.5,...
000000c0: 0001351b 48802682 a8000135 1b489813  ..5.H.&....5.H..
000000d0: 41a40001 351b485c 09a19c00 01351b49  A...5.H\.....5.I
N)ext, P)revious, S)ector offset, or B)yte offset (CR=exit): <
```



Use the prompts at the bottom of the screen to navigate through information:

- N Displays the next screen of hex/ASCII information
- P Displays the last screen of hex/ASCII information
- S Prompts you for the sector at which to continue viewing information
- B Prompts you for byte location at which to continue viewing information
- <CR> Exits dumpobject

### **Filetoobject <cache\_id> <ssn> <object\_id> <page> <filename> [ageable]**

(Requires ALLOWUPDATES.)

Writes a UNIX file to the cache of the object specified. If you specify the ageable keyword, it writes the object to cache and flags it as ageable; otherwise, the object is locked in cache. The object's FileNet security is set to ANYONE for all types of access.

### **HARdcopy [<filename>]**

Writes all CSM\_tool output to a file in addition to the display. (See [“TERMOFF | TERMON” on page 243](#) to control output to the display.)

The following command writes all CSM\_tool output to the 113096.out file:

```
<CSM_tool> hardcopy 113096.out
```

To turn off output to the 113096.out file, enter the hardcopy command again without a file name:

```
<CSM_tool> hardcopy
```

**INITialize [<cache\_id>]**

(Requires ALLOWUPDATES.)

Initializes all caches or a specified cache. You can run this command while the system is running, but once it starts, all system access is denied to all caches on the server until initialization completes.

**CAUTION**

---

Do not use the initialize command under normal system operating conditions. Doing so could corrupt your system.

---

The initialize command removes all objects from the specified caches but does not remove the references to objects in the cache that could be stored elsewhere. Therefore, do not use initialize unless all other means of cleaning up the cache have failed. Running the initialize command in the wrong situation can cause inconsistencies in the database and many other types of errors.

**Note**

---

Use **scavenge**, not initialize, if free space becomes corrupted.

---

During initialization, progress messages display:

```
<CSM_tool> initialize
Deleting csm_free_space. Number of rows deleted:
1
Deleting cache used space. Number of objects deleted:
1065 (of 1065 or 100.00%) done.
Cache(s) successfully initialized.
<CSM_tool>
```

**INITPartition** <partition\_name> <outputfilename>

(Requires ALLOWUPDATES.)

Initializes a single cache partition by deleting all objects in that partition. You should run this command after a magnetic disk failure causes data loss in a partition. Initpartition could leave the system with missing page cache objects that you must correct at a later time.

**CAUTION**

---

This command could corrupt your system. Do not use it under normal system operating conditions.

---

The file specified by <outputfilename> lists the deleted objects (by cache\_id, ssn, object\_id, and page) and indicates whether an object is ageable.

During initialization, progress messages display:

```
<CSM_tool>initpartition /fnsw/dev/1/cache0 out
Deleting cache used space in the partition. Number of objects scanned:
511 scanned (510 found in the partition and deleted).
Deleting csm_free_space. Number of rows deleted:
4
Refreshing cache used space. Number of objects refreshed:
0
Current stats shows a total of 0 objects in cache.
Scanning all cache used space. Number of objects scanned:
0
New stats determined from offline scan:
cache  inuse_objs  inuse_secs  locked_objs  locked_secs
-----  -
0          0          0          0          0
1          0          0          0          0
2          0          0          0          0
3          0          0          0          0
4          0          0          0          0
Partition '/fnsw/dev/1/cache0' initialized, 510 objects deleted
<CSM_tool>q
```

**LBackward [<objectid>] [FOR <count>]**

Lists objects from the specified object ID backward for the specified number of objects. If you do not specify an object ID, listing starts from the next available object immediately before the last object found and displayed from the previous Listobjects, LForward, or LBackward command. If no object was previously found and displayed, listing starts from the last object for the LBackward command. If you do not specify a count value with the FOR phrase, CSM\_tool lists **all** objects from the starting object backward.

Specify objectid in one of the following formats:

```
[<cache_id> [<ssn> [object_id> [<page>]]]]
```

```
<cache_id> '-' [<cache_id>]
```

If you do not specify a range value after the hyphen ('-') in the second format above, CSM\_tool assumes the maximum value of the specified object level (cache ID level, ssn level, object ID level, or page level).

See [“object range” on page 225](#) for more information on range level.

LBackward output displays in paging mode. However, you can use the **PAGING OFF** command to set output to scrolling mode. (See [“Paging \[ON | OFF\]” on page 231](#).)

You can repeat the command to display the next batch of objects by pressing the carriage return (<cr>) key. The object batch size for display is determined by the last FOR count, if one was originally specified; otherwise, listing continues to the end in either scrolling or paging mode. The LBackward command being repeated displays as the first line of the next group in the list.

A less than (<) sign indicates the list operation is in the backward direction beginning with the **next lowest** objectid from the displayed objectid, as shown below:

```
<CSM_tool> lb
(objs <      3      1717      100106      1)
cache_id      ssn      object_id      page      max_length
-----      ---      -----      ----      -----
3              1717      100105      1          10240
```

See the Listobjects command for a comprehensive example of listing cache objects with the Listobjects, LForward, and LBackward commands.

### LForward [<objectid>] [FOR <count>]

Lists objects from the specified object ID forward for the specified number of objects. If you do not specify an object ID, listing starts from the next available object immediately after the last object found and displayed from the previous Listobjects, LForward, or LBackward command. If no object was previously found and displayed, listing starts from the first object for the LForward command. If you do not specify a count value with the FOR phrase, CSM\_tool lists **all** objects from the starting object forward.

Specify objectid in one of the following formats:

```
[<cache_id> [<ssn> [object_id> [<page>]]]]
```

```
<cache_id> '-' [<cache_id>]
```

If you do not specify a range value after the hyphen ('-') in the second format above, CSM\_tool assumes the maximum value of the specified object level (cache ID level, ssn level, object ID level, or page level). See [“object range” on page 225](#) for more information on range level.

LForward output displays in paging mode. However, you can use the PAGING OFF command to set output to scrolling mode. (See [“Paging \[ON | OFF\]” on page 231](#).)

You can repeat the command to display the next batch of objects by pressing the carriage return (<cr>) key at the CSM\_tool prompt. The object batch size for display is determined by the last FOR count, if one was originally specified; otherwise, listing continues to the end in either scrolling or paging mode. The LForward command being repeated displays as the first line of the next group in the list. A greater than (>) sign indicates the list operation is in a forward direction beginning with the next objectid **after** the displayed objectid, as shown below:

```
<CSM_tool> lf
(objs > 3      1717      100105      1)
cache_id      ssn      object_id      page      max_length
-----      ---      -----      ----      -----
3              1717      100105      1          10240
```

A greater than, equal to (>=) sign indicates the list operation is in a forward direction beginning with the displayed objectid, as shown below:

```
<CSM_tool> lf 1 1717 100100
(objs >=      3      1717      100105      1)
cache_id      ssn      object_id      page      max_length
-----      ---      -----      ----      -
1             1717      100100      1         4096
3             1717      100104      1         10240
3             1717      100105      1         10240
```

See the Listobjects command for a comprehensive example of listing cache objects with the Listobjects, LForward, and LBackward commands.

### Listobjects [**<object range>**] [**FOR <count>**] [**desc**]

Displays a list of specified objects or a range of objects. You can specify that the list begin with a particular object or you can list objects within a range from a specified beginning object. You can limit the number of objects listed by specifying a count value in the FOR phrase. You can also list objects in ascending order (the default) or descending order (by adding the **desc** option).

For example, the following command lists, in ascending order, from the first object in cache 1 until 100 objects are displayed or the end of object list is reached:

```
list 1 for 100
```

**Tip** Some Listobjects commands are equivalent to LForward and LBackward commands. For example, **listobjects 1 for 100** is essentially the same as the **listforward 1 for 100** command.



If you specify both object range and the FOR phrase, the FOR count supersedes the object range in determining the number of objects. If you do not specify object range, listing starts from the first or last object for ascending or descending order, respectively.

To list from the last object in cache 1 down to the first object in cache 1, use the following command:

### **list 1 desc**

The listobject output displays in paging mode. However, you can use the PAGING OFF command to set output to scrolling mode. (See [“PAging \[ON | OFF\]” on page 231.](#))

### **object range**

The four level numbers that identify the cache object: cache\_id, ssn, object\_id, and page. Each of these levels is described in detail below.

**Cache\_id** is the identification number of the cache. Each cache type (page cache, batch entry cache, and so on) has a unique ID number. See the CSM\_tool statistics command on [page 235](#) for information about cache names.

**ssn** is the system serial number.

**object\_id** is the identification number used by the system to track each scanned object in cache.

**page** is the page number of the cache object. If no page number exists, the default is 65535.

Specify <object range> in one of the following formats:

```
[<cache_id> [<ssn> [object_id> [<page>]]]]
```

<cache\_id> '-' [<cache\_id>]

<cache\_id> <ssn> '-' [<ssn>]

<cache\_id> <ssn> <object\_id> '-' [<object\_id>]

<cache\_id> <ssn> <object\_id> <page> '-' [<page>]

If you do not specify a range value after the hyphen ('-'), CSM\_tool assumes the maximum value of the specified range level (cache ID level, ssn level, object ID level, or page level).

The values for the range can be in ascending or descending order. If you specify both <object range> and the FOR phrase, the FOR count supersedes the <object range> in determining the number of objects. If you do not specify an object range, the listing starts from the first object for ascending order or last object for descending order.

You can repeat a Listobjects command by pressing the carriage return (<cr>) key at the CSM\_tool prompt. The repeated Listobjects command displays as the first line of the next group in the list.

To quickly identify objects in cache, you can specify a range of objects or use the **FOR** phrase to list a range and/or a specified number of objects. For example:

- **list 1** - lists objects from cache ID 1 to the last cache ID. Because no range is specified after the hyphen, CSM\_tool assumes the maximum range and lists all caches after cache ID 1.
- **list 1 6000 100000-200000** lists all the objects within the range of cache\_id 1, ssn 6000, and object ID 100000 to object ID 200000
- **list 1 for 3** lists the first 3 objects from the beginning of **cache ID 1**

- **list 1 for 3 desc** lists the last 3 objects from the end of **cache ID 1** in descending order

If no objects exist in the specified range, the following message displays:

No objects exist in the specified range

The output of listobjects displays the four level numbers (cache\_id, ssn, object\_id, page) and **max\_length**, which is the amount of space in bytes allocated for the object.

You can use the listobjects information as input for other cache-related commands (such as CSM\_exim) or other CSM\_tool commands. For example, to delete a document from the batch entry cache with the CSM\_tool deleteobjects command, you need that document's cache attributes. The listobjects command displays the attributes, as shown in the following sample output:

cache_id	ssn	object_id	page	max_length	
-----	---	-----	----	-----	
1	10000	101019	0	132	
1	10000	101019	1	43760	
4	10000	101032	65535	44820	
4	10000	101047	65535	46728	ageable
4	10000	101049	65535	48480	

You can also specify a `cache_id` number to obtain a listing for a single cache. For example, a listing of objects in `cache_id` 4 only is similar to the following:

```
<CSM_tool> list 4
cache_id      ssn      object_id  page      max_length
-----      ---      -
4             10000    101032     65535     44820
4             10000    101047     65535     46728
4             10000    101049     65535     48480
```

Use `MKF_tool` to display records in the `csm_used_space` table if you need more information about a particular object.

### Example Listobjects Output

The following example shows the use of the `Listobjects`, `LForward`, and `LBackward` commands. Each command issued is in bold typeface. You can repeat some commands by pressing the carriage return (`<cr>`) key. No command text appears after the prompt for repeated commands. The first line of output for repeated commands displays the command being repeated:

```
<CSM_tool> l
cache_id      ssn      object_id  page      max_length
-----      ---      -
1             1717     100001     1          4096
1             1717     100010     1          8192
1             1717     100020     1          4088
1             1717     10002      1          4096      ageable
1             1717     100023     2          4096      ageable
1             1717     100023     3          4096      ageable
1             1717     100100     1          4096
```

```

3          1717      100104      1          10240
3          1717      100105      1          10240
3          1717      100106      1          10240
<CSM_tool> 1 1 1717 100020 -
cache_id      ssn      object_id      page      max_length
-----
1          1717      100020      1          4088
1          1717      100023      1          4096      ageable
1          1717      100023      2          4096      ageable
1          1717      100023      3          4096      ageable
1          1717      100100      1          4096
<CSM_tool> 1 1 for 3
cache_id      ssn      object_id      page      max_length
-----
1          1717      100001      1          4096
1          1717      100010      1          8192
1          1717      100020      1          4088
<CSM_tool> 1 1 for 3 desc
cache_id      ssn      object_id      page      max_length
-----
1          1717      100100      1          4096
1          1717      100023      3          4096      ageable
1          1717      100023      2          4096      ageable

<CSM_tool> 1 1 1717 100100 for 3
cache_id      ssn      object_id      page      max_length
-----
1          1717      100100      1          4096
3          1717      100104      1          10240
3          1717      100105      1          10240
<CSM_tool> (<cr>)
1 1 1717 100100 for 3
1          1717      100100      1          4096
3          1717      100104      1          10240
3          1717      100105      1          10240
<CSM_tool> lf
(objs >      3          1717      100105      1)
cache_id      ssn      object_id      page      max_length
-----

```

```

3          1717          100106          1          10240
<CSM_tool> lb
(objs <          3          1717          100106          1)
cache_id      ssn      object_id      page      max_length
-----
3          1717          100105          1          10240
3          1717          100104          1          10240
1          1717          100100          1          4096
1          1717          100023          3          4096      ageable
1          1717          100023          2          4096      ageable
1          1717          100023          1          4096      ageable
1          1717          100020          1          4088
1          1717          100010          1          8192
1          1717          100001          1          4096
<CSM_tool> lf 1 1717 100100
(objs >=          1          1717          100100          0)
cache_id      ssn      object_id      page      max_length
-----
1          1717          100100          1          4096
3          1717          100104          1          10240
3          1717          100105          1          10240
3          1717          100106          1          10240
<CSM_tool> lf
(objs >          3          1717          100106          1)
No objects exist as specified.
<CSM_tool> lb for 3
(objs <          3          1717          100106          1)
cache_id      ssn      object_id      page      max_length
-----
3          1717          100105          1          10240
3          1717          100104          1          10240
1          1717          100100          1          4096
<CSM_tool> (<cr>)
lb for 3 (from obj <          1          1717          100100          1)
1          1717          100023          3          4096      ageable
1          1717          100023          2          4096      ageable
1          1717          100023          1          4096      ageable
<CSM_tool>lb for 3 (from obj <          1          1717          100023          1)
1          1717          100020          1          4088
1          1717          100010          1          8192

```

---

```

1          1717          100001          1          4096
<CSM_tool>
lb for 3 (from obj <          1          1717          100001          1)
No objects exist as specified.
<CSM_tool> q

```

---

**LOCKdoc <doc\_id>**

(Requires ALLOWUPDATES.)

Locks all pages of the document specified by <doc\_id> in the ageable cache (that is, the page cache) used by document services. To lock a document, it must already exist in this cache in an unlocked state before you use this command.

**OBjecttofile <cache\_id> <ssn> <object\_id> <page> <filename>**

Copies the cache object identified by <object\_id> to the UNIX file specified by <filename>.

Obtain cache attributes (cache\_id, ssn, object\_id, page) from the Listobjects command.

The objecttofile command can help you troubleshoot problems with corrupted images. For example, you might need to convert an image object to a UNIX file so that you can run the check\_page tool against the file to check for corruption.

**PAging [ON | OFF]**

Starts or stops paging mode for the list commands (listobjects, listforward, listbackward). ON is the default mode. PAging OFF sets the output display mode to scrolling.

**Quit**

Exits CSM\_tool.

**REBUILDstats [DEBUG]**

Scans all cache objects for statistics and rebuilds in-use and locked objects/sectors statistics.

The DEBUG option displays both the in-memory statistics and statistics from the transient database before updating.

When FileNet software starts, it scans cache objects for statistics if the CSM\_SCAN\_CACHE file exists. The presence of the CSM\_SCAN\_CACHE file causes an offline scan of the cache to occur every time the software restarts.

You must manually create the CSM\_SCAN\_CACHE file using a system command or utility (for example, using the touch command or the vi utility in UNIX platforms, or using edit or Notepad in Windows Server platforms). You should find the CSM\_SCAN\_CACHE file in the following directories:

/fnsw/local/sd                      for UNIX platforms

<drive>:\fnsw\_loc\sdc              for Windows Server platforms

or

<drive>:\fnsw\local\sdc

However, you can use the rebuildstats command to force the system to scan statistics whenever you deem it necessary.



The following example is the output from the rebuildstats command. To run this command, you must use the ALLOWUPDATES command first. Note that, for security reasons, the password you enter for the ALLOWUPDATES command does not display.

```
<CSM_tool>allowupdate
Password?
Updates now allowed
<CSM_tool>rebuild
This command requires the exclusive use of the cache to scan the entire
cache space, and may take a while to complete depending on the number of
objects in cache.
Use Ctrl-c to abort.
Current stats shows a total of 522 objects in cache.
Scanning all cache used space.  Number of objects scanned:
522
New stats determined from offline scan:
cache  inuse_objs  inuse_secs  locked_objs  locked_secs
-----  -----  -----  -----  -----
  0         522        16261         522        16261
  1          0          0           0           0
  2          0          0           0           0
  3          0          0           0           0
  4          0          0           0           0
Rebuildstats is done.
```

In the example below, the DEBUG option of the rebuildstats command displays old (in-memory and transient database) statistics and new statistics. Then rebuildstats prompts you to confirm the update before proceeding.

---

<CSM\_tool>**rebuild debug**

This command requires the exclusive use of the cache to scan the entire cache space, and may take a while to complete depending on the number of objects in cache. Use Ctrl-c to abort.

Current stats shows a total of 522 objects in cache.

Scanning all cache used space. Number of objects scanned:

522

New stats determined from offline scan:

cache	inuse_objs	inuse_secs	locked_objs	locked_secs
----	-----	-----	-----	-----
0	522	16261	522	16261
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

Old stats in memory: (\* indicates disparity from offline scan)

cache	inuse_objs	inuse_secs	locked_objs	locked_secs
----	-----	-----	-----	-----
0	522	16261	522	16261
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

Old stats in trandb:

cache	inuse_objs	inuse_secs	locked_objs	locked_secs
----	-----	-----	-----	-----
0	522	16261	522	16261
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

Ok to update stats (y/n)? **y**

Rebuildstats is done.

---

## SCavenge

(Requires ALLOWUPDATES.)

Rebuilds the free space chain for all caches on the server. Progress messages display during operation and an incremental counter indicating the number of objects scanned advances. The output is similar to the following:

```

<CSM_tool>allowupdate
Password?
<CSM_tool>sc
Deleting csm_free_space.  Number of rows deleted:
8
Refreshing cache used space.  Number of objects
refreshed:
522 (of 522 or 100.00%) done.
Current stats shows a total of 522 objects in cache.
Scanning all cache used space.  Number of objects
scanned:
522
New stats determined from offline scan:
cache  inuse_objs  inuse_secs  locked_objs  locked_secs
-----
-
    0          522      16261         522      16261
    1           0         0           0         0
    2           0         0           0         0
    3           0         0           0         0
    4           0         0           0         0
Scavenge is done.

```

### Statistics [<cache\_id>] [ { long | sect } ]

Displays statistics for the specified cache or for all caches if you do not specify a cache\_id. You can specify the long or sector format for your

output listing. If you do not specify an format, a summary report is the default. Each line in the report provides information on one type of cache as shown below:

Cache Id	Name	% locked	% full	% free
-----	-----	-----	-----	-----
1	page_cache1:corona:FileNet	1	1	99
2	print_cache1:corona:FileNet	0	0	100
3	fillin_cache1:corona:FileNet	0	0	100
4	bes_cache1:corona:FileNet	1	1	99
*	Physical space summary	1	1	99

This report provides cache statistics as described in the following table.

#### Statistics Report Description

Column Head	Description
<b>Cache ID</b>	Cache identifier
<b>Name</b>	Cache name, based on function, such as batch entry (bes)
<b>% locked</b>	Percentage of locked objects in the cache
<b>% full</b>	Percentage of cache space occupied by locked and unlocked objects
<b>% free</b>	Percentage of cache space available for this cache

**Note** The cache space includes the reserved space (as specified by the `min_cache_size` in the system configuration file) and the shared space.

When reviewing the **% free** information, be aware that the percent full plus the percent free do not necessarily equal 100. Consider this example: 1000 sectors of free space are available on a server and two caches are configured on this server. Each cache has a `min_cache_`

size of 200 and a `max_cache_size` of 800. When the first cache becomes 100% full (800 sectors in use) and the second cache is empty, the first cache is 0% free, but the second cache is only 25% free. This situation occurs because there are only 200 sectors available in the magnetic disk pool. Therefore, the second cache can only become  $200/800 \times 100$ , or 25%, full before space is exhausted.

You can configure caches so that even if no objects exist in any cache on a server, no cache has 100% free space. Consider the example in which 1000 sectors of magnetic disk are available for cache and two caches are configured with a maximum cache size of 800 and a minimum size of 400. In this case, neither cache could contain more than 600 sectors of data because the other cache has 400 sectors (the minimum size) reserved.

The amount of free space for a cache could be negative. This condition can occur when a system is reconfigured to reduce the cache size but objects exist in the cache before the reconfiguration. Negative free space can also occur when MKF transactions are aborted and objects that were moved between caches are moved back. Negative free space in a cache does not cause a system malfunction. You can resolve it by deleting items in cache.

Overflow from the MKF transient database can prevent the addition of new objects to cache, even if free space in the cache exists. If this occurs, the following message displays at the bottom of the statistics information:

```
MKF transient database full--cache freespace not usable
```

If you see this message, configure a larger MKF transient database and run `CSM_tool scavenge` to recover the lost cache free space. No cache data is lost when system operation resumes.

If you specify the long option for your output, CSM\_tool produces more detailed information on each type of cache. The following is sample output from the long option:

---

```

Statistics for cache #1, name = 'page_cache1:corona:FileNet'
min_cache_sectors  20000      locked_sectors  44      inuse_sectors  44
max_cache_sectors  20000      locked_objects  2      inuse_objects  2
free_sectors       19956      ageable         T      refcnts        F
self_cleaning      F

```

```

-----
Statistics for cache #2, name = 'print_cache1:corona:FileNet'
min_cache_sectors  10000      locked_sectors  0      inuse_sectors  0
max_cache_sectors  30000      locked_objects  0      inuse_objects  0
free_sectors       30000      ageable         F      refcnts        T
self_cleaning      F

```

```

-----
Statistics for cache #3, name = 'fillin_cache1:corona:FileNet'
min_cache_sectors  1000      locked_sectors  0      inuse_sectors  0
max_cache_sectors  10000      locked_objects  0      inuse_objects  0
free_sectors       10000      ageable         F      refcnts        F
self_cleaning      T

```

```

-----
Statistics for cache #4, name = 'bes_cache1:corona:FileNet'
min_cache_sectors  10000      locked_sectors  44      inuse_sectors  44
max_cache_sectors  69000      locked_objects  1      inuse_objects  1
free_sectors       68955      ageable         F      refcnts        F
self_cleaning      F

```

```

-----
Physical space summary
reserved_sectors  41000      locked_sectors  88      inuse_sectors  88
max_cache_sectors  99999      locked_objects  3      inuse_objects  3
free_sectors       99911      largest_fs_blk  99901

```

```

-----
Prefetch duration.....300 secs

```

```
Migrate duration.....300 secs  
Refresh duration.....120 secs  
Cache daemon threshold.....80%  
Locked object threshold.....85%  
Writes critical threshold....90%
```

---

The output is divided into three general sections. The first section consists of information on each cache (see below for descriptions of this information). The second, titled “Physical space summary,” contains space information on the entire cache. The third section contains additional general information on the entire cache.

## Statistics Long Output - First Section

The table below describes the first section of the Statistics output. This information is printed for each cache:

Statistics Long Output - First Section

Field Name	Description
min_cache_sectors	Minimum size of the cache in sectors
max_cache_sectors	Maximum size of the cache in sectors
free_sectors	Number of cache sectors not in u
self_cleaning	T indicates a self-cleaning cache F indicates cache is not self-cleaning All objects in a self-cleaning cache are deleted when the FileNet system starts. COLD and fill-in caches are examples of self-cleaning caches.
locked_sectors	Number of cache sectors used by locked objects
locked_objects	Number of locked objects in cache. Locked objects cannot be aged out of cache
ageable	T indicates an ageable cache F indicates all objects in the cache are not ageable
inuse_sectors	Number of sectors currently used by objects in the cache (An object uses at least one sector.)
inuse_objects	Number of objects that exist in the cache
refcnts	T indicates all objects in the cache are reference countable F indicates the objects in the cache are not reference countable A reference countable cache measures current usage of a cache object by using a reference count to the object's attribute values. In contrast, an ageable cache measures usage of an object by the duration an object is in cache.



### Statistics Long Output - Second Section (Physical Space Summary)

The second section of the long display displays the following physical space information:

Statistics Long Output - Second Section (Physical Space Summary)

Field Name	Description
reserved_cache sectors	Number of reserved or in-use sectors, whichever is larger, for all caches The value is the larger of one of the following: a) The total of the minimum cache sizes from all caches b) The number of in-use sectors in all the caches
max_cache_sectors	Amount of disk space available to all caches on this server
free_sectors	Number of unused sectors in the cache
locked_sectors	Number of locked sectors in all caches
locked_objects	Number of locked objects in all caches
largest_fs_blk	Size in kilobytes of the largest block of free space in the cache. Free space is allocated in sectors; each sector is 1 KB in size
inuse_sectors	Number of in-use sectors in all caches
inuse_objects	Number of in-use objects in all caches

### Statistics Long Output - Third Section

The final section of the display provides cache summary information. The following cache configuration attributes apply on to the ageable cache.

## Statistics Long Output - Third Section

Field Name	Description
Prefetch duration	<p>Minimum number of seconds a prefetched object is guaranteed to remain in cache from the time is prefetched before becoming a candidate for deletion by the CSM_daemon</p> <p>(The value should be large enough so that prefetched objects remain in the cache for user access.)</p>
Migrate duration	<p>Minimum number of seconds after being written to media that an object remains in cache before becoming a deletion candidate</p>
Refresh duration	<p>Minimum number of seconds an object remains in cache after being accessed for a second time</p>
Cache daemon threshold	<p>Percentage of cache that must be full of in-use cache objects before the cache_daemon begins deleting ageable objects</p> <p>This threshold controls free space within the cache. The CSM_daemon automatically begins deleting ageable objects once the cache reaches this threshold.</p>
Locked object threshold	<p>Percentage of cache that must be full of locked objects before scanning is stopped</p> <p>Once this threshold is reached, no additional uncommitted images could be added to the cache.</p>
Writes critical threshold	<p>Percentage of cache that must be full of in-use objects for write requests to take precedence over read request</p> <p>When the cache reaches this threshold, write requests (which make locked objects ageable) have priority over read requests.</p>

**TERMOFF | TERMON**

The CSM\_tool list commands normally write output to the display device. This type of output is also called “terminal output.” Use TERMOFF to disable output to the display device. You want to do this when printing long lists to a file via the hardcopy command. (See [“HArdcopy \[<filename>\]” on page 217.](#)) To turn terminal output back on, use the TERMON command.

**TRACE [ON [<pid>] | OFF]**

(Requires ALLOWUPDATES.)

Starts and stops tracing activity. If you specify a process ID (pid), the system traces only the specified process. Otherwise, it traces the cache activities of all processes. It also enables CSM\_daemon and CSMs to display trace information on the screen.

The system stores trace data in the standard error log file under the file path specified for your platform:

/fnsw/local/elogs/elog<date>    UNIX

SW\_LOC\logs\elogs\el<date>    Windows Server

## Checklist

Before you use CSM\_tool, be aware of the following:

- You must be a member of the fnadmin group or have a valid fnlogon session to run CSM\_tool.
- You must run CSM\_tool commands from the server, not remotely.
- You can use CSM\_tool only at a server that contains a transient database and the transient database must be running.
- Do not use the deleteobjects, initialize, and initpartition commands under normal operating conditions. Doing so could corrupt your system. Contact your service representative for assistance before running these commands.
- At the <CSM\_tool> prompt, enter ? to get help text for available commands.

---

**Note** To use some commands, you must first provide a password through the ALLOWUPDATES command. If you have not issued ALLOWUPDATES, CSM\_tool help text does not display the commands that require ALLOWUPDATES.

---

## Procedure

- 1 At a FileNet server, enter the following command:

**CSM\_tool**

The <CSM\_tool> prompt appears.

- 2 Call your service representative for the ALLOWUPDATES password, if necessary.

To use commands that change the cache (such as deleting objects and initializing cache and partitions), contact your service representative for the required ALLOWUPDATES password. Then enter the ALLOWUPDATES command.

- 3 Enter commands at the <CSM\_tool> prompt.

## Related Topics

[“check\\_page” on page 154](#)

[“CSM\\_exim” on page 190](#)

[“fnlogon” on page 489](#)

[“MKF\\_tool” on page 801](#)

# cstat

## Description

**UNIX**

cstat is available only on a UNIX-based Image Services server.

A program creates a core file when it aborts. If a running program creates a core file, the cause could be with that program, with a shared library, or with a file that the program calls. Core files contain information that could help to determine the cause of the failure. The cstat tool reads the core file and displays information about the core file, such as the name of the program that generated the core file.

A program debugger, such as dbx or xdb, is necessary to perform a complete analysis of the core file. For more information about these debugger programs, use the appropriate UNIX **man** command to display the related on-line manual pages.

## Use

Use cstat to obtain information about a core file.

The cstat tool is most useful when you have many core files and need to know the names of the programs that aborted and the date and time the programs aborted.

## Syntax

```
cstat [<pathname>]
```

**<pathname>** Full path name of the core file to examine

## Sample Output

```
corona(root)/> cstat /fnsw/local/tmp/core
====> AIX Version <====
core file      /fnsw/local/tmp/core
signal        (6) used by abort, replace SIGIOT in the future
command       bes_commit
process id    18105
program file  bes_commit
last 15 linked libraries ...
library file  /fnsw/lib/shobj/BESr
library file  /fnsw/lib/shobj/DOCb
library file  /fnsw/lib/shobj/DOCp
library file  /fnsw/lib/shobj/DOCr
library file  /fnsw/lib/shobj/OSIr
```

## Checklist

Before you use `cstat`, be aware of the following:

- Image Services for Windows Server does not support `cstat`.
- You should regularly monitor your system for the presence of core files. A recurring problem can create numerous core files that occupy large amounts of disk space.
- You should copy core files to tape. Your service representative could request that you send the tapes for diagnosis.

## Procedure

- 1 At the command prompt, enter **cstat** and the full path name of the core file.

Information about the core file displays.

- 2 Locate the failing program.

Find the “program file” line in the core file display. (The program file in [“Sample Output” on page 247](#) is bes\_commit.) This is the name of the program whose failure caused the core file to be created.

## Related Topics

[“cordebug” on page 163](#)

[“cormon” on page 166](#)



# cti

## Description

The cti tool counts the number of distinct non-null values recorded in the index database for a specified index. Performance varies depending on the uniqueness of the index. If the index is unique, cti optimizes its search by using the COUNT function available in SQL. If the index is not unique, the SQL COUNT function is not used and additional resources are required to determine the count, possibly affecting performance.

The cti tool establishes the interface to the RDBMS (Oracle, Microsoft SQL Server, or IBM DB2) before counting begins. When successfully logged on to the RDBMS, cti establishes a cursor for the SQL statement and builds the SQL statement to select the row for the specified index from the index table. After the cursor is established, the SQL statement executes, fetching the row for the specified index.

When specifying the index name, you must spell and capitalize it exactly as it was when originally created. It must be a retrieval key, not just an informational key, but need not be unique. However, if the key is not unique, performance could be affected. If the index name you enter is not a retrieval key, cti displays the following message:

Index not inverted. Count of distinct values for the index is not possible using this program.

Output from cti displays to standard output (usually the terminal screen or window from which you invoke cti).

## Use

Use **cti** to count the number of unique values recorded in the RDBMS index database for a specified index.

## Syntax

```
cti <index_name>
```

**<index\_name>** The name of a user index or a system index.

Enter **cti** with no index name to display help information.

## Sample Output

The following sample output shows two reports for a very small database:

```
costa5(kehr)/home/kehr> cti CreationRelease

Number of distinct values for the index is : 0
Total number of index values examined is : 0

costa5(kehr)/home/kehr> cti F_DOCNUMBER
Direct database count is used.  There will be no status prompts.

Number of distinct values for the index is : 715
Total number of index values examined is : 715
```

## Checklist

Before you use cti, be aware of the following:

- You must run cti locally on the Index server.
- You must spell and capitalize the index name exactly as it was originally created.
- The index name must be a retrieval key. (We recommend assigning a unique retrieval key; otherwise, it could affect performance.)

## Procedure

No specific procedure is required.

## Related Topics

See the “Database Maintenance” chapter of your *System Administrator’s Handbook* for a discussion of indexes and keys. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

# dbp

## Description

The dbp tool examines the document buffer pool to obtain statistics. Tool output displays the average and maximum wait times for buffers (in milliseconds), the number of available buffers, and the minimum available buffers.

- Wait times of 0 to 100 milliseconds indicate no problem.
- Wait times of 100 to 1000 milliseconds indicate a minor performance problem.
- High wait times (>5000 milliseconds) indicate a severe performance problem.

## Use

Use the dbp tool to diagnose performance problems such as excessive wait times in Document Services.

The -p option is rarely used and then only to display the process IDs of the buffer holders, if any.

## Syntax

```
dbp [-r | -s | -p]
```

- r Reset statistics
- s Display statistics
- p Display the process ID of the page owner

## Sample Output

In the example below, dbp is run with the -s option to retrieve statistics, and again with the -p option to retrieve process IDs for the page holders (none in this case).

```
corona(root)/> dbp -s
Directory buffers:
  Avg_wait_msec:      0,  Max_wait_msec:      0
  Num_avl_bufs:      4,  Min_avl_bufs:      4
Page buffers:
  Avg_wait_msec:      0,  Max_wait_msec:      0
  Num_avl_bufs:      4,  Min_avl_bufs:      4
Descriptor buffers:
  Avg_wait_msec:      0,  Max_wait_msec:      0,  Cache hits:
0
  Num_avl_bufs:      4,  Min_avl_bufs:      4,  Cache allocs:
0

corona(root)/> dbp -p
Page owner pids:
Desc owner pids:
Dir owner pids:
```

## Checklist

Before you use dbp, be aware of the following:

- The -s option is useful to analyze performance problems; the -p option is rarely used.
- You must run dbp on the Storage Library server.

## Procedure

No specific procedure is required.

---

# dbverify

## Description

The dbverify tool identifies those documents that are:

- In the index database but missing from the permanent database on the document locator server. If documents meeting this criteria are found, dbverify stores their document numbers in:

```
/fnsw/local/logs/inx_logs/Mloc<date_time>UNIX
```

```
<drive>:\fnsw_loc\logs\inx_logs\Mloc<date_time>Windows Server
```

- In the permanent database on the document locator server but missing from the index database. If documents meeting this criteria are found, dbverify stores their document numbers in:

```
/fnsw/local/logs/inx_logs/Midx<date_time>UNIX
```

```
<drive>:\fnsw_loc\logs\inx_logs\Midx<date_time>Windows Server
```

- In the permanent database on the document locator server but not written to media. If documents meeting this criteria are found, dbverify stores their document numbers in:

```
/fnsw/local/logs/inx_logs/Modk<date_time>UNIX
```

```
<drive>:\fnsw_loc\logs\inx_logs\Modk<date_time>Windows Server
```

Refer to the -o option under “Syntax” below.

The system date and time in `yyyymmdd_hhmmss` format are appended to each output file name.

The dbverify tool requires exclusive use of the databases. If another application has a lock on the database, dbverify can initiate deadlock retry operations. You can set parameters specifying the number of retries that dbverify should attempt and the delay between retry attempts. If dbverify attempts the maximum number of retries and a deadlock still exists, dbverify fails and displays an error message that indicates a deadlock condition exists.

When dbverify completes, a summary of missing documents and fetched documents displays.

## General Use

Use dbverify to verify that the records of the DOCTABA table in the index database and DOCS table in the permanent database compare correctly.

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3 and higher)

Unless you are confident that you know which IS documents have or do not have associated entries in DOCTABA, the dbverify tool is not a good way to verify database integrity.

As always, dbverify compares records in the DOCTABA and DOCS tables but will now report mismatches depending on how the documents are indexed in Image Services using the FileNet P8 Content Federation Services (CFS) functionality. Documents will always be indexed in the DOCS table on the MKF database, but sometimes not indexed in DOCTABA on the Index database, so this mismatch will be caught by dbverify. For example, documents that are stored in the Image Services system and cataloged in the Content Engine system could retain their Image Services index information in DOCTABA. Depending on how the document originally entered the FileNet



system, cataloging could have occurred only in the Content Engine. By design, some IS documents will not have entries in DOCTABA.

Consequently, you need to search through the output of dbverify to identify mismatches, because if the image is not indexed in DOCTABA but the image is stored on the IS system, dbverify will report a mismatch. Refer to the table below to see where this mismatch can occur. Where you see an **N** in the table is where dbverify could report a mismatch.

<b>Document Activity</b>	<b>Index on CE</b>	<b>Index on IS</b>
New documents entered via Content Engine	Y	<b>N</b>
New documents entered via Image Services		
• Indexed on both IS and C	Y	Y
• Indexed only on CE	Y	<b>N</b>
Existing document images transferred to CE		
• Indexed on both IS and CE	Y	Y
• Index deleted from IS	Y	<b>N</b>

## Syntax

```
dbverify -s<document number> -e<document number> -o [-h]
          [-d] [-R<retries>] [-D<delay>]
```

- s<document number>** Specifies the starting document number of a document range. If not specified, the default starting number is 0.
- e<document number>** Specifies the ending document number of a document range. If not specified, default ending number is 0xFFFFFFFF.
- o** Identify documents that are not written to the media. If not specified, dbverify does not check media.
- d** Use defaults for unspecified options. The defaults are 0 for starting document number, 0xFFFFFFFF for ending document number, and no check of media.
- If you do not specify -d, dbverify prompts you for each parameter. See [“Sample Output” on page 259](#).
- h** Help lists the description of dbverify.
- [-R<retries>]** The maximum number of retry attempts in case a deadlock occurs. The default is 20 retries. However, you can specify 0.
- [-D<delay>]** The delay period between retries in 0.10-second units. The default is 100.

## Sample Output

If you enter dbverify with no options, the program prompts you to enter the starting document number, as shown below:

```
corona(fnsw)/home/fnsw> dbverify
enter starting document number or 'h' for help : h
```

where

- s specifies the starting document number, with 0 as the default.
- e specifies the ending document number, with 0x7FFFFFFF as the default.
- o specifies to identify the documents that are not written to the media, which do not identify the default.
- d uses defaults for the unspecified options.
- h list the description of dbverify.
- R deadlock retries (default 20)
- D deadlock delay in 0.010 second units (default 100)

dbverify is a utility that identifies

- a. those documents that are in the index database but are missing in the locator database, and stores their document numbers in file /fnsw/local/logs/inx\_logs/Mloc\*
- b. those documents that are in the locator database but are missing in the index database, and stores their document numbers in file /fnsw/local/logs/inx\_logs/Midx\*
- c. those documents that are not written to the media if -o option is specified, and stores their document numbers in file /fnsw/local/logs/inx\_logs/Modk\*

If you do not specify the `-d` option, and you do not need help, `dbverify` prompts you for each parameter, as shown in the example below (responses to prompts are in bold type):

```
corona(fnsw)/home/fnsw> dbverify
enter starting document number or 'h' for help : 0
enter ending document number or 'h' for help : 0x7fffffff
identifies document missing in media (y/n/h) : y

Number of documents missing in the index database : 0 (/fnsw/local/logs/
inx_logs/Midx9702)
Number of documents missing in the locator database : 0 (/fnsw/local/logs/
inx_logs/Mloc97)
Number of documents missing in the media : 0 (/fnsw/local/logs/inx_logs/
Modk970214)
Number of documents fetched from the index database : 0
Number of documents fetched from the locator database : 0
corona(fnsw)/usr/fnsw>
```

In the following output list, `dbverify` lists information on the activities performed as the verify takes place. This information is followed by a summary statement of the number of documents found to be missing in the index and locator databases, and the full path name of the file into which `dbverify` places the missing document numbers, if any. The output below is a partial listing:

```
corona(fnsw)/home/fnsw> dbverify -d -R0
Number of documents missing in the index database : 0 (/fnsw/local/logs/
inx_logs/Midx970214_153504)
Number of documents missing in the locator database : 19999 (/fnsw/local/
logs/inx_logs/Mloc970214_153504)
Number of documents fetched from the index database : 19999
Number of documents fetched from the locator database : 0
```

If you do not specify the -d option, dbverify prompts you for each parameter, as shown in the example below (responses to prompts are in bold type):

```
corona(fnsw)/home/fnsw> dbverify
enter starting document number or 'h' for help : 0
enter ending document number or 'h' for help : 0x7fffffff
identifies document missing in media (y/n/h) : y

Number of documents missing in the index database : 0 (/fnsw/local/logs/
inx_logs/Midx9702)
Number of documents missing in the locator database : 0 (/fnsw/local/logs/
inx_logs/Mloc97)
Number of documents missing in the media : 0 (/fnsw/local/logs/inx_logs/
Modk970214)
Number of documents fetched from the index database : 0
Number of documents fetched from the locator database : 0
corona(fnsw)/usr/fnsw>
```

## Procedure

No specific procedure is required.

# dclview

## Description

The dclview tool creates views on DOCTABA corresponding to document classes. You can then use these views from sqlplus (Oracle), SQL Report Writer (Oracle), isql (MS SQL Server), or Command Line Processor (IBM DB2) to provide a user-oriented view of DOCTABA.

For each document class, dclview creates a different view. The format of the view name F\_<document class name>, that is F\_ followed by the document class name. In addition, dclview creates a general view, F\_DOCTABA, of all columns in the table.

Each view contains all FileNet columns and user columns for the document class. However, some columns in the view are different from those in DOCTABA. For example, f\_entrydate in DOCTABA is a number but it is displayed in a view in ddmmyyyy format, where dd is the day, mm is the month, and yyyy is the year.

## Use

Use dclview to create views on DOCTABA which you can then use to create user-oriented views.

An option is also available for simply viewing the statements that create the view. Use this option if you want to examine the statements before actually creating the view on the database.

## Syntax

```
dclview [ -c ] [ -g ] { -a | dclname ... }
```

- c** Create the views in the database. If not specified, the default is to display the CREATE VIEW statements on the standard output.
  - g** Create the general view on all columns.
  - a** Create the general view and create views on all document classes. If you specify -a, -g option is automatically set. Mutually exclusive with the use of dclname.
- dclname** Create views on only the specified document classes. Mutually exclusive with the -a option.

## Sample Output

In the following example, the `dclview -g` is run to display the CREATE VIEW statements for a general view of DOCTABA.

```
corona(fnsw)/home/fnsw> dclview -g
create view "F_DOCTABA"
( F_DOCNUMBER,F_DOCCLASSNAME,F_ARCHIVEDATE,F_DELETEDATE
,F_ENTRYDATE,F_RETENTOFFSET,F_PAGES,F_DOCTYPE,F_RETENTBASE,F_RETENTDISP,
F_ACCESSRIGHTS,"I1") as select
  F_DOCNUMBER,F_DOCCLASSNAME,TO_DATE('01-JAN-70')+D.F_ARCHIVEDATE,
TO_DATE('01-JAN-70')+D.F_DELETEDATE,TO_DATE('01-JAN-70')+D.F_ENTRYDATE,
NVL(D.F_RETENTOFFSET,0),NVL(D.F_PAGES,1),
DECODE(D.F_DOCTYPE,null,'Image',1,'Text',2,'Form',3,
'Mixed',4,'Other',D.F_DOCTYPE),
DECODE(D.F_RETENTBASE,null,'Closing',1,'Entry',D.F_RETENTBASE),
DECODE(D.F_RETENTDISP,null,'Delete',1,'Archive',D.F_RETENTDISP),
SUBSTR(DUMP(D.F_ACCESSRIGHTS,16),16,35),substr(A31,1,6)
  from document_class dc, doctaba d where
dc.f_docclassnumber = d.f_docclassnumber
```

## Checklist

Before you use `dclview`, be aware of the following:

- You must run `dclview` from the Index server.
- You must run `dclview` with no options to get online help.



## Procedure

No specific procedure is required.

## Related Topics

See the *Index and WorkFlo Database Contents Manual* for a discussion of DOCTABA.

See your *System Administrator's Handbook* for information about document classes.

To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

# ddexim

## Description

The ddexim tool saves (exports) and restores (imports) document class and index information in the index database.

The ddexim tool exports/imports all Data Dictionary information currently in system memory (document classes, indexes, CE Relationships, keys, clusters, forms, menus, validation tables, aperture card files, media families, WorkFlo queues, surfaces [if specified]) to a user-specified file on magnetic tape or magnetic disk. You can also control the amount of information that ddexim processes. For example, you can exclude the following information from the export:

- Oracle information
- Forms
- Menus
- Validation tables
- Aperture card information
- Family information
- WorkFlo queue information
- CE Relationships

You can selectively include document classes or media surface information during the export.

Normally, ddexim directs the output of an export operation to an ASCII file. However, you can redirect this output to a terminal device or window.

After exporting index information to a file, you can import that information to current system memory. All information is imported from the file unless you specify options to exclude certain information. You can also cause ddexim to update information in the index database during the import. This update can occur only when index information in the database matches information in the imported ASCII file. For example, if you are trying to update the index database, the index name in the database must be the same as the index name in the imported ASCII file.

## Use

Use ddexim to save (export) or restore (import) document class information and index database information. You can also use ddexim to examine database configurations. The program writes to the standard output so you can view its output.

Use the `-sd` option to exclude export or import of dumped information (for example, ASCII or hexadecimal dumps of old forms, menus, validation tables, and aperture card files).

Any FileNet user can run ddexim export. However, only the SysAdmin user or members of the fnadmin group, or a user with a valid fnlogon session, can run ddexim import.

If you are working on a system with a **multi-server configuration**, ddexim importing and/or exporting should be run on the Root/Index server **only**.

## Exporting

The ddexim export operation writes index, document class, family, form, WorkFlo queue, and media surface information to an ASCII file.

For FileNet P8 Content Federation Services systems with configured Content Engine Object Store mapping, ddexim will now always export the Content Engine configuration information except when the `-sc` (skip CE Configuration) option is selected.

## Importing

The ddexim import operation restores all information in the ASCII file directly to the index database.

You can use ddexim during recovery procedures to import index and document class information to the index database. Import recreates the information without re-entering all the index and document class information.

If you attempt to import a cluster that already exists in the index database, ddexim issues the following informational message:

```
Importing cluster on index 'cluster_string'... (already exists)
```

However, if you use the modify option (`-m`), ddexim performs the import, overwriting existing cluster information.

If you attempt to import a document class that has associated remote families, and the domain for the remote families does not exist, ddexim issues the following message and continues processing:

```
Can't update remote family for local family <family name>...
```

---

If you attempt to import a document class or work space that does not have a security group defined in the local system, ddexim issues the following warning message and sets the security group to (NONE):

```
Security group '<group>' does not exist for
<workspace|class>, '<workspace_name>|<class_name>', setting
to '(NONE)'
```

On an FileNet P8 Content Federation Services system with configured Content Engine Object Store mapping, if you perform a ddexim import and the system serial number (SSN) in the export file matches the local SSN, the content Engine configuration will be imported unless the -sc option is selected. If the local SSN and export file SSN do not match, by default the Content Engine configuration file will not be imported. The following message will be displayed:

```
CE to DCL Configuration was not applied because the SSNs did not match.
Import File SSN=<File SSN> Local SSN=<Local SSN>
To configure CE INFO anyways use the '-C' option
```

To override this, use the -C (always import CE configuration) option when the SSNs do not match.

---

**Note** The ddexim export files generated with ddexim that has Content Engine information will not be compatible with older Image Services systems like 4.0.20.

---

---

## Syntax

```
ddexim { -e [> <filename>] | -i <impfile>} [options]
```

- e** Export information from the index database and write the results to standard output.
- > filename** Directs exported index information to the specified ASCII file. Used only with -e option. If you do not include the > symbol, Data Dictionary information prints to the terminal screen but is not saved to an ASCII file.
- i <impfile>** Import information from the specified file to the index database. All existing index information is deleted.

## Options

- sn** Skips (exclude) import or export of *n* where *n* is one or more of the following:
  - o** RDBMS information: document class, index, key, and cluster descriptions
  - d** Dumped information (for example forms, menus, validation tables, and aperture card files)
  - f** Family information: media family descriptions
  - w** WorkFlo queues
  - c** CE Configuration: for IS systems that participate in FileNet P8 Content Federation Services with Content Engine systems, the -sc option skips the Content Engine configuration information.
- as** Includes import/export information about surfaces. Import occurs only if source SSN = target SSN

**-c <classname>**

Export/import only the specified document class and related indexes and families, but no media surface information.

For more than one document class, run ddexim with this option for each document class.

---

**Note**

Clustering information is exported for specified document classes **only** if clustering is set for that document class.

---

**-C** Option used while importing only. For IS systems that participate in FileNet P8 Content Federation Services with Content Engine systems, the -C option indicates that Content Engine information should be imported even though the local system serial number (SSN) does not match the SSN in the export file. This -C option always imports the Content Engine configuration.

**-m** Modifies existing information in the Data Dictionary during import. Only new or updated information is added to the Data Dictionary. Existing keys, indexes, and surfaces are **never** modified. On FileNet P8 Content Federation Services systems, this also holds true. If document classes already exists, they will not be updated unless the modify flag is set.

For sample FileNet P8 Content Federation Services system output where the modify flag is not set, see [\*\*“FileNet P8 Content Federation Services Output - Modify Flag is Not Set” on page 275.\*\*](#)

For sample FileNet P8 Content Federation Services system output where the modify flag is set, see [\*\*“FileNet P8 Content Federation Services Output - Modify Flag is Set” on page 276.\*\*](#)

## Sample Output

Use a text editor to view the ASCII file that was created by ddexim export. The ASCII file contains blocks of Data Dictionary information. Each block contains the following elements:

- Index information type (document class, index, or WorkFlo queue)
- Index information name

This is the name you assigned to the document class, index, or WorkFlo queue.

- Index information attributes

Attributes include descriptions of a document class or index. For example, the attributes for a document class can include pages per document, batch size, and family name.



---

The following sample output shows several blocks of information in an ASCII file created by the -e option. The blocks include a document class, WorkFlo queue, family, and index.

---

```
class dialog {
description "dialog"
security_read_name "(ANYONE)"
security_write_name "(ANYONE)"
security_exe_name "(ANYONE)"
retent_disp delete
retent_base rel_to_entry
retent_offset 12
pages_per_doc 0
tab_out_flag false
verify_images false
verify_indexes false
batch_total false
batch_size 5
family_name HPrint
no_catalog_flag false
index Description: required=f batch_totals=f verify=f
source=1
index TestDate: required=f batch_totals=f verify=f source=1
index TestNotes: required=f batch_totals=f verify=f source=1
index Description2: required=f batch_totals=f verify=f
source=1
index numeric: required=f batch_totals=f verify=f source=1
index processdate: required=f batch_totals=f verify=f
source=1
}
wflq q1 joec {
    nch_object_name WflServer
    desc_read_sec_name "(ANYONE)"
    desc_write_sec_name "(ANYONE)"
}
```

```
desc_exe_sec_name "(ANYONE)"
content_read_sec_name "(ANYONE)"
content_write_sec_name "(ANYONE)"
content_exe_sec_name "(ANYONE)"
description "queue for testing dump operations."
field: fld1 typ=1 len=16 unique=0 req=t rendez=f disp=t
}
...
family tranlog {
  is_primary false
  tran_families
  interleave_cnt 1
  disk_type 5
  server 2: desired_cur_surfs=0 preferred_osars=*
}
...
index Description {
  description "Description Uppercase Convert"
  internal_name A31
  type ascii
  maxstrlen 60
  upper false
}
index TestDate {
  description "Date of the Test"
  internal_name A32
  type date
  upper false
}
```

---

---

## FileNet P8 Content Federation Services Output - Object Store Configuration Data

Sample object store configuration data from the ddexim export file:

---

```
ce_object_store_guid {11111111-1234-1234-1234-123456789013} {  
  ce_object_store_name "OS1111"  
  ce_domain_guid {12345678-1234-1234-1234-123456789012}  
  ce_domain_name "CE1"  
  class ce_im_only not_default_dcl_os  
  class ce_class default_dcl_os  
  class ce_class2 not_default_dcl_os  
}
```

---

## FileNet P8 Content Federation Services Output - Modify Flag is Not Set

The following is displayed if the Content Engine Object Store to Document Class mapping already exists and the modify flag was not set:

---

```
Importing Object Store:  
Object Store = 'OS1111' GUID = '{11111111-1234-1234-1234-123456789013}'  
CE domain = 'CE1' GUID = '{12345678-1234-1234-1234-123456789012}' .already  
exists
```

---

**FileNet P8 Content Federation Services Output - Modify Flag is Set**

The following is displayed if the Content Engine Object Store to Document Class mapping already exists and the modify flag was set:

---

```
Importing Object Store:  
Object Store = 'OS1111' GUID = '{11111111-1234-1234-1234-123456789013}'  
CE domain = 'CE1' GUID = '{12345678-1234-1234-1234-123456789012}' .modified  
Importing DCL='ce_class' default Object Store ...modified
```

---

The first line in each block of ddexim output contains a type and a name. Not all types and names are shown in the sample output. The table below contains the complete list:

class	Document class name
index	Index name
key	Key name
cluster	Cluster index name
form	Name of the class that uses the form (not currently used)
menu	Menu name (exported only if referenced by one of the indexes)
validation_table	Validation table name (exported only if referenced by one of the indexes) Validation tables could exist even though the PC WorkForce Desktop System does not currently use them
aperturecard_file	Aperture card file name (not currently used)
family	Media family name
surface	Surface ID of side A
wflq	WorkFlo queue name
ce_object_store_guid	Unique identifier assigned to the Image Services system when it has been defined as a fixed content device by a Content Engine system.

## Checklist

Before you use ddexim, be aware of the following:

- You should contact your service representative before you attempt an import.
- Verify that the Image Services software and the MKF and RDBMS databases are up before running this tool or the program will hang. Also, if the RDBMS is configured remotely, verify that it is also brought up by the Database Administrator.
- If the databases reside remotely, verify that the IS software and databases are all up and running on the remote servers to avoid a program hang. For example, when running this tool on an Application server, verify that the Root/Index server is up and running.
- To use the ddexim import function, you must be the SysAdmin user, a member of the fnadmin group, or have a valid fnlogon session.
- The -i option replaces all Data Dictionary information with information in the specified file.
- If you want to import security definitions assigned to document classes on a Windows server with MS SQL Server installed, you must follow procedures described in **[“Import Security Definitions Procedure” on page 280.](#)**

- If ddexim is running in the foreground, press the Control+c key sequence to abort ddexim. If running in the background, enter one of the following commands depending upon the server's operating system, supplying the process ID (<pid>) for ddexim:

AIX**kill -31 <pid>**HPUXSOL**kill -17 <pid>**

## Procedure

### Export Procedure

- 1 Logon to the FileNet system.
- 2 Verify that the Image Services software and the MKF and RDBMS databases are up before running this tool or the program will hang. If the databases reside remotely, verify that the IS software and databases are all up and running on the remote servers to avoid a program hang.
- 3 Enter the ddexim -e command with appropriate options.

Use the ">" symbol to redirect output to a file.

- 4 View the file created by the export option.

If the output file contains system logon information on the first line, use a text editor to delete that information. The following is an example of system logon information:

```
SCT_logon(FN_USER=SysAdmin,PASSWORD=SysAdmin,TER-  
MINAL=)
```

### **Import Procedure**

- 1 Call your service representative.

Discuss Data Dictionary import options with your service representative. Choosing incorrect options could destroy all previously entered document class and index information.

- 2 Logon as the SysAdmin user, or as a user that is a member of the fnadmin group, or establish a valid fnlogon session.
- 3 Verify that the Image Services software and the MKF and RDBMS databases are up before running this tool or the program will hang. If the databases reside remotely, verify that the IS software and databases are all up and running on the remote servers to avoid a program hang.
- 4 Enter ddexim -i to import the index information.

Use the appropriate import options and specify the correct ASCII file name.

### **Import Security Definitions Procedure**

A red, slanted rectangular icon with the word "WIN" in white, bold, uppercase letters.

Before using ddexim to import security definitions for document classes, you must perform these procedures if you're running on a Windows server with a SQL Server installation.

On your Image Services server with an MS SQL Server installation, follow these steps:

- 1 Stop the FileNet software and verify that MS SQL Server is also stopped.
- 2 Open the Start menu and select the Programs menu.



- 3 Select MS SQL Server 6.5 from the Programs menu.
- 4 Select SQL Client Configuration Utility.
- 5 Verify that the Automatic ANSI to OEM option is turned off: If it is checked, click it to remove the check mark.
- 6 Start the FileNet software.
- 7 Import the data using this command:

```
ddexim -i <filename>
```

where <filename> is the name of the file containing the security definitions for document classes.

## Related Topics

### **[“fnlogon” on page 489](#)**

See the “Database Maintenance” chapter of your *System Administrator’s Handbook* for a description of document class, family information, index information, and WorkFlo queues.

See the *Index and WorkFlo Database Contents Manual* for descriptions of the index database and WorkFlo Queue database tables.

To download IBM FileNet documentation from the IBM support page, see **[“Accessing IBM FileNet Documentation” on page 21](#)**.

# deldocs

## Description

The deldocs tool deletes documents from the Image Services index database and/or the MKF permanent database. It could also delete selected documents from cache, given selected settings.

## Use

Use deldocs to delete documents that have been committed using the “no cataloging” option of Database Maintenance.

You can use deldocs to delete a single document, multiple documents, or all documents for a specified media surface. To delete multiple documents, specify a list of document IDs in a file that deldocs uses as input.

You can also delete all documents on a media surface and use Background Job Control Erase Media or Consolidate Media to erase the surface (only when using erasable media). The Consolidate Media function provides an option to erase the media as it consolidates the media.

---

**CAUTION**

This command deletes all references to the documents from the Image Services index and MKF permanent databases. You can, however, re-import these deleted documents from storage media.

For details on deleting documents from cache, see [“Documents in Cache” on page 285](#).

---

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3 and higher)

The output of the deldocs tool could report some confusing information when being used with the FileNet P8 Content Federation Services functionality. Since the function of deldocs is to delete entries from the DOCS table and the DOCTABA table (when the image has been indexed on the Image Services system), but not the images themselves, the CE catalog entry could reference an image that it believes does not exist when the documents are stored on the Image Services system but indexed on the Content Engine system.

Additionally, in the instances where documents are indexed on the Image Services system and the deldocs tool is run, Index Services will create an export\_log entry to tell the FileNet P8 CFS Server for Image Services to delete the corresponding index entry on the Content Engine system. Refer to the table below to determine when documents on the IS server will not have the index information associated with them (the **Ns**) because if index information does not exist in DOCTABA for an image, there is no way to notify the Content Engine system to delete its catalog information for that image.

Document Activity	Index on CE	Index on IS
New documents entered via Content Engine	Y	<b>N</b>
New documents entered via Image Services		
• Indexed on both IS and C	Y	Y
• Indexed only on CE	Y	<b>N</b>
Existing document images transferred to CE		
• Indexed on both IS and CE	Y	Y
• Index deleted from IS	Y	<b>N</b>

## Syntax

```
deldocs { -f <filename> | -s <surfaceid> [-b] } [-i] [-v]
```

- f <filename>** Text file name containing document IDs, one ID per line
- s <surfaceid>** Delete all documents on the surface specified by <surfaceid>. This option does not erase documents from the media surface.
  - b** Delete documents from both sides of the surface specified by <surfaceid>. Must be used in conjunction with the -s <surfaceid> parameter.
  - i** Ignore single document storage (SDS) retention settings.
  - v** Verbose mode - lists individual document ID's that have been successfully deleted. This mode is less efficient but the totals are more exact because documents are deleted one at a time and are easier to keep track of. Use verbose mode if you need exact statistics. If the filename (-f) option is used with verbose mode, documents that are skipped because they don't exist in the permanent database are also listed.

---

**Note** **deldocs** will not delete a document if it is stored on an SDS device, the **SDS delete** option is turned on for the SDS unit, and the document retention setting is set to one of the following: infinite, Event Based Retention (EBR), or chronological retention. In the case of chronological retention, deldocs will not delete a document only if the date has not yet been reached. To ignore the SDS retention setting, use the **-i** option.

---

## Documents in Cache

The deldocs -f command behaves differently depending on two things:

- 1 Whether the class assigned to the document is set to migrate to disk, and
- 2 Whether the operator overrides this document class setting by modifying the Capture Committal Component.

### Document Class Setting

When scanned into the FileNet system, the document is assigned to a class. The document then inherits any parameter values set for the class to which it is assigned.

The System Administrator configures classes using the Database Maintenance application, available through the Application Executive (Xapex). The Database Maintenance Classes option, Define/Update Document Classes, enables the administrator to set the Migration to O.D. field to either Yes or No:

- Yes specifies that any documents associated with this class will be migrated to disk. It also flags the document as "ageable," unlocking it in cache. As long as the Capture Professional application does not override this setting before committal, deldocs does not delete the document from cache. Instead, either CSM\_daemon can remove the document from cache at an appropriate time or you can remove the document manually using the CSM\_tool.
- No specifies that any documents associated with this class will not be migrated to disk. After committal has completed successfully, the document resides in cache as a "locked" document. As long as the Capture Professional application does not override this setting

before committal, the deldocs command deletes a locked document from cache.

---

**Note** Use the CSM\_tool utility to see if a document in page cache contains a “locked” or “ageable” status.

---

### **Capture Committal Component**

Under normal conditions, the Capture Committal Component inherits the default setting for the given document class, whether Migration To O.D. is set to Yes or No. If set to Yes, Capture Professional shows Commit with Migration checked. If set to No, Capture Professional shows Commit without Migration checked. If the operator modifies the Capture Committal Component, it overrides the setting assigned to the document class.

### **Checklist**

Before you use deldocs, be aware of the following:

- You must be logged on to FileNet security (Windows Server users, logon through the Application Executive) before using deldocs.
- Even if a document is not in both the index and document databases, deldocs deletes the document.

### **Procedure**

- 1 Log on with appropriate user privileges.
- 2 Enter the deldocs command with desired options.

## Related Topics

See [“CSM\\_tool” on page 204](#) for instructions on deleting documents locked in cache.

### [“fnlogon” on page 489](#)

See the “Database Maintenance” chapter of your *System Administrator’s Handbook* for a discussion of the Enable Cataloging option.

See the “Background Job Control” chapter in your *System Administrator’s Handbook* for information about consolidating and erasing media.

To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

## del\_osvr

### Description

The del\_osvr tool permanently or temporarily removes a Storage Library server from the system. The del\_osvr tool automatically reassigns (logically moves) optical media and families to a destination Storage Library server by deleting rows in the MKF permanent database surface tables for the media on the original Storage Library server and creating those rows in the surface tables on a destination Storage Library server. In addition, del\_osvr removes records for the deleted Storage Library server from the family locator table of the MKF permanent database.

---

### CAUTION

This tool is typically used as part of a procedure and should not be used independently.

---

For a permanent deletion operation, del\_osvr begins by checking the Storage Library servers for documents that are not yet written to optical media. If it finds any, the program displays informational messages and terminates. You must take **one** of the following actions:

- Restart the Storage Library server and allow pending write requests to complete.
- Run WRT\_clean to remove pending write requests.

You can temporarily remove a Storage Library server such as when a Storage Library server must be repaired or replaced. This option eliminates the restriction that no writes be pending for the media. When you use the temporary option of del\_osvr, the Storage Library server appears as inaccessible to the Image Services system.



Image Services does not change the statistics or the high-water marks for the optical media del\_osvr is logically moving to a new destination server. The del\_osvr tool temporarily moves the optical media to the destination server by copying (not moving) MKF permanent database surface table records and flagging the media as read-only. The media is then available for read operations at the destination server. When you later add back the deleted Storage Library server, the original media statistics are still valid.

---

**Tip** Changes to the family references made by del\_osvr might not be the ones you desire. You can use Database Maintenance tools to change family information. To move optical media from one Storage Library server to another, use move\_disk.

---

If the Image Services software fails or the server is rebooted before del\_osvr completes, you can safely rerun del\_osvr.

If del\_osvr fails during execution, correct the problem and rerun del\_osvr until it successfully completes.

---

**CAUTION** Do not run any other program (such as add\_osvr) after an unsuccessful run of del\_osvr. Doing so leaves the databases in a partially converted state, producing unpredictable results.

---

## Use

Use `del_osvr` to permanently or temporarily delete a Storage Library server from the system. You can also use the tool to move optical storage media from one Storage Library server to another.

If you do not specify the `-t` option, `del_osvr` checks the Storage Library servers for documents that are not yet written to optical storage media. If it finds any, the program displays informational messages and terminates. Either restart the Storage Library server and allow outstanding write requests to complete or run `WRT_clean` to remove them.

You might typically use this tool as part of a procedure that includes using the `add_osvr` and `move_disk` tools. You shouldn't use the `del_osvr` tool independently.

### Typical Scenario

For detailed procedures, see [“Procedure” on page 293](#). The following procedure represents a typical scenario in which a Storage Library server is deleted, then added back. Finally, the media is restored to its original storage library.

- 1 Run `del_osvr` to delete storage library 3 and logically move its media to storage library 4:

```
del_osvr 3 4
```

- 2 Add storage library 3 back to the system. (`add_osvr` does not move media.)

```
add_osvr 3
```

- 3 For each medium originally in 3, run `move_disk` to restore the medium, identified by `<surf_id>`, to storage library 3:

```
move_disk <surf_id> 3
```

## Syntax

```
del_osvr [-t] <svrid1> <svrid2> ... <svridn> <dest_server_num>
```

**-t** Temporarily delete the Storage Library server

**<svrid1> <svrid2>**  
**... <svridn>** Storage Library server IDs for the Storage Library servers to be deleted

**<dest\_server\_num>** Destination Storage Library server to receive information from the deleted Storage Library server

## Checklist

Before you use del\_osvr, be aware of the following:

- Before running del\_osvr, you must shut down document services.
- Before running del\_osvr, ensure that FileNet software is up to the point where NCH is started and MKF permanent and transient databases are accessible on all Storage Library servers.
- The del\_osvr tool automatically reassigns optical storage media and families to the destination Storage Library server.
- You cannot add a partially full optical disk to the destination family's surface array if the array is full. If it detects this condition, del\_osvr logs a message to the system event log and continues.
- The del\_osvr tool terminates if, during a permanent storage library delete operation, del\_osvr finds documents that are not yet written to optical storage media. You must either allow outstanding write requests to complete or remove them before re-running del\_osvr.

## Procedure

Perform the following steps to permanently delete a Storage Library server:

- 1** Back up the system to tape.

If an unrecoverable failure occurs before `del_osvr` completes successfully, restoring the backup is the only way to return the system to its original state.

- 2** Eject all media from the Storage Library server you want to delete.

In most cases, you would physically eject the media from the storage library before you start `del_osvr`. However, you could choose to skip this step if both of the following conditions are met:

- The storage library you are deleting is to be unavailable for only a short time, then be added back.
- You can tolerate the optical media being inaccessible during the period the storage library is unavailable.

- 3** Ensure all Storage Library servers are in single-user mode and that Document Services is shut down.
- 4** Ensure that the system is up to the point where NCH is started and MKF permanent and transient databases are accessible on all Storage Library servers.
- 5** Run `del_osvr`. The program updates permanent and transient MKF databases.
- 6** Shut down the Storage Library server and reboot the system.

When del\_osvr completes, shutdown the Storage Library server and reboot the system. Reboot all Storage Library servers to multi-user mode, including the Storage Library server you just deleted.

- 7 Delete the Storage Library server from the configuration using the System Configuration Editor.

As an option, you can run the move\_disk tool to move optical media to other servers. You can also run Database Maintenance tasks to reassign family references.

## Related Topics

[“add\\_osvr” on page 126](#)

[“move\\_disk” on page 826](#)

[“WRT\\_clean” on page 1341](#)

See the “Database Maintenance” chapter of the *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

# dialout

## Description

### UNIX

The dialout tool is available only on a UNIX-based Image Services server.

The dialout tool connects a FileNet server to a serial port. You can run dialout on any FileNet server equipped with a modem.

The following table describes the serial port assignment for the supported FileNet platforms.

Platform	Serial Port Assignment	Description
RS/6000®	S1 or S2	S1 port defined as /dev/tty0 S2 port defined as /dev/tty1
HP 9000	Modem serial port	Defined through SAM Default definition is server dependent. See your HP documentation.
Sun SparcStation	See your server documentation for modem setup information.	

Use `fn_edit` (the FileNet System Configuration Editor) to define port configurations for servers in a UNIX environment.

A `getty` must be available before you start dialout. Use the following command to check for the presence of a `getty`:

```
ps -ef | grep getty
```

If a `getty` is not enabled, reset the port before you run dialout.

## Use

Use dialout to connect your server to a remote system or to test your dial-up connection to a remote system.

When dialout starts, default values display for making a remote modem connection. To change these values, enter a new value after the colon as each option setting displays.

If you dial a particular site frequently, you can set up dialout to use information in the `/usr/lib/uucp/Systems` file to connect to that site. To use this feature, add a line to the `Systems` file to include the name and phone number of the site. Then enter the site name when dialout prompts you for Remote System or Phone Number.



Limited help is available by specifying the `-h` option when you start dialout. If you use the `kermit` option, additional commands display when you enter a question mark (`?`) at the `kermit` prompt, as shown below.

```
C-Kermit>? Command, one of the following:
ask          askq          assign          bug
bye          cd              clear          close
comment     connect        declare       decrement
define      delete         dial          directory
disable     do             echo          enable
end         exit          finish        for
get         getok         goto         hangup
help        if           increment     input
log         mail         msend        open
output      pause       print        push
pwd         quit        read         receive
redial     reinput     remote       rename
return     run         script       send
server     set         show         space
statistics stop        suspend     take
telnet     translate  transmit     type
version    wait        while        who
write      xif

or one of the tokens '!#;:@'
C-Kermit>
```

## Syntax

dialout [-k] [-h]

- k Use kermi to connect to the modem and dial out

---

**Note** kermi is not required to connect to the modem.

---

- h Display help information

## Sample Output

The sample below shows the default version of dialout:

```
corona> dialout
Default values are printed inside of `[ ]'.
Accept the default, type <Enter>.
Line [/dev/tty0]:
Speed [9600]:
Remote System or Phone Number (e.g. 9=18006330850) [tac_cronos]:

~.: To quit dialout

Hayes Modem Compatible Commands:
at<cr>: Attention to modem to accept commands
atdt<phone no.><cr> : To dial phone number of remote station
  use atdp for pulse dial
ath<cr>: To hang up modem
+++ : Force modem from on line to off line
WARNING:The above commands are supported by most modems. For modems
not supporting these commands, please refer to the appropriate
manual for specific modem commands.

connecting at 9600 baud -- please wait
Connected
```

Using the kermit option of dialout provides some additional commands to disconnect from the dialout session. The sample below shows dialout using the kermit option:

```
corona> dialout -k
```

```
Default values are printed inside of '['].
```

```
To accept the default, type <Enter>.
```

```
Line [/dev/tty0]:
```

```
Speed [9600]:
```

```
c                : To connect modem  
^\c              : To exit connection  
quit             : To quit dialout (kermit option)
```

```
Hayes Modem Compatible Commands:
```

```
at<cr>           : Attention to modem to accept commands  
atdt<phone no.><cr> : To dial phone number of remote station  
                    use atdp for pulse dial  
ath<cr>         : To hang up modem  
+++             : Force moden from on line to off line
```

```
WARNING: The above commands are supported by most modems. For modems  
         not supporting these commands, please refer to the appropriate  
         manual for specific modem commands.
```

```
connecting at 9600 baud -- please wait
```

```
C-Kermit 5A(179) BETA, 8 Feb 92, IBM RS/6000 (AIX 3.x)
```

```
Type ? or HELP for help
```

```
C-Kermit>
```

## Checklist

Before you use dialout, be aware of the following:

- You can run dialout on any server equipped with a modem.
- You can use the following command to check for a getty:

```
ps -ef | grep getty
```

If you do not find a getty, reset the serial port.

- The default configuration supports most modems. For other modems, refer to your modem documentation.
- You can set up dialout to use information in the `/usr/lib/uucp/Systems` file to connect to a frequently-dialed site. See [\*\*“Description” on page 295.\*\*](#)

## Procedure

- 1 Start dialout by entering the appropriate command from the list below:

**dialout>** to use dialout with default options

**dialout -k>** to use kermit to connect to the modem and dial out

When you use the -k flag, the kermit prompt (C-Kermit>) displays. Enter a question mark (?) to display a list of kermit commands or enter **help** followed by the name of a command to display command-specific help.

- 2 If you do not want to use default settings, enter new values followed by a carriage return at the prompt for each setting you want to change.
- 3 When your session is complete, log off the modem.

Use ~. (tilde, period) or other appropriate command for your environment to log off the modem. This command returns you to your host system, completes the dialout script, and issues a connection closed message.

## Related Topics

**[“kermit” on page 650](#)**

See your operating system manuals for a description of getty.

# docchk

## Description

The docchk tool inspects a specified set of document pages on a storage media for image corruptions. The program first checks the primary copy of each document and inspects header information, such as document headers, page headers, and band headers and reports any errors. The program next checks the header information on the transaction log copy of each document. The program then compares checksums for the primary and transaction log pages. Checksums that do not match indicate a potential image corruption problem. The program writes an output file listing the total number of documents and pages that were checked and the total number that were mismatched (“bad”). During the image verification, docchk decompresses the document image to check the image integrity.

The docchk tool only recognizes the following six compressed image types:

- Original FileNet Proprietary Image Format (FILENET)
- Computer Aided Logistics Services (CALIS)
- Tagged Image File Format (TIFF)
- Image Business Systems (IBS)
- Distributed Image Systems (DIS)
- Kodak Image Management System (KIMS)

Any other image types are skipped for image decompression checking and counted as illegal source image types and reported at the end of the output file.

For TIFF or CALS images that have been committed with checksumming enabled, docchk verifies that the checksums are correct.

The docchk tool saves temporary information about document locations and sizes in the /tmp directory. You can determine the amount of /tmp space required with the following formula:

$$(168 * \text{\#documents}) + (8 * \text{\#pages}) = \text{\#bytes of space required}$$

The program runs as a series of phases: scan, sort, check primary documents, and check tranlog (transaction log) documents. If you specify the -c option, docchk compares the primary documents and tranlog documents. Since this method causes more disk swaps and possible performance degradation, use it only if you need to check small sets of documents.

When mismatched pages are found, docchk writes error messages and summary information to a user-specified output file. Error messages have prefixes of **ERROR** followed by 80 bytes of identity information. The identify information includes the document ID, page number, and surface ID. Error text follows the prefix.

Summary information indicates the total number of documents and pages checked, and the total number that were bad. The report also reports the **run time**, the amount of time docchk took to execute. After the run time message, if an unrecognized image type was encountered in the check, a warning message displays the number of illegal source image types.



---

## Use

The docchk tool checks data (stored on storage media) that is associated with page format documents on a FileNet system against tranlog copies of the data. Use docchk to examine documents which could have been corrupted during transfer to media.

## Syntax

```
docchk [-f<firstdoc>] [-l<lastdoc>] [-s<surfid>] [-n<inplist>] [-b]
[-c] [-h] [-i] [-d] [-m] [-p] [-t] [-u] [-x<#mismatch>] [-y<compsz>]
[-z<decompsz>] <outfilename>
```

- f<firstdoc>** Identifies document ID of the first document to check (default is 0).
- l<lastdoc>** Identifies document ID of the last document to check (default is 4 billion, which is the maximum document ID).
- s<surfid>** Checks only those documents on surface <surfid>. For documents in a given range, check only those with the specified surface ID.
- n<inplist>** Identifies the input file name containing document IDs. Invalid if you specify -s, -l, or -f options. Only documents listed in this text file are checked. Use a text editor to change the file contents.
- b** Turns off large buffering. When the -b option is specified, docchk reads each page as a separate I/O. If you specify this option, docchk does not read unwritten or bad areas on the storage media. Use this option to avoid the overhead of disk error handling when a high number of disk errors have occurred on the system.
- c** Compares the tranlog copy of the document against the primary copy of the document. A hexadecimal report describes any data that differs

---

between the two compared copies of the page. Requires two drives. Mutually exclusive with the -p or -t options.

- h** Runs docchk at high priority (default is low priority).
- i** Skips page inspection. Reports only errors found in document headers, media I/Os, or compares.
- d** Skips decompression check of image data. Does not detect decompression errors, but reports errors found in the image and band headers.
- m** Enables the map function. If specified, does not read unwritten sectors on storage media.
- p** Checks primary copy only (default is to check both primary and transaction log copies)
- t** Checks tranlog copy only (default is to check both primary and transaction log copies)
- u** If a primary or transient document is bad, updates the document status in DOCS table with the information that either the primary or transient document is bad. Therefore subsequent retrievals always select the alternative good copy of the document, if any.
- x<#mismatch>** Specifies the maximum number of mismatches of image per block when the -c option is specified. This option is used for capping the docchk comparison time. The -x option is only valid when used with the -c option.
- y<compsz>** Specifies the maximum size in kilobytes of the largest compressed image (default=512, minimum=512, maximum=8192). If <compsz> is less than 512, the system uses the default value of 512 KB.

- z<decompsz>** Specifies the maximum size in kilobytes of the largest decompressed image (default=512, minimum=512, maximum=8192). If <decompsz> is less than 512, the system uses the default value of 512 KB.
- <outfilename>** Identifies the name of file to which docchk directs output.

## Sample Output

The following docchk command returns summary information to the output file specified as outfile1:

```
docchk -s4080 outfile1
```

The output file lists the following sample summary information.

```
** Program done.
** #docs.....10           #pages.....20
** #bad_primary_docs..0     #bad_primary_pages...0
** #primary_images...10     #bad_primary_images..0
** #bad_tranlog_docs..0     #bad_tranlog_pages...0
** #tranlog_images....0     #bad_tranlog_images..0
** Run time (hh:mm:ss) = 00:00:33
** Warning: Number of Illegal Source Image Type = 2
```

A brief explanation of all fields in the output file follows below:

**#docs** The number of documents checked.

### **#pages**

The number of pages checked. The document header is page 0. The minimum number of pages in a document is two—the document header (page 0) and one data page.

**#bad\_primary\_docs**

The number of documents that are bad on primary media.

**#bad\_primary\_pages**

The number of pages that are bad on primary media, including page 0 errors. If page 0 is bad, docchk might not be able to locate the remainder of a document to inspect it.

**#primary\_images**

The number of bit-mapped pages on primary media. This counter is listed separately because better error checking can be performed on images.

**#bad\_primary\_images**

The number of images on primary media that do not decompress correctly.

**#bad\_tranlog\_docs**

The number of documents that are bad on transaction log media.

**#bad\_tranlog\_pages**

The number of pages that are bad on transaction log media.

**#tranlog\_images**

The number of pages on transaction log media that are bit-mapped images.

**#bad\_tranlog\_images**

The number of images on transaction log media that do not decompress correctly.

## Checklist

Before you use docchk, be aware of the following:

- If you have a very large number of documents to check, be sure that enough space is available in /tmp for docchk to store information during execution.
- If a system failure or reboot occurs while docchk is running, docchk does not automatically restart when you restart the system.
- A check of a large number of documents could take considerable time. Consider breaking up the task into multiple docchk runs using the -f, -l, and -n options.
- The following signals affect program operation.

### AIX

On an AIX/6000 platform:

kill -30 <pid> causes docchk to report status

kill -31 <pid> causes docchk to abort

### HPUX

### SOL

On an HP-UX or Solaris Operating Environment platform:

kill -16 <pid> causes docchk to report status

kill -17 <pid> causes docchk to abort

where <pid> is the docchk process ID.

An example of the status reported resembles the following:

```
Check primary docs, # processed: 0, total: 30, errors: 0
```

## Procedure

No specific procedure is required.

# doccnt

## Description

The doccnt tool scans the DOCS table of the MKF permanent database and counts the number of documents on each surface. It then verifies that the value in num\_act\_docs in both the surf\_dyn\_info and surf\_locator tables in the permanent database agree.

If the values do not agree, doccnt takes action based on options you specify. If you use the -setodt option, doccnt uses the value calculated from scanning the DOCS table to correct the num\_act\_docs field in both tables. However, if you use the -list option, doccnt writes its results to standard output upon completion. No correction is attempted.

## Use

Use doccnt tool when discrepancies in document counts occur due to events, such as a system crash. The doccnt tool counts documents and verifies the count against values in the locator tables. You can run the -list option first to determine whether discrepancies exist. Then use the -setodt option to update the surf\_dyn\_info and surf\_locator tables.

In order to run the doccnt tool, you must first shutdown the FileNet software with an **initfnsw -y stop**, and then startup the MKF and relational databases with an **fn\_util startdb**.

---

**Note** Running doccnt with the FileNet software up results in the following message:

This program may not be run when document services is up

## Syntax

doccnt [ -setodt | -list ]

- setodt** Updates num\_act\_docs value in the MKF permanent database tables, surf\_dyn\_info and surface\_locator.
- list** Writes results of update operation to standard output when the program completes

---

**Note** The **num\_act\_docs** value refers to the number of active documents on a given surface. An active surface is the primary surface and the active tranlog. Images on either of these active surfaces can be directly accessed by IS. The num\_act\_docs value for the primary surface and the active tranlog will be increased after a successful image committal.

Alternate (or secondary) tranlogs are not considered active. Images stored in these alternate tranlogs must be imported into the system in order to access these images. If your system is set up to use multiple tranlogs, only one of them contains active documents and the last tranlog selected is considered to be the active tranlog. The num\_act\_docs value for any alternate tranlog(s) is not increased after a successful image committal.

---

## Example

This example requests the output be sent as a listing to standard output.

```
corona(root)/> doccnt -list
```

## Checklist

Before you use doccnt, be aware of the following:

- This tool locks the MKF permanent database for its exclusive use until doccnt completes.
- When doccnt is running in the **foreground**, the Control+c key sequence aborts doccnt.
- When doccnt is running in the **background**, you can abort the process using the method described for your operating system.

### AIX

- From the UNIX command line, enter these commands:

kill -30 <pid> (doccnt reports status of process)

kill -31 <pid> (doccnt aborts the process)

where <pid> is the doccnt process ID.

### HPUX

### SOL

- From the UNIX command line, enter these commands:

kill -16 <pid> (doccnt reports status of process)

kill -17 <pid> (doccnt aborts the process)

where <pid> is the doccnt process ID.

### WIN

- From the Windows Task Manager, select the process you want to abort, click the right mouse button, and choose the End Task option.



## Procedure

The following steps are recommended:

- 1 As fnsf user, logon to the Document Locator server.
- 2 Shutdown the FileNet software by entering:  
  
**initfnsf -y stop**
- 3 Start the MKF and relational databases by entering:  
  
**fn\_util startdb**
- 4 Run doccnt with the desired option(s).
- 5 Once you have finished running doccnt, start the FileNet software by entering:

**initfnsf restart**

---

## DOC\_dumpfb

### Description

DOC\_dumpfb provides information on fast batch objects in cache. This information includes document IDs, cache locations, and hexadecimal dumps. You might use fast batch for high-volume committal procedures when you need to reduce the time between entering documents and committing them. The Computer Output to Laser Disk 2 (COLD 2) application uses fast batch committals exclusively.

---

**Note** You must enable fast batch committal capability when you configure your FileNet system.

---

### Use

Use DOC\_dumpfb to obtain information about fast batch objects in cache.

### Syntax

DOC\_dumpfb <flag> <objectid>

**<flag>** View the specified object ID. If you don't specify any flags, it prints only the document IDs for the documents in the fast batch. See [“Flags” on page 315](#) for flag descriptions.

**<objectid>** The object ID of the fast batch for which you want information. This ID is always above 4 billion (4000000000). Cache object IDs under 4 billion are not fast batch objects. Use CSM\_tool listobjects to obtain the object ID for your fast batch.

You must specify an object ID.

## Flags

- p** Displays summary information for the fast batch and a hexadecimal dump of each object data in the fast batch.
- h** Displays attributes, such as document class and size, for each document in the fast batch.
- s** Displays summary information for each page in the batch.
- e** Displays summary information for each sector in the batch.
- d** Displays summary information for the fast batch and document IDs for all documents in the fast batch.
- o** Displays the storage location information.

## Sample Output

This section contains sample output of each DOC\_dumpfb flag and a brief explanation of the output.

### DOC\_dumpfb -p

The -p flag lists summary information and a hexadecimal dump for each fast batch object, as shown in the following sample.

The following example shows the first portion of output for doc\_id 106015.

```
corona(root)/> DOC_dumpfb -p 416000000
Dump of fast batch object with id=416000000, size=406676
Cacheloc_offset=400384, odloc_offset=403456, #pages=200, has_chksum=0
-----
Page and doc header dump
-----
Doc=106015 docpage=1 batchpage=0 offset=0 #bytes=842
-----Page sector 0-----
0000: c5000000 0000034a 00000060 00000000 000002ea 00000060 00000000 00000060
0020: 00000000 00000000 00000000 00000000 54414c4b 00000002 00000000 00000000
0040: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0060: 120e3230 300a176c 61736572 2f4d6f6e 6f2e3130 300a0d0e 30302e30 3030492c
0080: 0e302c0e 30302e31 3235492c 0e300a0a 0a0a0a0a 0a0a0a0a 0a202020 20202020
00a0: 20205452 4156454c 4c455253 20434845 51554520 41444a55 53544d45 4e540a20
00c0: 20202020 20202020 432f4f20 42424e59 204c414b 45205355 43434553 530a2020
00e0: 20202020 20202052 45434f52 44532041 4e442041 444a5553 544d454e 54532044
```

The top portion of the output shows the ID of the fast batch object (416000000) and its size in bytes (406676). Summary information is followed by a hexadecimal dump of the data associated with each document ID in the fast batch. Fast batches contain multiple documents, so you see a hexadecimal dump for each document ID in the fast batch.

**DOC\_dumpfb -h**

The -h flag displays the attributes of each document in the fast batch.  
The following is an example of output:

```
corona(root)/> DOC_dumpfb -h 4160000000
Dump of fast batch object with id=4160000000, size=406676
Cacheloc_offset=400384, odloc_offset=403456, #pages=200, has_chksum=0
-----
Page and doc header dump
-----
Doc hdr, batchpage=1
Doc hdr at sector: 1, offset: 0x000, len: 164
check_word....0xad0cde5c  format.....1
num_pages.....1  total_sectors.....2
doc_id.....106015  system_id.....10000
doc_location.....0  alt_surf_ssn.....0
annot_base_doc.....0  alt_doc_loc.....0
annot_page_num.....0  alt_surf_id.....0
doc_type.....'1'  key_info_len.....43
class_name_len.....10  sec_rd_nm_len.....9
sec_wrt_nm_len.....9  sec_exe_nm_len.....9
contig_sectors.....0  back_contig.....0
checksums.....0  character_set.....2
cluster_id.....0x0000000000000000
class_name.....'PartsList'
sec_rd_name.....'(ANYONE) '
sec_wrt_name.....'(ANYONE) '
sec_exe_name.....'(ANYONE) '
...
```

The summary information for the specified object is followed by detailed information about the object's document class and storage location. In the example, the document class name for document ID 106015 is PartsList. The document has two index keys (PartNum and Description).

**DOC\_dumpfb -s**

The -s flag displays summary information about each document in the fast batch. The following sample displays information for document IDs from 106015 to 106019:

```
corona(root)/> DOC_dumpfb -s 4160000000
Dump of fast batch object with id=4160000000, size=406676
Cacheloc_offset=400384, odloc_offset=403456, #pages=200, has_chksum=0
-----
Summary listing for all pages
-----
Doc=106015 docpage=1 batchpage=0 offset=0 #bytes=842
      sector 0 data: c5000000 0000034a 00000060 00000000
Doc=106015 docpage=0 batchpage=1 offset=1024 #bytes=164
      data: 00010000 ad0cde5c 00000001 00000002
Doc=106016 docpage=1 batchpage=2 offset=1188 #bytes=2804
      sector 0 data: c5000000 00000af4 00000060 00000000
Doc=106016 docpage=0 batchpage=3 offset=4096 #bytes=164
      data: 00010000 ad0cde5c 00000001 00000004
Doc=106017 docpage=1 batchpage=4 offset=4260 #bytes=3830
      sector 0 data: c5000000 00000ef6 00000060 00000000
Doc=106017 docpage=0 batchpage=5 offset=8192 #bytes=164
      data: 00010000 ad0cde5c 00000001 00000005
Doc=106018 docpage=1 batchpage=6 offset=8356 #bytes=995
      sector 0 data: c5000000 000003e3 00000060 00000000
Doc=106018 docpage=0 batchpage=7 offset=10240 #bytes=164
      data: 00010000 ad0cde5c 00000001 00000003
```

As shown in the above example, the output provides the following information for each document (Doc):

**docpage**

Page of the document

**batchpage**

Page within the entire fast batch

**offset**

Location of the page in cache

**#bytes**

Size of the page in bytes

**data**

Hexadecimal list of the first four addresses of the document

**DOC\_dumpfb -e**

The -e flag is similar to the -s flag. However, the -e flag provides summary information for each sector (rather than for each document) in the fast batch. The following is a sample of output:

```
corona(root)/> DOC_dumpfb -e 4160000000
Dump of fast batch object with id=4160000000, size=406676
Cacheloc_offset=400384, odloc_offset=403456, #pages=200, has_chksum=0
-----
Summary listing for all pages
-----
Doc=106015 docpage=1 batchpage=0 offset=0 #bytes=842
      sector 0 data: c5000000 0000034a 00000060 00000000
Doc=106015 docpage=0 batchpage=1 offset=1024 #bytes=164
      data: 00010000 ad0cde5c 00000001 00000002
Doc=106016 docpage=1 batchpage=2 offset=1188 #bytes=2804
      sector 0 data: c5000000 00000af4 00000060 00000000
      sector 1 data: 20202020 20202020 20202020 20202020
      sector 2 data: 414e2041 2052414d 4f532020 2f2f5452
Doc=106016 docpage=0 batchpage=3 offset=4096 #bytes=164
      data: 00010000 ad0cde5c 00000001 00000004
Doc=106017 docpage=1 batchpage=4 offset=4260 #bytes=3830
      sector 0 data: c5000000 00000ef6 00000060 00000000
      sector 1 data: 44202020 20202020 20202020 20202020
      sector 2 data: 20202020 20373737 37303030 3533204e
      sector 3 data: 312c3130 302c3030 302e3030 0a202020
```



The output is divided into rows by document ID. The first line of each row contains summary information (see the description of the `-s` flag for details). After summary information, hexadecimal addresses for the first row of each sector are listed.

### **DOC\_dumpfb -d**

The `-d` flag displays summary information for the entire fast batch followed by a list of document IDs for all documents in the fast batch as shown below:

```
corona(root)/> DOC_dumpfb -d 4160000000
Dump of fast batch object with id=4160000000, size=406676
Cacheloc_offset=400384, odloc_offset=403456, #pages=200, has_chksum=0
Documents in batch:
106015
106016
106017
106018
106019
106020
106021
106022
106023
106024
106025
106026
106027
106028
106029
106030
```

The fast batch summary information includes the following:

**size**

Size of the fast batch in bytes

**cacheloc\_offset**

Location of the object in cache

**odloc\_offset**

Location of the object on storage media

**#pages**

Number of pages in the fast batch

**has\_chksum**

Current setting of error checking

**DOC\_dumpfb -o**

The -o flag lists storage location information for the fast batch. The following is an output sample:

```
corona(root)/> DOC_dumpfb -o 4160000000
Dump of fast batch object with id=4160000000, size=406676
Cacheloc_offset=400384, odloc_offset=403456, #pages=200, has_chksum=0
-----
Optical disk location array dump
-----
iteration1: 0, iteration2: 3111006483
pages_written: 0, cur_fam: 0, last_surf 0
```

Summary information for the fast batch is followed by the fast batch location on storage media. (See the -d flag description for summary information.)

## Checklist

Before you use DOC\_dumpfb, be aware of the following:

- The system must contain fast batch objects (IDs above 4000000000).
- You must know the fast batch object ID for which you want information. Use CSM\_tool listobjects to locate object IDs.

## Procedure

The following steps are recommended:

- 1 Use **CSM\_tool listobjects** to find object IDs for fast batches in cache.
- 2 Enter **DOC\_dumpfb** at the command line, specifying a flag and appropriate object ID.

## Related Topics

[\*\*“CSM\\_tool” on page 204\*\*](#)

[\*\*“DOC\\_tool” on page 326\*\*](#)

See the FileNet system configuration tool (System Configuration Editor or fn\_edit) online help for information on defining fast batch committal capability.

# docfetch

## Description

The docfetch tool retrieves an image into page cache. You can retrieve the image immediately using synchronous retrieval or defer the retrieval to a later time. When you specify synchronous retrieval, the docfetch tool retrieves the image immediately and you cannot enter additional commands from the command line until the retrieval completes. If you prefer asynchronous retrieval, docfetch queues the retrieval request and the actual retrieval takes place at a later time.

This tool writes most docfetch error messages to the standard output device, usually the window from which you run the tool. However, it writes a few error messages only to the system error log. If you suspect a problem, check the system error log for messages.

## Use

Use docfetch to retrieve an image into page cache. For example, for diagnostic purposes, you might need to retrieve an image that is no longer in cache. Once the image is in page cache, you can use CSM\_ tool to examine the image or use a system copy or dump utility to place the image in a file for further examination.

## Syntax

```
docfetch [ -s ] <doc_id>
```

- s** Synchronously retrieves an image into page cache. If you do not specify this option, this tool queues the retrieval to perform at a later time.

**<doc\_id>** Identifies the document to retrieve into page cache.

## Example

This example retrieves document ID 1145546 into page cache.

```
corona> docfetch 1145546  
Doc id 1145546 is retrieved into the page cache.
```

## Checklist

Before you use docfetch, be aware of the following:

- You must run docfetch on the Storage Library server.
- To retrieve an image into page cache, you must be logged on as a user who has access privileges to the document you are trying to retrieve.

## Procedure

No specific procedure is required.

## Related Topics

**[“CSM\\_tool” on page 204](#)**

# DOC\_tool

## Description

DOC\_tool provides statistics on media and slots in a storage library or FileNet OSAR. After specifying the storage library for which you want information, you can view information on all drives, slots, and read/write requests for that library.

DOC\_tool provides many of the same options included in the Storage Library Control program (SLC). However, DOC\_tool only displays information. To make changes, you must use SLC.

## Use

Use DOC\_tool to examine the status of a storage library or to troubleshoot Storage Library server problems.

Enter **DOC\_tool** at the system command line. The DOC\_tool prompt displays:

```
<DOC_tool:lib a>
```

---

**Note** If the server has a storage library, **DOC\_tool** will find the first library (**lib a** in the above example) and information from DOC\_tool will apply to the library shown in the prompt. If the <DOC\_tool> prompt displays without a library, the tool could not find a library on the server. If this is the case, a library will need to be set to talk using **“SETLibr” on page 371**.

---

From the DOC\_tool prompt, enter commands to perform functions and tasks. See the **“Commands” on page 327** for command descriptions. Some commands require special permission. To run these commands,

you must enter the Allowupdates command and supply a password. You can obtain the password from your service representative.

## Syntax

### DOC\_tool

## Commands

DOC\_tool commands are listed in alphabetic order.

---

**Note** Some commands do not display in the help menu if you have not used Allowupdates and provided a password. Commands that require Allowupdates are noted in the command description.

---

UPPERCASE letters indicate the minimum number of characters that you can enter to obtain output from a given command.

## ACtsurfs

Lists the surfaces that currently have outstanding requests and the types of requests. The output from this command is similar to that of the surfsum command. The difference is that actsurfs lists only sur-

faces with outstanding read or write requests. The following is an example of actsurfs output:

```

9087: a in_drive  0                               W=150
#xfer: 0
9088: a in_slot   34                               W=72
9069: a in_slot   27                               R1=2
9004:  in_gripper                               R1=2
9056:  not_in_osar                               R1=1
Total of 5 surface records

The first column lists the surface id

```

See [“Surfsum” on page 374](#) for descriptions of the information in act-surf output.

## ALLOWUPDATES

Establishes correct permissions for running certain commands.

## ARMLog

Displays the arm history log of storage library commands issued and storage library responses received on a particular storage library. The following is an example of output from the ARMLog option:

```

ARM Command?Status log:
ESIO IE  MS  RDY  ESgrp
ESgrp ESio  ESgrp ESio  MV s1->io

```

## BACKupmode

This command is used to set/reset MSAR libraries in Backup mode.



The following are Backup mode-specific anomalies:

- An MSAR library can only be placed into Backup mode if any of the drives with surfaces are NOT exclusively reserved by an oddump.
- On a system with Multiple MSAR libraries, if some of the libraries are in Backup mode and others are in Normal mode, inserting media is not allowed.
- When an MSAR library is in Backup mode, write requests associated with the MSAR library will not be serviced and all MSAR surface files are opened in **ReadOnly** mode. However, read requests are still handled.
- An MSAR library will stay in Backup mode across an IS software restart.
- All annotation copy jobs started when an MSAR library is in Backup mode will wait until the library is placed in Normal mode and which time it will complete.
- All stsurfupdate jobs will wait when the target MSAR library is in Backup mode.
- The Docpurge utility will wait when the target MSAR library is in Backup mode.
- If an MSAR library is in Backup mode for an extended period of time, the page cache could become full, which could stop commital to the system. Therefore, a message will be logged in the syslog when the write critical threshold and/or lock threshold is reached.

- The Backup mode impacts the **Oddump** tool in a variety of ways. To review these changes, review [“oddump” on page 886](#).

## CEINFO

Displays shared memory management variables while exporting annotations to the FileNet P8 Content Engine (CE) in a Content Federated Services for Image Services (CFS-IS) environment.

---

**Note** Allowupdates must be invoked before running this command.

---

If you do not want to see the display in verbose mode, enter **n** at the verbose mode prompt or simply press return:

```
<DOC_tool:lib a>ceinfo
Verbose mode (y/n) [n]: : n
Number CE Info records allocated = 1
Number of active CE IS_import_agent(s) = 1
HighWaterMark Sequence numbers: seq_num1=1148661855 seq_num2=171000000

Entry=1 ce_os_id=1000 host_id='ben:1968'
  num_entries checked out=4
  first_checked out doc_id=100000
  last_checked out doc_id=100001
  elapse checkout time: 495.845053 seconds

Outstanding requests (0):
```

Enter **y** at the Verbose mode prompt to see the following additional information:

doc_id	page	annot_id	action	
100000	1	1	1	<-----action 1 = insert an annotation
100000	1	2	1	2 = export an annotation
100001	1	1	1	3 = delete an annotation
100001	1	2	1	4 = update an annotation

Outstanding requests (0):

## DEMANDS

Lists all requests made to a given media surface. You must specify the surface ID at the prompt. The output sample below indicates a number of reads and writes on surface 9069. Two high priority reads (HI\_READ count=2) have occurred. Information on each read request includes the document ID, numbers of pages read, priority level, and cache statistics associated with the request.

```
surface_id: 9069

HI_READ:   : count = 2, oldest = 709925911
  RETRIEVAL (7): time = 709925911, locked = 0
    doc_id=15421658, surf_id=9069, od_loc=169964, other_cpy_bad=0
    first_page=1, #pages=1, contig_sectors=67, back_contig=0
    next=00000000, prev=402e978c, qind=0, priority=2
    notify_ptr=40147bd8, cache_id=1, cache_ssn=2623082, pages_done=0
    alt_surf_id=0, alt_od_loc=0
  RETRIEVAL (7): time = 709925912, locked = 0
    doc_id=15421658, surf_id=9069, od_loc=169964, other_cpy_bad=0
    first_page=2, #pages=0, contig_sectors=67, back_contig=0
    next=402e8e24, prev=00000000, qind=0, priority=2
    notify_ptr=40147b90, cache_id=1, cache_ssn=2623082, pages_done=0
    alt_surf_id=0, alt_od_loc=0

MED_READ:  : count = 0, oldest = 0
WRITE Q:   : count = 0, oldest = 0
LOW_READ:  : count = 0, oldest = 0
BKG:       : count = 0, oldest = 0
```

## DEVsum

Displays the contents status and the surface ID for the gripper and each drive and slot in the storage library. You can limit the display to non-empty, enabled slots by answering **no** to each of devsum's two "print status" prompts as shown in the example below:

```
Print status of empty slots? (y/n):n
Print status of disabled slots? (y/n):n
Gripper Contents=known Surfid=9088
Drive 0 Contents=known Surfid=9087
Drive 1 Contents=known Surfid=9069
Slot 21 Contents=known Surfid=9050
Slot 28 Contents=known Surfid=9083
Slot 29 Contents=known Surfid=9035
Slot 30 Contents=known Surfid=9084
```

The contents status can be one of the following:

Status	Description
known	A known surface is in the drive. If the contents is known, the surface ID displays in the Surfid= field.
empty	The drive is empty.
reserved	The drive is reserved in the current library. A drive becomes reserved when the <b>oddump</b> program selects a particular drive.
unident	The media in the drive cannot be identified.
unlabld	The media in the drive is blank.

## DISable

Disables a slot, drive, optical library, platter (surface), or SDS unit. (Requires Allowupdates.)

---

**Note** If a site, for disaster recovery, needs to force a tranlog to be taken off daily, then the platter (p) option described below is available. To replace this surface, see also **“ENable” on page 340.**

---

Before you can disable a slot, drive, optical library, platter, or SDS unit, you must issue the Allowupdates command. After entering Allowupdates, the program prompts you for a password (supplied by your service representative).

---

**Note** Before using this command, use 'setlibr' command to specify which storage library you want to disable.

For MSAR libraries, DOC\_tool will not allow a user to disable an MSAR slot, an MSAR gripper or an MSAR library. MSAR surfaces could be disabled with the **p** for the platter option.

---

The following example shows the screen display when disabling a specific **slot**:

```
<DOC_tool:lib a>dis
Slot, drive, OpticalLibrary, Platter(surface), SDS?('s', 'd', 'o', 'p', 'SDS'):s
Starting slot number: 7
Ending slot number: 8
Message sent to scheduler
```

---

**Note** If the library being disabled is an MSAR library, there will be no **o** option. If the SDS unit is not in an ENABLED state, there will be no **SDS** option.

---

The following example shows the screen display when disabling a specific **drive**:

```
<DOC_tool:lib a>dis
Slot, drive, OpticalLibrary,or Platter(surface)?('s', 'd', 'o', or 'p'):d
Drive number: 1
Message sent to scheduler
```

To be able to disable for writes the current TRANLOG surface, the following option is provided. The following example shows the screen display when disabling a specific **platter**:

```
<DOC_tool:lib a>dis
Slot, drive, OpticalLibrary,or Platter(surface)?('s', 'd', 'o', or 'p'):p
Surface id: 5188
Both sides? (y/n): y
Disable reads and writes (no => disable just writes)? (y/n): y
Surface successfully disabled
```

The program writes corresponding messages in syslog, as shown in the following example:

```
1998/11/02 12:17:16.941 <fnsw> DOC_tool (28600) ...  
Slot 7 of StorageLib 1 disabled by DOC_tool utility  
  
1998/11/02 12:17:16.953 <fnsw> DOC_tool (28600) ...  
Slot 8 of StorageLib 1 disabled by DOC_tool utility  
  
1998/11/02 12:17:52.076 30,0,109 <fnsw> DOC_tool  
(28600) ...  
Drive 1 of StorageLib 1 disabled by DOC_tool utility
```

### SDS-specific options

**Note** It is now possible to enable a Single Document Storage (SDS) unit. The SDS Implementation provides the ability to store single documents to third-party storage units (such as EMC Centera, NetApp SnapLock, IBM DR550, etc.), and not in surfaces. When a document is being read to or written from an SDS storage unit, it will be associated with an SDS unit ID. This is in contrast to when a document is stored on an optical or MSAR surface and the document is associated with a surface ID.

When an SDS unit is disabled, new requests will not be dispatched to the SDS worker (SDS\_CSAR\_reader, for example). In fact, no new requests will be enqueued to the SDS worker until the unit is re-enabled, but any outstanding SDS requests will continue to be processed. However, outstanding requests will **not** be re-directed to surface-based storage.



This section shows the different prompts that will display at the **DISable** command when an SDS device is configured on the system.

The following example shows the screen when the selected library is an MSAR library with the SDS option:

```
<DOC_tool:lib a>dis
drive, Platter(surface), SDS?('d', 'p', 'SDS'):
Starting slot number: 7
Ending slot number: 8
Message sent to scheduler
```

The following example shows the screen when the selected library is an optical library with the SDS option:

```
<DOC_tool:lib a>dis
Slot, drive, OpticalLibrary, Platter(surface), SDS?('s', 'd', 'o', 'p', 'SDS'):
```

When the server is a Cache-only server, there is no library selected. If an SDS unit is in an **ENABLED** state, the following prompt will display for the **DISable** command:

```
SDS unit? <y=yes>
```

If the **DISable** command is invoked on a Cache-only server that does not have an SDS configured, the following message will display:

```
Function only available on system with storage libraries configured
```

If the **DISable** command is invoked on a Cache-only server that does have an SDS configured but not in an **ENABLED** state, the following message will display:

```
No storage libraries configured and no SDS units are ENABLED
Use the SDSinfo command to get more info on the SDS units
```

For more information, go to [“SDSinfo” on page 356](#).

## DMDsum

Provides a summary of all read and write requests to the storage library. Output includes read and write information for storage media in the storage library as well as request information for media in an ODU. The following is an example:

```
All disks? (y/n):y

Summary for all disks in osar:

DmdHighPriReads.....2      DmdMidPriReads..0      DmdLowPriReads...0
DmdWrites.....222      DmdBackgrounds..0      DmdTotal.....224
Identifies.....0
OduHighPriReads.....1      OduMidPriReads..0      OduLowPriReads...0
OduWrites.....0      duBackgrounds..0      OduTotal.....1
Disks with requests..4
```

## DRIVE

Provides information about a specific drive. The output for this command is the same as for the drives command.

To obtain information about all drives in a storage library, use the drives command.

## Drives

Provides information on every drive in the storage library, including media ID, the number of errors and transfers on the media, and whether or not the drive is disabled.

```
Drive 0:
Contents:      known          Disk id:      9087
Error count:   0              Mount time:   709925651
Disabled:      0              Config:       1
Reserved count: 0          Transfers done: 60
Reserved pid:  0

Drive 1:
Contents:      known          Disk id:      9069
Error count:   0              Mount time:   709925907
Disabled:      0              Config:       1
Reserved count: 0          Transfers done: 2
Reserved pid:  0
```

---

**Tip** To obtain information about a specific drive, use the DRIVE command.

---

The Reserved count field value is the number of processes reserving this particular drive. The Reserved pid field contains the ID of the process that is reserving this particular drive.

## DRIVESTATUS

Indicates whether a specific storage library drive is loaded. Specify the drive number for which you want information.

---

**Tip** To obtain a list of all storage library drives, use the drives command.

---

## EJmsar

This option will eject an MSAR (Magnetic Storage and Retrieval) surface based on a surface ID as input.

---

**Note** Allowupdates must be invoked before running this eject command.

Only surfaces in the current library will be ejected.

---

## ENable

Enables a slot, drive, optical library, platter (surface), or SDS unit. (Requires Allowupdates.)

---

**Note** If a site, for disaster recovery, needs to force a tranlog to be taken off daily, then the platter (p) option described below is available. To replace this surface, see also **[“DISable” on page 334](#)**.

---

Before you can enable a slot, drive, optical library, platter, or SDS unit, you must issue the Allowupdates command. After entering Allowupdates, the program prompts you for a password (supplied by your service representative).

---

**Note** Before using this command, use 'setlibr' command to specify which storage library you want to disable.

For MSAR libraries, DOC\_tool will not allow a user to disable an MSAR slot, an MSAR gripper or an MSAR library. MSAR surfaces could be disabled with the **p** for the platter option.

---

The following example shows the screen display when enabling a specific **slot**:

```
<DOC_tool:lib a>en
Slot, drive, OpticalLibrary, Platter(surface), SDS?('s', 'd', 'o', 'p', 'SDS'):s
Starting slot number: 7
Ending slot number: 8
Message sent to scheduler
```

---

**Note** If the library being enabled is an MSAR library, there will be no **o** option. If the SDS unit is not in a DISABLED state, there will be no **SDS** option.

---

The following example shows the screen display when enabling a specific **drive**:

```
<DOC_tool:lib a>en
Slot, drive, OpticalLibrary, or Platter(surface)?('s', 'd', 'o', or 'p'):d
Drive number: 1
Message sent to scheduler
```

To be able to enable for writes the current TRANLOG surface, the following option is provided. The following example shows the screen display when enabling a specific **platter**:

```
<DOC_tool:lib a>en
Slot, drive, OpticalLibrary, or Platter(surface)?('s', 'd', 'o', or 'p'):p
Surface id: 5188
Both sides? (y/n): y
Enable reads and writes (no => enable just reads)? (y/n): y
Enable active writes (no => nonactive)? (y/n): y
Surface successfully enabled
```

The program writes corresponding messages in syslog, as shown in the following example:

```
1998/11/02 12:17:33.523 <fnsw> DOC_tool (28600) ...
Slot 7 of StorageLib 1 enabled by DOC_tool utility

1998/11/02 12:17:33.525 <fnsw> DOC_tool (28600) ...
Slot 8 of StorageLib 1 enabled by DOC_tool utility

1998/11/02 12:18:00.253 30,0,109 <fnsw> DOC_tool
(28600) ...
Drive 1 of StorageLib 1 enabled by DOC_tool utility

1998/11/02 12:18:00.256 30,0,109 <fnsw> dsched (16292)
...
Drive 1, StorageLib b enabled
```

## SDS-specific options

**Note** It is now possible to enable an SDS unit. The SDS Implementation provides the ability to store single documents to third-party storage units, and not in surfaces. When a document is being read to or written from an SDS storage unit, it will be associated with an SDS unit ID. This is in contrast to when a document is stored on an optical or MSAR surface and the document is associated with a surface ID.

This section shows the different prompts that will display at the **ENable** command when an SDS is configured on the system.

The following example shows the screen when the selected library is an MSAR library with the SDS option:

```
<DOC_tool:lib a>en
drive, Platter(surface), SDS?('d', 'p', 'SDS'):
Starting slot number: 7
Ending slot number: 8
Message sent to scheduler
```

The following example shows the screen when the selected library is an optical library with the SDS option:

```
<DOC_tool:lib a>en
Slot, drive, OpticalLibrary, Platter(surface), SDS?('s', 'd', 'o', 'p', 'SDS'):
```

When the server is a Cache-only server, there is no library selected. If an SDS unit is in a **DISABLED** or **DISCONNECTED** state, the following prompt will display for the **ENable** command:

```
SDS unit? <y=yes>
```

If the **ENable** command is invoked on a Cache-only server that does not have an SDS configured, the following message will display:

```
Function only available on system with storage libraries configured
```

If the **ENable** command is invoked on a Cache-only server that does have an SDS configured but not in an **DISABLED** or **DISCONNECTED** state, the following message will display:

```
No storage libraries configured and no SDS units are DISABLED  
Use the SDSinfo command to get more info on the SDS units
```

For more information, go to [“SDSinfo” on page 356](#).

## GRipper

Provides summary information on the storage library gripper, including the media ID and the gripper status (disabled or enabled).

## Help or ?

Displays a help menu of DOC\_tool commands with brief description of each. By default, it includes only the commands available without using the allowupdates command. To display all available commands, you



must first enter the allowupdates command, as shown in the following example:

```
<DOC_tool:lib a>allowupdates
password? Updates allowed
<DOC_tool:lib a>?
Capitals in commands are minimum abbreviations
'LIBRStatus'...get StorageLibr status 'DRIVESTATUS'..libr drive status
'Surfsum'.....surface summary      'SURFace'.....display surface rec
'SLot'.....info about slot          'DRIVE'.....info about drive
'DEVsum'.....slot/drive summary     'DMDsum'.....libr demand summary
'Drives'.....all drive info         'ACTsurfs'.....surfaces w/requests
'DEMands'.....on a surface          'ODtinfo'.....optical disk table
'LIBRINfo'.....info about StorageLibr 'ODStats'.....optical disk stats
'DIsable'.....dis. slot/drive/libr  'ENable'.....ena. slot/drive/libr
'GRipper'.....show gripper contents 'SETLibr'.....set libr to talk to
'!'.....execute shell cmd          'SCRIPT'.....to start a script
'Quit'.....quit program             'Help','?'.....print help menu
'!!'.....repeat last cmd
```

## INSmsar

This option will either insert or incorporate an MSAR (Magnetic Storage and Retrieval) surface into an MSAR library based on the MSAR surface file input (an absolute path to a link [.lnk] or data [.dat] file). If the surface is not incorporated yet, this option will automatically incorporate it. If the surface is local or already incorporated, this option will insert it.

When an MSAR surface file is being inserted or incorporated, the checksum files and file high water mark stored in the MSAR label will be validated. If the validation fails, the MSAR surface file will not be inserted and an appropriate message will be displayed. The error messages can be found in the LSI section of the *System Messages Manual*.

---

To download IBM FileNet documentation from the IBM support page, see [“\*\*Accessing IBM FileNet Documentation\*\*” on page 21](#).

---

**Note:** Allowupdates must be invoked before running this insert command.

For incorporation of a foreign MSAR surface, a media family will also need to be selected.

---

## LIBRINfo

Provides summary information for all storage libraries attached to a system, including the number of slots, disks, and drives, the type of storage library, and the current mode. (See [“\*\*LIBRStatus\*\*” on page 347](#) for information on storage library mode.) In particular, the information concerning the number of disks helps determine how full the storage library is getting. The following is an example of LIBRINfo output:

```
StorageLibrary a: #slots=96, drives=0,1, librtype=FileNet12, mode=normal
StorageLibrary b: #slots=48, drives=0, librtype=Hitachi5, mode=normal
StorageLibrary c: #slots=64, drives=0,1, librtype=FileNet12, mode=normal
```

The following example shows the information returned for a FileNet OSAR. The Windows Server system in this example has a SCSI

---

## FileNet OSAR 50GTL and a Hewlett Packard model 40T storage library and an MSAR library attached:

```
<DOC_tool>librinfo
StorageLibrary a: #slots=32, drives=1,2, librtype=HP Autochanger, mode=normal
StorageLibrary b: #slots=50, drives=1,2, librtype=FileNet_gt, mode=normal
StorageLibrary c: #slots=128, drives= 1,2,3,4,5,6,7,8,9,10,11,12,
    librtype=MSAR, mode=normal
    disk(s) in library = 0
    MSAR surface creation directory = d:\msar\9076.dat
```

---

**Note** In the previous and following examples, the MSAR surface creation directory is in Windows format.

The UNIX format would be /fnsw/local/msar.

---

### LIBRStatus

Displays information on a particular storage library. The following is an example of output from this option:

```
Storage Library mode = normal
I/O station status = N/A (No I/O station)
Gripper content status = empty
disk(s) in library = 0
MSAR surface creation directory = d:\msar\9076.dat
```

**Storage Library mode** indicates whether the storage library is in normal, manual, disabled, or backup mode. In normal mode, the robotic arm within the storage library inserts and removes media. In manual mode, media are inserted and removed by hand. An Optical Disk Unit (ODU) is always in manual mode. In disabled mode, the

robotic arm in the storage library is disabled, but optical drives could still be read from and written to. Backup mode only applies to MSAR libraries.

**I/O station status** is the status of the storage library's input/output station. In particular, the information concerning the number of disks helps determine the how full the storage library is getting. In the example above, the I/O station is empty and the I/O station door is closed. For RapidChanger, this field displays "N/A (No I/O Station)."

**Gripper content status** is the status of the storage library gripper. In the example above, "one full" indicates the gripper is in use. For RapidChanger, this field displays "N/A (No Gripper)."

**Gripper disabled** indicates if the gripper is disabled (1) or enabled (0).

**MSAR surface creation directory** indicates the directory where MSAR surfaces files will be created. This field will only be displayed if the library is an MSAR library.

## ODtinfo

Provides information on the media surface you specify. The following is an example of output for surface ID 9069:

```
Surface ID: 9069
surface_id.....9069   fam_id.....18
next_free_sector..3374816 num_used_sects....3363666
num_act_docs.....50235 num_del_docs.....0
num_clusters.....0   old_hw.....0
num_pages.....50235  disk_status.....0
write_protect.....0  sides.....2
last_desc_sect....3257475 nxt_short_desc....3256783
num_unwrt_desc.....0
label_date.....0x2a07beb5 full_date.....0x00000000
archive_date...0x00000000 last_disk_sect....3374815
orig_ssn.....2623082  orig_surfid.....9069
short_doc_ids:    0
short_sects:     0
```

Most of this information is extracted from permanent database tables such as surf\_dyn\_info and surf\_stat\_info.

## ODStats

Displays statistics for the media you specify. Output might not appear if odstatus detects no errors.

## Quit

Exits DOC\_tool.

## READMsarlabel

Displays the volume label (sector 16) and MSAR label (sector 0). The command prompts the user for an absolute path to the MSAR surface file name (.dat). The MSAR surface does NOT need to be in the MSAR library.

If the surface is out of the library, the checksum and high water mark values will be validated. A message will be displayed to indicate if the values are valid or invalid.

For MSAR surface files that have the old format (no checksum), the MSAR label and the volume label content is displayed without doing checksum validation. This command is NOT password protected. The following is an example of readmsarlabel output:

```
<DOC_tool:lib a>readm
MSAR surface file (absolute path with extension .dat): /msar/003440.dat
VOLUME LABEL
=====
Surface_ID.....3440          family_type.....'PRIMARY'
format_level.....21         system_id.....3287
directory_extent_start....64  event_extent_start.....984
directory_extent_end.....983  event_extent_end.....1999
byte-sex.....0x1020304      character_set.....1
sector_size.....1024
date.....2002/09/09
time.....12:27:16
volume_name.....''
write_vendor_name.....'FileNet'
format_vendor_name.....'FileNet'
```

Output continued on next page.

**(<DOC\_tool:lib a>readm continued)**

MSAR LABEL

=====

```

disk_type.....21
format_level.....3
sector_size.....1024
label_name.....'MSAR'
file HWM in sectors.....4002
number of checksums.....10
chk_array[0].start_sect...3986      chk_array[1].start_sect...1
chk_array[0].num_sect....16         chk_array[1].num_sect....16
chk_array[0].cks.....414e5ab2      chk_array[1].cks.....327e1604
chk_array[2].start_sect...2000      chk_array[3].start_sect..500
chk_array[2].num_sect....16         chk_array[3].num_sect....16
chk_array[2].cks.....89949a66      chk_array[3].cks.....0
chk_array[4].start_sect...1000      chk_array[5].start_sect..1500
chk_array[4].num_sect....16         chk_array[5].num_sect....16
chk_array[4].cks.....0             chk_array[5].cks.....0
chk_array[6].start_sect...2000      chk_array[7].start_sect..2500
chk_array[6].num_sect....16         chk_array[7].num_sect....16
chk_array[6].cks.....89949a66      chk_array[7].cks.....0
chk_array[8].start_sect...3000      chk_array[9].start_sect..3500
chk_array[8].num_sect....16         chk_array[9].num_sect....16
chk_array[8].cks.....0             chk_array[9].cks.....0
Label checksum.....d45e4936

```

Not validating MSAR checksums because surface is inside the library

## SATClear

Allows the user to remove surface activity records that are no longer needed or meaningful. This option could be used if a surface was erased or consolidated. Records can be removed by date or by surface.

```
<DOC_tool>satc
Enter tracking removal date (mm/dd/yyyy) (all records prior to this date
will be removed) or <CR>, to remove tracking data by surface id.

Clear surface logging info from mm/dd/yyyy and older? (y/n):

Enter a surface id or <CR> for no surfaces:

Clear surface activity logging for surface xxx? (y/n):
```

After the first prompt above is answered, checking will be done for invalid dates. All tracking entries with a date equal to or older than the date entered will be removed. If a valid date is entered, the second prompt will display. When this is answered, the user is done.

If <CR> is entered at the first prompt, the third prompt will display and the second prompt will not appear. After the third prompt is answered, the fourth prompt will display. After the fourth prompt is answered, the user is done.



## SATRDetail

Displays all logged activity for all surfaces or for a selected surface, for any period of time that logging has been invoked. A report is generated for a time period you specify with a start date and duration of days. Any past date could be entered as the starting date of the report, but only activity from the days when logging was enabled is displayed.

```
<DOC_tool>satrd
```

```
Enter start date (mm/dd/yyyy) for report or <CR>for all days logged:  
Enter the duration of the report (in days):  
Enter a surface id or <CR> for all surfaces:
```

If you enter a start date at the first prompt, the date is verified. If you entered a valid date, the second prompt displays.

After you enter the number of days duration in the second prompt, a third prompt asks you to specify the surfaces to include in the report.

(If you press <CR> at the first prompt, SATRDetail skips the second prompt.)

Specifying a non-existent surface at the third prompt results in an empty report.

(If you press <CR> at the third prompt, SATRDetail displays logged activity for all surfaces.)

Based on a start date of 9/22/2001, a duration of 20 days, and all surfaces selected, a report similar to the following displays.

Surface Activity Summary Report					
09/22/2001 - 10/11/2001. (20 days)					
Date of report: 10/12/2001					
Surf_id	date	mounts	surface reads	surface writes	total surface transfers
5001	09/22/2001	5	100	0	100
5001	09/23/2001	3	50	2	52
5001	09/28/2001	10	81	0	81
...					
5004	09/22/2001	8	56	20	76
5004	09/24/2001	12	1000	0	1000
5004	09/29/2001	12	400	0	400

There is one report line for any daily activity for each surface during the specified duration of days. If there is no activity for a day of the report, that date does not appear for that surface.

---

**Note** The value in the **mounts** column is increased only when the dtp process recognizes a disk change in the drive. If no disk is in a drive, inserting a disk is not counted as a mount.

---

### SATRSummary

Displays the total of all activity logged for each surface or for a single selected surface for the time period specified by the user. The same user interface as described in the satrdet option applies to this option. There is one line of report data for each surface in the permanent data-

base even if no activity occurred. The activity for all days in the specified duration is be summed and reported.

A report similar to the following displays:

Surface Activity Summary Report							no
09/22/2001 - 10/11/2001. (20 days)							surf_info
Date of report: 10/12/2001							record
Surf_id	lib	mounts	reads	writes	total requests		
5001	a	8	150	2	152		
5002	a	0	0	0	0		
5003	a	0	0	0	0		
5004	a	42	1537	0	1537		
5023	NP	0	0	0	0		
5024	a	0	0	0	0		

The NP in the screen above stands for Not Present.

---

**Note** The value in the **mounts** column is increased only when the dtp process recognizes a disk change in the drive. If no disk is in a drive, inserting a disk is not counted as a mount.

---

## SCRIPT

Starts execution of a previously-written script from a file name that you specify. The program prompts you for the script name when you enter the SCRIPT command.

## SDSinfo

Allows the user to view the current state of the SDS unit and its associated SDS worker (either SDS\_CSAR\_reader or SDS\_FS\_reader) as well as performance information.

**Note** Only one SDS worker could be configured on a system, either the SDS\_CSAR\_reader or the SDS\_FS\_reader. Various SDSinfo fields are impacted and will only display the information for the configured worker.

### SDSinfo command prompt and options:

If the SDSinfo command is invoked and there are no SDS units configured on the server, the following message is displayed:

```
<DOC_tool:lib a>sdsinfo  
No SDS have been configured on this server
```

The **SDSinfo** command supports four options allowing the user to display a summary, an SDS queue, a detail worker, or an all information report, as shown below:

```
<DOC_tool:lib a>sdsinfo  
Summary info, Queued request info, Worker info, or All info?  
( 's', 'q', 'w', 'a' )
```

Basic information is displayed for all reports, regardless of which **SDSinfo** option is selected. The following table describes this basic information:

## Basic SDS Information

Field Name	Description
SDS unit number	The SDS unit will always be set to 1 in this release.
SDS state	The SDS unit could be set to a DISABLED state (via the <b>DISable</b> option of DOC_tool, to an ENABLED state (via the <b>ENable</b> option of the DOC_tool, and a DISCONNECTED state (via the SDS_worker [SDS_FS_reader] because the SDS_worker has encountered a problem making it impossible to process requests.)
SDS Priority	This can be set to high, medium, or low and this setting is extracted from the sds_conf file. This setting is for when a document is not found in cache. High priority means the document will first be looked for on the SDS unit, then in "in box" MSAR, and finally on optical. Medium priority means the documents will first be looked for in "in box" MSAR, then on the SDS unit, and last on optical. Low priority means the document will first be looked for in "in box" MSAR, then on optical and finally on the SDS unit.
SDS Worker name	The SDS Worker name uses the format SDS_<device>_reader NetApp SnapLock - SDS_FS_reader IBM DR550 - SDS_ISAR_reader EMC Centera - SDS_CSAR_reader
Number SDS worker instances	This indicates how many SDS worker instances are running. The number of worker instances is configurable through the sds_config file and the valid number instances are 1 to 99.
SDS info	The SDS Worker expects the following label: workingdirectory = <directory name> This is a maximum of 1024 characters.
SDS queue length	This displays the current number (0-n) of SDS queued requests. A high number here could be the indication of a problem.

When the Summary option is selected, the basic information as described above is displayed and the summary worker report is displayed:

The following table describes this Summary worker report information:

SDS Counter Summary Worker Information

Counter Name	Description
Requests Processed	Total number of requests that have been processed by all of the SDS worker processes.
Successful requests processed	Total number of requests that have been successfully processed.
Requests satisfied via SDS device (IBM DR550)	Total number of requests that have been processed using SDS. By subtracting the "Requests satisfied via SDS" from the "Requests Processed," the number of requests processed using the NLS working directory could be determined.
Errors	Total number of errors encountered. These are only errors that were not corrected after retries and did not result in termination of SDS worker processes.
Last_error	Last error encountered by the last SDS worker process. If no errors were encountered by SDS workers, this field is not displayed.
Images retrieved	Total number of images retrieved.
Data retrieved	Total amount of data, in K-bytes, transferred to CSM cache from the NLS working directory.
AVG request size	Average size of data transfer per requests. A standard deviation for this field is also displayed, and this could be used to gauge the other Standard Deviations to follow.
AVG image size	Average size of image page transferred.
Number of SDS device reads (API calls)	<b>Centera:</b> Total number of calls to the FPTag_BlobReadPartial API entry point. <b>IBM DR550:</b> Total number of this sequence of API calls to read a DR550 Tivoli® object.

## SDS Counter Summary Worker Information, Continued

Counter Name	Description
Number of SDS device query calls	<b>IBM DR550 Only:</b> Number of queries for an internal object ID based on the object name.
FPTag_BlobRead-Partial calls	Total number of calls to the FPTag_BlobReadPartial API entry point. *This field will <b>not</b> display if the SDS_FS_reader is configured on your system.
Number of SDS device requests where the whole document fits into the internal image_buffer (128K)	<b>Centera and IBM DR550:</b> Requests where the whole document fits into the internal buffer. The size of configurable internal buffer is displayed. <b>Centera Only:</b> The BLOB size is part of the CLIP meta-data. If the BLOB size is less than or equal to the internal buffer size, the whole BLOB will be read into the internal buffer with a single BLOB Read call. Using a single BLOB read will maximize system performance, so it is important to have DBP buffer size equal to the BLOB size if possible.
Number of SDS device requests where the requested images fit into the internal image_buffer (128K)	<b>Centera and IBM DR550:</b> Requests where all of the images fit into the internal buffer. The total size of these images is displayed. <b>Centera Only:</b> If the whole document cannot fit into the internal buffer, the header and table of contents (TOC) information will be read into the internal buffer with one BLOB read. The TOC includes the offset and size of each page of the document.
Number of SDS device requests where the requested images do not fit into the internal image_buffer (128K)	<b>Centera and IBM DR550:</b> Requests where all of the images do not fit into the internal buffer. <b>Centera Only:</b> If the requested pages do not fit in the internal buffer, the header and TOC information will be read into the internal buffer with one BLOB read, and each requested page will require at least one BLOB read. In the case where the individual page exceeds the internal buffer, multiple BLOB reads will be required per page. Only requested pages are read to the internal buffer and written to cache.
Cache hits	Number of pages already in cache when a read request for a multi-page document is executed.

The following table describes the Accumulated Average Time Summary Worker Report information, which is also displayed using the Summary option:

SDS Accumulated Average Time Summary Worker Report

Average Accumulated Times	Description
Up time	Accumulated average up time of all the SDS workers in seconds/workers.
Idle time	Accumulated average up time of all the SDS workers in seconds/worker. This is the time that workers are idle waiting for requests to be processed.
Total processing time	Total elapsed processing time per worker. The average time elapsed per request is also displayed in seconds/workers with its standard deviation. The average time elapsed per image page (seconds/image page) and elapsed per K-byte (seconds/K-byte) is also displayed.
Time in get_doc_from_file_system()	Accumulated average time in the get_doc_from_file_system() subroutine in seconds/workers. This is the accumulated time taken to execute the get_doc_from_file_system subroutine which includes extracting the requested document pages from the file system and writing them to page cache.
Time in get_doc_from_device()	<b>Centera and IBM DR550:</b> Accumulated average time in the get_doc_from_device() subroutine in seconds/workers. This is the accumulated time taken to execute the get_doc_from_device subroutine which includes extracting the requested document pages from the SDS device and writing them to page cache.



## SDS Accumulated Average Time Summary Worker Report, Continued

Average Accumulated Times	Description
AVG time in SDS device open and read (API calls)	<p>Accumulated average time doing API calls (Seconds/workers).</p> <p><b>Centera:</b> This includes operations such as Open Pool, Open Clip and BLOB reads. It is a subset of the operations performed by the get_doc_from_centera subroutine. The average time elapsed per request is also displayed (seconds/requests) with the Standard Deviation. A summed squared of each elapsed time to do the Centera API calls per requests is calculated and stored. This is done so the Standard Deviation for Centera API calls time per requests can be calculated. The average time elapsed per image page (seconds/image page) and elapsed per K-byte (seconds/K-byte) is also displayed.</p> <p><b>IBM DR550:</b> Includes the set of API calls to connect to the DR550, query session information and API version number, and the sequence of API calls to read a DR550 object.</p>
AVG time in SDS device read only (API calls)	<p><b>Centera:</b> Accumulated average time doing FileNet BLOB reads.</p> <p><b>IBM DR550:</b> Includes the set of API calls to read a DR550 object.</p> <p>For both, this is a subset of the accumulated time doing API calls. The average time elapsed per request is also displayed (seconds/requests) with the Standard Deviation. The average time elapsed per image page (seconds/image page) and elapsed per K-byte (seconds/K-byte) is also displayed.</p>
AVG time in SDS device query calls	<p><b>IBM DR550 Only:</b> The average in doing a query call to obtain an internal object ID based on the object name.</p>

## SDS Accumulated Average Time Summary Worker Report, Continued

Average Accumulated Times	Description
AVG time in Writing to page cache (CSM)	Accumulated average time doing CSM cache calls. The accumulated time includes time spent writing to cache. The average time elapsed per request is also displayed (seconds/requests) with the Standard Deviation. The average time elapsed per image page (seconds/image page) and elapsed per K-byte (seconds/K-byte) is also displayed.
AVG requests queue wait time	The average wait time in seconds per request. The wait time is the elapsed time of when a request is enqueued by the dispatch process, and the time the SDS_FS_reader dequeues the request. The wait time is accumulated and an average is displayed here. In addition, a Standard Deviation of wait time in seconds per request is also calculated and displayed.

---

**Note** The samples below display the output that would be seen when the SDS\_CSAR\_reader is configured. When the SDS\_FS\_reader is configured, none of the Centera related fields will be displayed as indicated on the previous pages with a \*.

---

## Sample output - Summary Option

```
<DOC_tool:lib a>sdsinfo
Summary info, Queued request info, Worker info, or All info?
('s', 'q', 'w', 'a'): : s

The current time is Mon Aug 28 14:37:35 2006

SDS info: SUMMARY option

***** SDS unit = 1 state = ENABLED SDS priority = medium *****
  DEBUG Setting = MAX
  Worker = 'SDS_ISAR_reader' Number Instances = 1
  info = 'WorkingDirectory=/NLS/tivoli domain=10.10.8.35'
  Current number of outstanding requests in the SDS queue: 0

Total Accumulated counters from all workers(1):
TOTAL WORKER COUNTERS:
  Requests processed = 3
  Successful requests processed = 2
  Requests satisfied via SDS device (IBM DR550): 2
  Errors = 1
  Images retrieved = 9
  Data retrieved = 0.719210MB
  AVG request size = 245.490234K (STDEV=477699.203356)
  AVG Image Size = 81.830078K
  Number of SDS device read (API calls) = 11
  Number of SDS device query calls = 0
  Number of SDS device requests where the whole document fits
  into the internal image_buffer (128K): 0
  Number of SDS device requests where the requested images fits
  into the internal image_buffer (128K): 1
  Number of SDS device requests where the requested images do not
  fit into the internal image_buffer (128K): 1
  Cache hits: 0
AVERAGE ACCUMULATED ELAPSE TIMES:
  Up time:      76152.546358 secs/workers (1269.209106 mins)
  Idle time:   76147.776404 secs/workers (1269.129607 mins) (99.99%)
(continued on next page)
```

## Sample output - Summary Option, continued

```
Total processing time: 4.752298 secs/workers
                        (1.584099 secs/reqs)
                        (0.848008 STDEV secs/reqs)
                        (0.528033 secs/image page)
                        (0.006453 secs/KB)
Time in get_doc_from_filesystem(): 0.000790 secs/workers
Time in get_doc_from_device():      4.076062 secs/workers
AVG time in SDS device open and read (API calls): 2.991772 secs/workers
                                                (0.997257 secs/reqs)
                                                (0.840033 STDEV secs/reqs)
                                                (0.332419 secs/image page)
                                                (0.004062 secs/KB)
AVG time in SDS device read only (API calls): 2.133966 secs/worker
                                                (0.711322 secs/reqs)
                                                (0.237107 secs/image page)
                                                (0.002898 secs/KB)
AVG time in SDS device query calls: 0.000000 secs/worker
AVG time in writing to page cache (CSM): 0.438663 secs/workers
                                                (0.146221 secs/reqs)
                                                (0.179214 STDEV secs/reqs)
                                                (0.048740 secs/image page)
                                                (0.000596 secs/KB)
AVG requests queue wait time: 0.247177 secs/reqs
                                (0.426779 STDEV secs/reqs)
<DOC_tool:lib a>
```

When the Queue option is selected:

All the Basic SDS information is displayed and detailed information about any outstanding SDS queued requests is displayed.

### Sample output - Queue Option

```
<DOC_tool:lib a>sdinfo
Summary info, Queued request info, Worker info, or All info?
('s', 'q', 'w', 'a'): : q

The current time is Mon Aug 28 14:42:15 2006

SDS info: QUEUED option

***** SDS unit = 1 state = ENABLED   SDS priority = high *****
DEBUG Setting = MAX
Worker = 'SDS_CSAR_reader' Number Instances = 1
info = 'WorkingDirectory=/NLS/centera emcdomain=10.10.16.1,10.10.16.2'
Current number of outstanding requests in the SDS queue: 2
Request Type: READ_REQUEST
doc_id = 104148, doc_index = 104148, ssn = 8815170,
cache_id = 1, cache_ssn = 8815170, firstpage = 1, numpages = 1,
prim_surf_id = 3014, tran_surf_id = 0
notify_ptr =5f60abf8, has_clipid = TRUE
Request Type: READ_REQUEST
doc_id = 104148, doc_index = 104148, ssn = 8815170,
cache_id = 1, cache_ssn = 8815170, firstpage = 1, numpages = 1,
prim_surf_id = 3014, tran_surf_id = 0
notify_ptr =5f60abb0, has_clipid = TRUE
<DOC_tool:lib a>
```

If there are no requests in the SDS queue, a message similar to the following will be displayed:

### Sample output - Empty Queue Option

```
<DOC_tool:lib a>sdsinfo
Summary info, Queued request info, Worker info, or All info?
('s', 'q', 'w', 'a'): : q

The current time is Mon Aug 28 14:46:53 2006

SDS info: QUEUED option

***** SDS unit = 1 state = ENABLED   SDS priority = high *****
DEBUG Setting = MAX
Worker = 'SDS_CSAR_reader' Number Instances = 4
info = 'workingdirectory=/surf/HSM_DISK emcdomain=10.10.16.1'
Current number of outstanding requests in the SDS queue: 0

SDS (unit=1) queue is currently empty
```

When the Worker option is selected:

The Basic SDS information is displayed, along with information about each active worker, and a summary worker report. The information displayed is similar to worker summary reports but the counters and time accumulates are specific to each worker. The fields displayed in the Summary report and Worker report are the same except there is an additional field, the “Last request done” field which is a time stamp of the last request done.

## Sample output - Worker Option

```
<DOC_tool:lib a>sdsinfo
Summary info, Queued request info, Worker info, or All info?
('s', 'q', 'w', 'a'): : w

The current time is Mon Aug 28 14:51:10 2006

SDS info: WORKER option

***** SDS unit = 1 state = ENABLED SDS priority = medium *****
  DEBUG Setting = MAX
  Worker = 'SDS_ISAR_reader' Number Instances = 3
  info = 'WorkingDirectory=/NLS/tivoli domain=10.10.8.35'
  Current number of outstanding requests in the SDS queue: 0

Worker instance 1:
WORKER COUNTERS:
  Requests processed: 3
  Requests satisfied via SDS device (IBM DR550): 2
  Successful requests processed: 2
  Errors: 1 (last error = ca64000a)
  Retrieved image pages count: 9
  Retrieved data: 0.719210MB
  AVG Image Size = 81.830078K
  AVG request size = 245.490234K
  Number of SDS device read (API calls): 11
  Number of SDS device query calls: 0
  Number of SDS device requests where the whole document fits
  into the internal image_buffer (128K): 0
  Number of SDS device requests where the requested images fits
  into the internal image_buffer (128K): 1
  Number of SDS device requests where the requested images do not
  fit into the internal image_buffer (128K): 1
  Cache hits: 0

(continued on next page)
```

---

**Sample output - Worker Option, continued**

```
WORKER ACCUMULATED TIMES:
Up time =          76481.966902 secs (1274.699448 mins)
Total Idle time =  76477.196948 secs (1274.619949 mins) (99.99%)
Last request done = 76355.779166 secs (1272.596319 mins)
Total processing time: 4.752298 secs
                        (1.584099 secs/reqs)
                        (0.528033 secs/image page)
                        (0.006453 secs/KB)
Time in get_doc_from_filesystem(): 0.000790 secs
Time in get_doc_from_device():    4.076062 secs
Time in SDS device open and read (API calls): 2.991772 secs
                                           (0.997257 secs/reqs)
                                           (0.332419 secs/image page)
                                           (0.004062 secs/KB)
Time in SDS device read only (API calls): 2.133966 secs
                                           (0.711322 secs/reqs)
                                           (0.237107 secs/image page)
                                           (0.002898 secs/KB)
Time in SDS device querying database: 0.000000 secs
Time in writing to page cache(CSM): 0.438663 secs
                                           (0.146221 secs/reqs)
                                           (0.048740 secs/page)
                                           (0.000596 secs/KB)
Total SDS requests queue wait: 0.741531 secs
                                (0.247177 secs/reqs)
                                (0.082392 secs/page)
                                (0.001007 secs/KB)
.
.
.
```

When the all option is selected, all the information above is displayed.



## Sample output - All Option

```
<DOC_tool:lib a>sdsinfo
Summary info, Queued request info, Worker info, or All info?
('s', 'q', 'w', 'a'): : a

The current time is Mon Aug 28 14:59:32 2006

SDS info: ALL option

***** SDS unit = 1 state = ENABLED   SDS priority = medium *****
        DEBUG Setting = MAX
        Worker = 'SDS_ISAR_reader' Number Instances = 1
        info = 'workingdirectory=/NLS/tivoli domain=10.10.8.35'
        Current number of outstanding requests in the SDS queue: 0

SDS (unit=1) queue is currently empty

Worker instance 1:
WORKER COUNTERS:
  Requests processed: 3
  Requests satisfied via SDS device (IBM DR550): 2
  Successful requests processed: 2
  Errors: 1 (last error = ca64000a)
  Retrieved image pages count: 9
  Retrieved data: 0.719210MB
  AVG Image Size = 81.830078K
  AVG request size = 245.490234K
  Number of SDS device read (API calls): 11
  Number of SDS device query calls: 0
  Number of SDS device requests where the whole document fits
    into the internal image_buffer (128K): 0
  Number of SDS device requests where the requested images fits
    into the internal image_buffer (128K): 1
  Number of SDS device requests where the requested images do not
    fit into the internal image_buffer (128K): 1
  Cache hits: 0
(continued on next page)
```

## Sample output - All Option, continued

```
WORKER ACCUMULATED TIMES:
  Up time =                76594.523156 secs (1276.575386 mins)
  Total Idle time =        76589.753202 secs (1276.495887 mins) (99.99%)
  Last request done =      76468.335420 secs (1274.472257 mins)
  Total processing time: 4.752298 secs
                        (1.584099 secs/reqs)
                        (0.528033 secs/image page)
                        (0.006453 secs/KB)
  Time in get_doc_from_filesystem(): 0.000790 secs
  Time in get_doc_from_device():      4.076062 secs
  Time in SDS device open and read (API calls): 2.991772 secs
                                                (0.997257 secs/reqs)
                                                (0.332419 secs/image page)
                                                (0.004062 secs/KB)
  Time in SDS device read only (API calls): 2.133966 secs
                                                (0.711322 secs/reqs)
                                                (0.237107 secs/image page)
                                                (0.002898 secs/KB)
  Time in SDS device querying database: 0.000000 secs
  Time in writing to page cache(CSM): 0.438663 secs
                                                (0.146221 secs/reqs)
                                                (0.048740 secs/page)
                                                (0.000596 secs/KB)
  Total SDS requests queue wait: 0.741531 secs
                                                (0.247177 secs/reqs)
                                                (0.082392 secs/page)
                                                (0.001007 secs/KB)

Total Accumulated counters from all workers(1):
TOTAL WORKER COUNTERS:
  Requests processed = 3
  Successful requests processed = 2
  Requests satisfied via SDS device (IBM DR550): 2
  Errors = 1
  last error = ca64000a
.
.
.
```

## SETLibr

Allows you to specify the storage library for which you want information. Some DOC\_tool commands display information about the default storage library, **a**, unless you first select a specific storage library with the setlibr command.

To specify a new storage library, enter **setlibr** at the DOC\_tool prompt. Specify the storage library ID at the following prompt:

```
Current Storage Library is: a
New Storage Library (a, b, ...,h):
```

If you specify an invalid storage library, the following message displays:

```
New Storage Library is not configured, current osar unchanged
```

## SLOT

Provides information on a slot within the storage library.

DOC\_tool slot first prompts you for a slot number so before using this command, use the surfsum command to display a list of possible slot numbers.

The following is an example of output for slot 35, which is entered at the prompt in the first line of the output below:

```
Slot number: 35
Slot 35:
Contents:      known           Disk id:      9018
Error count:  0                Mount time:   709841382
Disabled:     0                Config:       1
Assigned to toq: 0
```

The slot command output displays the following information:

**Contents** field displays a description of the slot contents (known, empty, unident, unlabld). Each descriptive value is described below:

Contents	Description
known	A known surface is in the slot. If the contents is known, the media ID displays in the Disk ID field.
empty	The slot is empty.
unident	The media in the slot cannot be identified.
unlabld	The media in the slot is blank.

**Error count** displays the number of failed attempts to access the slot. The slot is disabled if the error count exceeds a set limit.

**Mount time** is the number of seconds the media has been mounted in the slot.

**Disabled** is a value that describes whether or not the slot is disabled. 0 indicates the slot is enabled, 1 indicates the slot is disabled.

**Config** is the configuration status. 1 means the slot is configured. This value is always 1 for a slot.

**Assigned to toq** is a disk scheduler indicator. If the value is 1, an RSVP has been posted to eject the media in the slot and the scheduler does not attempt to swap the disk into a drive. If the value is 0, you can schedule the media for a swap.

## SURFace

Displays surface records on specific media or in a specific slot. You specify the surface ID for which you want information, as shown in the sample output below for surface ID 9069:

```
Surface ID: 9069

Surfid..9069      OrigSurfid..9069      OrigSSN.....2623082
Fam id..18       NxtAvlSect..3374816  LastDiskSect..3374815  DiskType..5
Lock....0        LockPid.....0        LockCount.....0        Gdp.....00000000

Io_hist_counts: 19 92 51 46,  time: 709925743,  index: 0

seek_depth: high=10, mid=0, low=0, seek_ascend: 1, seek_sect: 169964

Osar#...0  Location...in_gripper  Slot...0  Unavail..0  DoNotUse..0
Primary..1  From_oper...0          Drive..0  Loaded...1  WrtProt...0
MSARReadOnly 1

      HighPriRead  MedPriRead      Write  LowPriRead      Bckgrnd      Total
      -----
Count:          2          0          0          0          0          2
Time:         10:39:01   16:00:00   16:00:00   16:00:00   16:00:00

Cur time: 709925965 = 10:39:25, ODU: R1=0 R2=0 W=0 R3=0 Bkg=0 Tot=0
MSAR surface file = d:\msar\9069.dat
```

Most of these records are extracted from permanent database tables such as surf\_info, surf\_locator, and scalar\_numbers.

The MSAR information in the example above can be defined as follows:

- If **MSARReadOnly** is **1**, the surface could be read from but not written to. If **MSARReadOnly** is **0**, the surface is available for reads and writes.

- MSAR surface files are the MSAR equivalent to optical surfaces. MSAR surface files are dynamically created and allotted by the MSAR software on an as-needed basis. The field **MSAR surface file** shows the last location of the data file for the surface.

### Surfsum

Provides information on media in a storage library, including the location and the number of reads and writes scheduled to the media.

---

**Tip** Many DOC\_tool commands prompt for a media surface number or slot number. Run surfsum to locate these numbers.

---

The following example of surfsum output has been abbreviated for readability). Each column of information is described after the example:

```

9087: a in_drive 0 #Op=4 W=172 #xfer: 52
9086: a in_drive 0
9069: a in_drive 1 R1=2 #xfer: 0
9068: a in_drive 1
9085: a in_slot 30
9084: a in_slot 30
9035: a in_slot 29
9034: a in_slot 29
9005: a in_slot 34
9004: a in_slot 34
9089: a in_slot 27
9088: a in_slot 27 W=56
9076: b in_slot 3 R1=1 DoNotUse
      MSAR surface file= d:\msar\9076.dat
9096: b in_slot 2 R1=1 MSARReadOnly
      MSAR surface file= d:\msar\9096.dat
9090: not_in_libr R2=3 W=5 R3=10 Bkg=1
      MSAR surface file= d:\msar\9090.dat
9081: not_in_libr
9080: not_in_libr R1=1
...
9014: not_in_osar
9057: not_in_osar
9056: not_in_osar R1=1
...
Total of 36 surface records

```

Column 1 - media surface in the storage library. In this example, all media surface numbers are in the 9000 range.

Column 2 - the storage library in which the media surface is located and the location of the surface within that storage library. In this example, the first set of surfaces (9087 through 9088) are located in storage library **a** and are either in a drive (in\_drive) or a slot (in\_slot). Surfaces 9014, 9056, and 9057 are not in a storage library (not\_in\_osar). Library **b** is an MSAR library.

Column 3 - the drive number or slot number for the media surface. In the example, surfaces 9086 and 9087 belong to drive 0. Surfaces 9084 and 9085 belong to slot 30. Note that each drive and slot has two associated surfaces, one odd and one even.

The remaining columns display additional information such as the number of reads (of type R1, R2, or R3) and writes (W) to a surface. Not all media surfaces display this information. The fields display only when requests are generated to a certain surface. Each field is described below:

Field Label	Description
#Op	Number of times an open operation has been performed on the surface.
#xfer	Number of transfers that have occurred to the surface
R1	Number of high priority read requests performed against this surface
R2	Number of medium priority read requests performed against this surface
R3	Number of low priority read requests performed against this surface
W	Number of write requests performed against this surface



**!**

Executes a shell command

**!!**

Repeats the last command

## Checklist

Before you use DOC\_tool, be aware of the following:

- You must have a valid fnlogon session or be logged on as a member of the fnadmin group before you use this tool.
- You must run DOC\_tool from within the /fnsw/bin directory.
- Before you start DOC\_tool, the permanent database must be running.
- You must run DOC\_tool from the server to which the storage library is connected.

## Procedure

- 1 At the command line, enter **DOC\_tool**.

The prompt changes to <DOC\_tool>.

- 2 Specify the storage library.

If you want information for other than the default storage library, use the `setlibr` command to specify a different storage library.

- 3 Enter desired commands.

To repeat a command, enter **!!** at the DOC\_tool prompt.

- 4 Exit DOC\_tool.

When your analysis is complete, enter **quit** at the prompt.

## Related Topics

See the “Storage Library Control” chapter of your *System Administrator’s Handbook*.

To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21\*\*](#).

## EBR\_clean

The EBR\_clean utility is for use **only by your service representative** to reclaim shared memory and interlocks for a failed backup or restore.

For additional information, refer to Appendix A of the *Enterprise Backup/Restore User's Guide*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21\*\*](#).

## EBR\_genscript

The EBR\_genscript tool generates dataset definition files, device specification files, and builds backup and restore scripts through a question-and-answer interface. In addition, EBR\_genscript optionally determines the number of tapes required to back up the selected datasets.

For additional information, refer to Appendix A of the *Enterprise Backup/Restore User's Guide*.

### EBR\_label

The EBR\_label utility labels and relabels the following:

- tapes in a stand-alone tape drive used for backup
- tapes in a tape library used for backup
- magnetic disk files used for backup

For additional information, refer to Appendix A of the *Enterprise Backup/Restore User's Guide*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21\*\*](#)

## EBR\_orreset

If an Oracle database in online backup mode terminates abnormally, the database is left in a state in which the rollforward of redo logs has not completed. To reset the database to a working state, run the EBR\_orreset utility.

For additional information, refer to Appendix A of the *Enterprise Backup/Restore User's Guide*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## EBR\_tdir

The EBR\_tdir utility displays the tape label on a tape in a stand-alone tape drive or tape library, or on a disk file.

For additional information, refer to Appendix A of the *Enterprise Backup/Restore User's Guide*.

## EBR\_ulmk

EBR\_ulmk unlocks MKF databases that have been locked for an offline backup that did not complete. Locking the database insures a clean backup by keeping the database offline for the duration of the backup.

For additional information, refer to Appendix A of the *Enterprise Backup/Restore User's Guide*.

# eladisp

## Description

Through a command-driven interface, eladisp displays statistics collected by the Event Logging Abstract (ELA). In certain instances, you simply need to monitor ELA data collection. In others, you need to modify ELA data collection capabilities to obtain specific information about your system or to aid in troubleshooting. Data collection capabilities are controlled by the contents of an event codes file.

---

**Note** The eladisp tool collects data only for Courier remote programs.

---

The event codes file contains ELA event codes, a numeric code under which data is collected and maintained in the ELA shared library. When you start eladisp, eladisp first attempts to read an event codes file. If you specified the -d option, eladisp reads the event codes file you specified. Otherwise, eladisp reads the default event codes file, /fnsw/lib/perf/ela/elacodes in UNIX and \fnsw\lib\perf\ela\elacodes in Windows Server. If the attempt to read the file is unsuccessful, eladisp terminates. If successful, eladisp initializes all internal tables and sets up its input/output environment.

You control the display of ELA statistics by entering commands. Event names are the most important command parameters. Event names are symbolic names that map to the ELA event codes. You can assign ELA codes to any software module or even across modules.

An event code has two parts: the major category and the minor category. These parts map to major category names and minor category names. A complete event name consists of a major category name and a minor category name (for example, an event name could consist of a remote program name and a procedure name). When only a major category name is specified in an eladisp command, eladisp considers all the minor categories under the major category as well.

ELA major and minor event names are input for many eladisp commands. These parameters are described below:

- <major>** A symbolic ELA major category name or a unique prefix for one, such as a remote program name like DOC or CSM. For remote programs, a “c” suffix is added to indicate client stubs. ELA names and codes are in /fnsf/lib/perf/ela/elacodes (UNIX) and \fnsf\lib\perf\ela\elacodes (Windows Server). Names must begin with an alphabetic character and are limited to 16 characters.
- <minor>** A symbolic ELA minor category name associated with a major category. Minor event categories are listed under the major event categories in /fnsf/lib/perf/ela/elacodes (UNIX) and \fnsf\lib\perf\ela\elacodes (Windows Server).

Use the show directory command to display a list of all major and minor event categories defined for your system.

The partial output below lists the major event code for a WorkFlo Queue Services (WQS) client with its 39 minor event names.

```

eladisp> show directory
.
.
.
WQSc                minors = 39                WQSc (client)

    Minor categories:
    *****

CloseQueue           Continue             CountEntries
CreateQueue          CreateWorkspace    DeleteEntry
DeleteQueue         DeleteWorkspace    DropTable
DumpQueueStats      EmptyQueue         EndDump
GetQueueDesc        GetServerStats     GetTableName
GetWorkspaceInfo    InsertEntry        InstallQueue
Logoff              Logon              Logon1
OpenQueue           ReadDump           ReadEntry
ReadQueue           StartDump          ToggleTrace
UpdateEntry         UpdateQueue        UpdateWorkspace

```

Compare the output above with information stored in the event codes file (elacodes) shown below. The syntax of file entries is:

```

.event <event_code> <symb_name> [<n_minors>] ["<help_
string>"]
<list_of_minor_statements>
.end

```

---

```
.event 90 WQSc 39 "WorkFlo Queue Services (client)"
Logon 1
Logoff 2
Continue 3
CreateQueue 4
DeleteQueue 5
GetQueueDesc 6
UpdateQueue 7
OpenQueue 8
CloseQueue 9
CountEntries 10
ReadQueue 11
ReadEntry 12
InsertEntry 13
DeleteEntry 14
UpdateEntry 15
StartDump 16
EndDump 18
#GetQueueNames 19
#GetServerName 20
GetServerStats 23
DumpQueueStats 24
ToggleTrace 27
DropTable 29
Logon1 31
GetTableName 32
CreateWorkspace 33
DeleteWorkspace 34
UpdateWorkspace 35
GetWorkspaceInfo 36
InstallQueue 37
EmptyQueue 38
.end
```

---



## Use

Use eladisp to display statistics collected by the Event Logging Abstract or to monitor or modify ELA data collection capabilities.

Enter **eladisp** at the command line to start the tool. An information message displays and the command prompt changes to eladisp>. At the prompt, enter eladisp commands or a? if you need help.

```
corona(root)/> eladisp
eladisp: found 27 event categories

                Type ? for help
eladisp>
```

For command help and descriptions, see [“Commands” on page 386](#).

## Syntax

eladisp [-d <filename>]

**-d <filename>** Full path name of the event codes file. If you do not specify a name, eladisp uses the default file /fnsw/lib/perf/ela/elacodes (UNIX) or \fnsw\lib\perf\ela\elacodes (Windows Server).

---

**Note** The -d option does not redirect output. Use the standard syntax available with your command shell to redirect the output.

---

## Commands

If you start eladisp with no arguments, it locates and displays the number of event categories currently being logged by ELA then returns a prompt (eladisp>) from which you enter commands. Each command has a set of options. Help is available for each command and its options.

### Getting Command Help

Three levels of help are available to help you select the appropriate eladisp command and options:

- To display a list of all commands, enter a question mark (?) at the eladisp prompt.
- To display a list of all options of a command category, enter a question mark followed by a command (**? show**) at the eladisp prompt.

- To obtain a list of command options and a description of the information this combination provides, enter the help command followed by the command (for example, **help show**) at the eladisp prompt.

A question mark (?) displays all eladisp commands:

---

```

eladisp>?
Main ELA commands
-----
Commands to assign ELA parameters
-----
ASSIGN BINDINGS <channel_id> {[<major_event> | <minor_event>]}
ASSIGN BINS <bins>
ASSIGN CHFILTER <channel_id> [['<hostnum>']] [<groupname>]
ASSIGN HISTOGRAM <minorevent> <histog_typ> <lowvalue> <highvalue>
ASSIGN MATCHCOND <minorevent> [['<hostnum>']] [<groupname>]
ASSIGN TRACE <size>

Commands to disable data collection
-----
DISABLE CHFILTER <channel_id>
DISABLE EVENT <majorevent_list>
DISABLE HISTOGRAM <minorevent> <histog_typ>
DISABLE MATCHCOND <minorevent>
DISABLE TRACE <channel_id>

Commands to enable data collection
-----
ENABLE EVENT <majorevent_list>
ENABLE HISTOGRAM <minorevent> <histog_typ>
ENABLE TRACE <channel_id>

Data display commands
-----
SHOW BINDINGS <channel_id>
SHOW CHANNEL [<major_event> | <minor_event> | <channel_id>]
SHOW CHFILTER <channel_id>
SHOW DESCRIPTION [<majorevent> | <minorevent>]

```

---

```

SHOW DIRECTORY      [<major> | <minor>]
SHOW EVENT          [<minor> | <major>]
SHOW HISTOGRAMS    [<major> | <minor>]
SHOW LOCKSTATS
SHOW STATUS        <channel_id>
SHOW TRACE         channel_id [<list_of_event_names> ["| <shell_cmd>", "> file"]---
-----

```

#### Special commands

#### Commands to clear ELA data

```

-----
CLEAR EVENT        *** No execution permission
CLEAR HISTOGRAM   <minor> <histog_typ>

```

#### Commands to remove ELA objects

```

-----
REMOVE BINDINGS   {[<major> | <minor>]}
REMOVE CHANNEL   *** No execution permission
REMOVE EVENT     *** No execution permission
REMOVE HISTOGRAM <minor>

```

#### Commands to reset trace channel data

```

-----
RESET CHANNEL    *** No execution permission
RESET READP     <channel_id> <offset>

```

#### Commands of general utility

```

-----
!      <command>
?      [<command>]
HELP   [<command>]
HISTORY
QUIT
READ   <filename>
SET    [<variable>] [<value>]
^      [<number> | <command>]
^^
-----

```

Compare the differences between the two levels of command help in the following examples:

```
eladisp>? show
```

```
Data display commands
```

```
-----
SHOW BINDINGS      <channel_id>
SHOW CHANNEL      [<major_event> | <minor_event> | <channel_id>]
SHOW CHFILTER     <channel_id>
SHOW DESCRIPTION  [<majorevent> | <minorevent>]
SHOW DIRECTORY    [<majorevent> | <minorevent>]
SHOW EVENT        [<minorevent> | <majorevent>]
SHOW HISTOGRAMS  [<majorevent> | <minorevent>]
SHOW LOCKSTATS
SHOW STATUS       <channel_id>
SHOW TRACE        <channel_id> [<list_of_event_names>] [" | <shell_cmd>", ">
file"]
```

```
eladisp>help show
```

```
Data display commands
```

```
-----
SHOW BINDINGS      <channel_id>
--- Show binding(s) for a channel
SHOW CHANNEL      [<major_event> | <minor_event> | <channel_id>]
--- Verify channel id or show binding for an event or event category
SHOW CHFILTER     <channel_id>
--- Show match (filter) conditions for a channel
SHOW DESCRIPTION  [<majorevent> | <minorevent>]
--- Print remote procedure description
SHOW DIRECTORY    [<majorevent> | <minorevent>]
--- Show list of events
SHOW EVENT        [<minorevent> | <majorevent>]
--- Show ELA event data
```

```
SHOW HISTOGRAMS  [<majorevent> | <minorevent>]
--- Show histogram data
SHOW LOCKSTATS
--- Show ELA lock statistics
SHOW STATUS      <channel_id>
--- Show status (description) of a channel
SHOW TRACE       <channel_id> [<list_of_event_names>] ["| <shell_cmd>", ">
file"]
--- Show event trace entries for a channel
```

---

Many eladisp commands are available. This section describes only the most frequently used, starting with general utility commands which affect the operation or the data display mode of many other commands.

### General Utility Commands

General utility commands change command execution modes, enhance your use of eladisp, or improve your productivity with the tool. The use of the help command is discussed in [“Getting Command Help” on page 386](#). Some of the other most commonly used general utility commands—read, set, and data display controls—are described below.

#### **read <filename>**

Reads and interprets commands from the specified input file. Within the file, all text on a line following a # symbol is interpreted as a com-

ment. The following sample file contains commands to collect all active ELA data in a UNIX file:

```
# eladisp script to log all ELA data
#
set journ ela.log      # open a journal
set scroll on          # start scrolling mode
set term off           # optional command
show event             # displays basic usage data
show histogram         # displays histograms
set journ off          # close journal
```

### **set [<variable>] [<value>]**

Sets or inspects a program variable value.

If you enter set with no parameters, the command lists the values of all variables.

If you enter set with one parameter, the set command lists the value of the variable specified by the first parameter. However, if you specify a second parameter, eladisp attempts to set the specified variable to the value specified by the second parameter.

### **Data Display Control Commands**

Another group of general utility commands control the display of data. The eladisp output is designed for display on tty device screens in one of three modes: **interactive**, **scrolling**, or **auto-refresh**.

In **interactive** and **auto-refresh modes**, a list of available display control commands appears at the bottom of the screen as a prompt. The program waits for you to select a control command before continuing. You can enter all control commands as a single character (without

pressing the return key). The following table describes each data display control command character:

Command Character	Action
q	Exit from the command or exit from the current level of detail (return to the previous level)
n	Move to the next data object, for example, the next active major event
p	Move the previous data object
d	Display data in greater detail, or move to a level of greater detail
r	Refresh data from the ELA shared library
space bar	Move in a circular fashion through data objects (Use only for histograms of a given minor category.)
h	Display the histogram, if any, for the minor event
c	Reset the histogram data to zeros

In **auto-refresh mode**, eladisp refreshes the screen after a specified interval, and when you enter another command such as p, r, or n.

**Scrolling mode** is available for show event and show histogram commands. The show trace command is always in scrolling mode. In scrolling mode, eladisp sends all data relevant to a particular display command to the screen without prompting you. Scrolling mode is only intended for redirection of output to a file or when you want data collection to occur in the background. Scrolling mode is enabled automatically when you specify the **read** general utility command or if you start eladisp with output redirected to a file.



If you activated a journal with the set command, scrolling is not automatically enabled and you are in interactive mode with the command. To manually enable scrolling mode, use the command:

**set scroll on**

To set scrolling mode back to interactive mode, use the command:

**set scroll off**

## Data Display Commands

In general, all data display commands accept either no argument, a major event name, or a minor event name. Some of the commonly used data display commands and ways to use them are described below (uppercase letters indicate the abbreviated form of the command and parameters). Data display control commands are described when available for command output displays.

### **SHow Bindings <channel\_id>**

Shows bindings for a channel. A channel ID is required input.

```
eladisp> sh bindings 0
CSM:  DOC:  BES:
```

### **SHow CHAnnel [<major\_event> | <minor\_event> | <channel\_id>]**

Verifies the channel ID or shows the binding for an event or event category.

```
eladisp> sh cha
Channels:  0
```

**SHow DDescription [<major> | <minor>]**

Displays remote procedure description.

**Note** The Show Description command is available only when trace symbols are set on.

If you do not specify an argument, eladisp prints a list of known Courier objects. These include the remote program (event category) names.

```

eladisp> sh de
Programs                                type          objects read from COR
document:
-----
BulkData                                : PROGRAM    63 VERSION   1 < PREFIX = Bulkdata
>
Clearinghouse                           : PROGRAM    61 VERSION   2 < PREFIX = NCH >
CacheServices                           : PROGRAM    0 VERSION   1 < PREFIX = CSM >
IndexServices                           : PROGRAM    2 VERSION   1 < PREFIX = INX >
DocServices                             : PROGRAM    1 VERSION   1 < PREFIX = DOC >
eladisp>

```

**SHow Directory [<major> | <minor>]**

Displays a list of event names from the event directory. This command is intended as a help feature. If you do not specify an argument, eladisp displays a list of major events and minor event names for each. If you specify a major event name, the events for only that category display. If you specify a minor event name, eladisp displays the single minor event and its major category.

The following example displays directory information for the major event CSM (Cache Services Manager).

```

eladisp>sh di CSM
-----
CSM                minors = 31                CSM (server)

  Minor categories:
  *****

ChgRefcnt          CloseCsumObject          CloseObject
ClsDelObject       Continue                CopyObject
CreateObject       DeleteObject            FindObjects
GetBucketCounts   GetCacheAttributes      GetObjectAttributes
IncRefCnt          Logoff                  Logon
Logon1             ModifyObjectAttributes  MoveObject
OpenCsumObject     OpenObject              ReadObject
RenameObject       ResizeObject            ToggleTrace
WriteObject

```

### **SHow EVent [<minorevent> | <majorevent>]**

Displays basic information and logging status for an event. If you do not specify a major or minor event, eladisp displays a list of major events assigned to your station, the logging state of each, number of

minors, and number of active minors. Valid display control commands are r, q, and d.

```

riesling ( 1 ) **** journal begin Tue Dec 10 14:12:46 1996

eladisp>show event

Tue Dec 10 14:12:50 1996

Event      Status      Minors      Active
=====
BES        enabled     60           0
CPT        enabled     8            0
CSM        enabled     31           5
CSMc       enabled     31           0
DOC        enabled     50           6
FIL        enabled     12           0
INX        enabled     80           9
NCH        enabled     39           2
PRI        enabled     20           0
PSMc       enabled     10           0
SEC        enabled     45           8
SQI        enabled     15           0
WQS        enabled     40           0

[r,d,q]>
q
eladisp>quit

```

If you select **d** (next level of detail), eladisp shows a summary of a major event with a list of active event names and activity counts in a fixed size window at the top of the screen (long lists are truncated). Only events with active minor events display at this level. Display control commands n, p, d and q are available in the window. The d com-

mand takes you to the minor category level, displaying use data for active minors, two at a time, below the major category summary. Use p and n to move through the list; q returns you to the previous level. Use r to refresh the data for the major category. (Refreshed data is indicated by an increase in the time displayed on the screen.)

### **SHow Histograms [<major> | <minor>]**

Displays histogram data. If you do not specify an argument, a list of histogram entries displays. The d command at this level displays the first histogram entry. Only assigned histograms (those with non-trivial parameters) display. Available display control commands are the space bar, r, n, p and q. To reset all histogram data to zero, use the c command.

## **Main ELA Commands**

You can selectively generate traces of individual events with trace initialization commands. For example, you could want to examine the sequence of Remote Procedure Calls (RPCs) arriving at various servers as a result of batch entry activity driven by a WorkFlo script.

Event tracing supports the creation of multiple circular trace logs, called channels. It creates each channel with a specified size, which is the maximum number of events it can contain. It supports a maximum of 16 channels, each containing up to 128 events. (The global memory requirement per event is approximately 128 bytes.)

Individual events (or event categories) are uniquely bound to a channel for tracing. Each channel can be enabled and disabled independently. In addition, you can specify a filter condition for a channel with the Assign Chfilter command. A filter condition is a source/destination host

address or FileNet security name or both. If you specify a filter condition, only events that satisfy the condition are recorded in the channel.

An event is logged into a channel if:

- the corresponding event category has been enabled for logging
- the event or event category has been bound to an existing channel
- the channel itself has been enabled
- the filter condition defined for the channel, if any, is satisfied

To initialize a trace log, you must issue eladisp commands that create or remove tracing channels. You can also dynamically modify trace logging parameters. The following commands perform these functions.

**ASsign BINDings <channel\_id> {[<major\_event> | <minor\_event>]}**

Binds event categories or minor events to a channel.

**ASsign TRace <size>**

Assigns a new tracing channel of the specified size (number of events), up to the maximum size of 128 events.

**ENable TRace <channel\_id>**

Enables event trace data logging for an existing channel. The trace channel must have been previously assigned with the assign trace command.

**REMove BINDings {[<major\_event> | <minor\_event>]}**

Removes bindings for event categories or minor events.

**DISable TRace <channel\_id>**

Disables event trace data logging for the specified channel.

**Checklist**

Before you use eladisp, be aware that eladisp is primarily intended for use by service representatives involved in performance analysis or technical support.

## Procedure

No specific procedure is required. However, the following steps represent a typical set of commands to create a trace log to channel 0 and start the collection and display of BES, CSM and DOC data.

- 1 Enter **eladisp** at the command line.

The prompt changes to eladisp>.

- 2 At the eladisp prompt, enter the following sequence of commands.

```
show channel  
assign trace 32  
assign bind 0 CSM BES DOC  
enable trace 0  
show binding 0  
show status 0  
.  
      (other commands as necessary)  
.  
remove trace 0
```

- 3 Enter **quit** to exit eladisp.

## Related Topics

[\*\*“getreports” on page 523\*\*](#)

[\*\*“perf\\_mon” on page 954\*\*](#)

[\*\*“perf\\_report” on page 968\*\*](#)



# endbackup

## Description

The endbackup script calls initfnsw restart to kill the RPC\_Listen, OCOR\_Listen, /etc/logproc, NCH\_daemon, and MKF\_clean processes that are running as a result of a backup operation. The script then issues a command to unmount the file systems and returns the server to multi-user mode. A message is written to the server's backup recovery log when endbackup runs. The system starts up various databases and displays progress messages as each starts.

## Use

Run endbackup after a backup has completed to kill any current processes and return the server to multi-user mode.

## Syntax

```
endbackup
```

## Sample Output

In the example below, endbackup is run on the server and the system message displays. In a production system, this message is followed by additional system messages indicating progress as the system starts up the databases.

```
$ endbackup
96/08/22 08:27:23 endbackup: returning to multi-
user mode
please wait
$
```

## Checklist

Before you use endbackup, be aware that you must run endbackup on each server, beginning with the Root server.

## Procedure

No specific procedure is required.

## Related Topics

[\*\*“endbackup” on page 401\*\*](#)

[\*\*“initfnsw” on page 583\*\*](#)

See your *System Administrator’s Companion for UNIX* or *System Administrator’s Companion for Windows Server* for a discussion of the backup operation. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## enlarge\_ncol

### Description

The Microsoft® SQL Server™ RDBMS and IBM DB2 RDBMS require you to specify a fixed length format in a user mask when you create a numeric index. You must specify the number of digits, called precision and scale, before and after the decimal point. This tool automatically adds padding digits to allow for future increases in the number of digits that make up the numeric index without creating another column. If you try to increase the mask in Database Maintenance to an amount larger than the allocated amount, you get an error message.

You can use tools provided by the RDBMS to create a new column with a larger format and move data from the old column to the new column. You can also use the `enlarge_ncol` tool to automatically perform this operation.

---

### CAUTION

The FileNet Image Services software **must** be shutdown when running `enlarge_ncol`.

---

The `enlarge_ncol` tool increases the precision and scale of an existing numeric column in a Microsoft SQL Server database and a DB2 database. The tool performs the following sequential operations to accomplish the column enlargement:

- Renames the old column
- Creates a new column with a larger format
- Copies the data from the old column to the new column
- Inserts null characters in the old column to recover the space

Before this tool enlarges the column, it displays a summary of the proposed updates with a confirmation prompt. You can enter **y** to continue or **n** to terminate the operation.

## Use

Use `enlarge_ncol` to increase the precision and scale of a numeric user index when the user index is enlarged to a size greater than its physical definition in the database.

---

### CAUTION

Limit the number of numeric index enlargements. Every index that is enlarged uses another column in the DOCTABA table of the index database. The space used by the old column is wasted.

---

## Syntax

```
enlarge_ncol [<options>]
```

Supply the input options on the command line or when prompted by `enlarge_ncol` when it runs. (See [“Examples” on page 406.](#))

### Options

- i <indexname>** Specifies the numeric index to enlarge
- p <precision>** Sets a new precision

Precision is the total number of digits on both the left and the right side of the decimal point (excluding the decimal point, commas, etc.). The new precision value must be greater than the old precision value. The maximum value for Microsoft SQL Server is 30. The maximum value for DB2 is 30.

---

**-s <scale>** Sets a new scale

Scale is the number of digits shown on the right side of the decimal point. The new scale value could be equal to or greater than the old scale value. The scale value must be less than or equal to the precision value. The scale value cannot be increased unless the precision value is also increased.

**-r <rows>** Specifies the maximum number of rows to update before committing (default: 2000)

**-n** Sets old column data to null. If you do not specify this option, the old column contains the old data.

**-h** Prints syntax help text

---

**Tip** Use the RDBMS native query tool to determine the current precision and scale settings. Also, the `enlarge_ncol` tool displays the current settings before prompting you for confirmation to update, as shown below:

```
Current database column is type numeric(21,7)
New database column will be type numeric(22,8)
Update (y/n):
```

---

## Examples

The following example is the result of a successful attempt to enlarge a numeric column (responses to prompts appear in bold type):

```
> enlarge_ncol

    Enter name of the numeric user index to enlarge:
User_index_num
    Enter precision: 22
    Enter scale (0-22): 8
    User index 'User_index_num' will be updated as
follows:

    Current database column is type numeric(21,7)
    New database column will be type numeric(22,8)

    Update (y/n): y
    Please wait...
    Copying data from column 'a38' to column 'a39'
    96/08/12 17:21:23  22100 rows updated
    Successfully enlarged numeric index 'User_index_
num'
```

The example below is the result of an attempt to enlarge a numeric index using an incorrect precision or scale (responses to prompts appear in bold type):

```
> enlarge_ncol

Enter name of the numeric user index to enlarge: User_
index_num
Enter precision: 23
Enter scale (0-22): 8
Precision must be greater than existing precision
Existing precision = 22, New precision = 23
Enlarging numeric index 'User_index_num' failed
```

If the attempt to enlarge a column fails, messages and a prompt display. You can rerun `enlarge_ncol` to finish the operation, as shown in the example (response to the prompt appears in bold type):

```
Attempt to enlarge numeric column 'User_index_num' failed.
```

```
In order to enlarge another numeric column you must either
finish or cancel the failed attempt. Canceling will, in
most cases, waste a column in DOCTABA.
```

```
Finish processing (y/n) ? y
Copying data from column 'a39' to column 'a40'
96/08/12 17:29:54 22100 rows updated
Successfully enlarged numeric index 'User_index_num'
```

## Checklist

Before you use `enlarge_ncol`, be aware of the following:

- This tool is I/O intensive. On a very large database, column enlargement can take a long time.
- Since `enlarge_ncol` is modifying the database configuration, run the tool only when the Image Services software is shut down.
- The database uses more space as a result of this operation, since the data has to be copied into the new column before the old column can be NULLed out. If there is no free space available in the database, you might have to increase the database size before running the tool.

## Procedure

- 1 Shut down the Image Services software.
- 2 Start the relational database management system if it is not still up.

---

**Note** With a DB2 system or a site-controlled system, the RDBMS is probably already up.

---

- 3 Enter **`enlarge_ncol`** at the command line.

If you do not specify options when you enter the command, the program prompts you for options.

- 4 When the tool completes, restart the Image Services software.

## Related Topics

See the Microsoft *SQL Server Administrator's Companion*



## export\_cdb

### Description

This export tool copies the latest Configuration Database (.cdb) file from /fnsw/local/sd/conf\_db to the /fnsw/local/sd/conf\_db/export directory. This tool is designed to be the first step in the process of cloning the IS system configuration of multiple IS systems from a single master .cdb file and could be done as a part of a fresh install of Image Services or as part of a migration. The other tools associated with this process are the [import\\_cdb](#) tool and the [migrate\\_cdb\\_changes](#) tool.

---

**Note** All the path names documented for this tool are shown in UNIX format (e.g., /fnsw/local/sd/conf\_db/export). For Windows Server path names, substitute the /fnsw/local/ with <drive>:\fnsw\_loc\ and use a back slash (\) instead of a forward slash (/) throughout the rest of the path (e.g., <drive>:\fnsw\_loc\sd\conf\_db\export).

---

## The Cloning Process

### Cloning Through Fresh Installs

The latest version of the .cdb file from a master IS system is used as the master .cdb file, and the clone system creates a new configuration database from this master file by doing the following:

- Replacing system-specific information with information from the clone systems. This includes domain name, server name, and system serial number (ssn).
- Copying the dataset configuration on the cloned system, including path and filenames.

- Copying all relational database information on the cloned system, including paths, server name, instance name, version, etc.

### **Cloning Through Migration**

The latest version of the .cdb file from a master IS server is used as the master .cdb file, and only changes to performance tuning parameters are being cloned.

- Only relational database performance parameters are modified by the process.
- The relational database type, version, etc. aren't modified.

### **System Requirements for Cloning**

Both the master and clone systems must have common attributes in order for the tools to work and the process to complete successfully. These are:

- Combined server systems
- No high availability supported
- Same operating system, IS version, relational database, including version and type (e.g., both FileNet-controlled, local site-controlled, or remote site-controlled).

## Safety Check Table

A safety check is automatically run as part of the tool operation and whenever a safety check fails, the application logs an error to the system log file and the program aborts. The following table describes the safety check rules:

### System Cloning Safety Check

Attribute	export_cdb	import_cdb	migrate_cdb_changes
IS_Release	Not checked	Master/Clone must match 100%	Master/Clone must match 100%
OS_Level	Not checked	Master/Clone must match 100%	Master/Clone must match 100%
DB_Version	Not checked	Not checked	Master/Clone must match 100%
Platform	Not checked	Master/Clone must match 100%	Master/Clone must match 100%
IS_Type	Check on the local IS box only	Not checked	Check on the local IS box only
No High Availability (Windows only)	Check on the local IS box only	Check on the local IS box only	Check on the local IS box only
RDB Control Type	Not checked	Not checked	Master/Clone must match 100%

## Use

You use this tool to export an `ims_exp_xx.cdb` file to `/fnsw/local/sd/conf_db/export` directory where `xx` is the revision number of the `.cdb` file. The tool always selects the latest version of the `.cdb` file to export.

The newly exported file name is READ ONLY and is in the following format: `ims_exp_xx.cdb`.

The `export_cdb` tool should be used in conjunction with the `import_cdb` tool or the `migrate_cdb_changes` tool.

## Syntax

```
export_cdb [-s <source_prefix>]
```

### Options

**-s <source\_prefix>** Optionally allows the user to specify a prefix for the source file other than the default “ims” prefix.

## Procedure

- 1 On the IS server being used as the master server, use `fn_edit` to define the configuration database parameters. This has probably been done and the latest `cdb` file is established.
- 2 On the IS server being used as the Master server, run **export\_cdb** to make a copy of the master IS server’s configuration database file by entering the following command:

### **export\_cdb**

In the above example, the export file would be created in the following directory structure: `/fnsw/local/sd/conf_db/export/ims_exp_xx.cdb`, where `xx` is the revision number of the `.cdb` file.

If you want to use the optional **-s <source\_prefix>** parameter, `<source_prefix>` is the value you use to associate the file with the clone server the configuration file is being exported to (for example, `california`). In this `-s` example, the export file would be created in the following directory structure: `/fnsw/local/sd/conf_db/export/california_xx.cdb`, where `xx` is the revision number of the `.cdb` file.

- 3 Manually move or copy the newly created file from the master IS server to the clone IS server by placing it in the following location on the clone IS server:

```
/fnsw/local/sd/conf_db/import/ims_exp_xx.cdb
```

- 4 On the IS server being used as the clone server, run one of the following commands to import the configuration database file, depending upon your current system scenario:

**import\_cdb** (create .cdb file for a fresh install)

**migrate\_cdb\_changes** (update .cdb file for a migration)

Again if you want to use the optional **-s <source\_prefix>** parameter, <source\_prefix> is the value you use to associate the file with the clone server the configuration file is being exported to (for example, california).

---

**Note** When you import or migrate using the optional -s parameter, you are still creating a new revision for the default IS configuration database file (ims\_xx.cdb). The xx in the file name will be one revision higher.

---

## Related Topics

[“import\\_cdb” on page 576](#)

[“migrate\\_cdb\\_changes” on page 709](#)

# FileNet System Manager

## Description

FileNet System Manager is a performance monitoring tool that will collect performance information from the FileNet products. In order to use FileNet System Manager on IS, the IS Server must be running a System Manager listener agent which talks to a remote FileNet System Manager through a specified port.

## Use

Since Image Services already has a performance monitoring tool, [perf\\_mon](#), it was just enhanced to add the listener agent functionality. With the listener agent mode turned on, perf\_mon continues to capture data to its perfllog in normal mode and it also acts as the listener agent by writing data to a specified port for the System Manager to read. The data capture intervals will be identical for both modes. If you want to have two different intervals for the two modes of operation, two instance of perf\_mon can be run.

In perf\_mon, the System Manager listener agent is turned off (set listener false) by default. The listener agent can be turned on temporarily by manually running an instance of perf\_mon and setting the Listener variable to True (see [Make the System Manager Listener Agent Run Temporarily](#)). Also, if you want the listener agent to always run whenever the IS software is started up, you can set the Listener variable to True in the perf\_mon.script start file (see [Make the System Manager Listener Agent Always Run](#)).

---

## Procedures

### Make the System Manager Listener Agent Run Temporarily

- 1 To start an instance of perf\_mon, enter:

**perf\_mon**

- 2 To turn the listener agent on through the perf\_mon tool, enter a sampling command and then exit perf\_mon, enter:

```
schedule 1 00:00:00 02:00:00
schedule 1 07:55:00 00:15:00
schedule 1 18:15:00 02:00:00
schedule 2 00:00:00 03:00:00
link 1 1
link 2 1
link 3 1
link 4 1
link 5 1
link 6 2
link 0 2
poll savefile 300
perf_mon> set listener true
perf_mon> sample outputfile 60 6
perf_mon> quit
```

### Make the System Manager Listener Agent Always Run

To set Listener to True as the default so the System Manager listener agent will always run whenever the IS software is started up, complete the following steps.

- 1 Go the /fnsw/local/sd directory (<drive>:\fnsw\_loc\sd - for Windows) to see if a perf\_mon.script exists. If one exists, skip to Step 2. If you don't have a perf\_mon.script you can create one by copying it from /fnsw/lib/perf/ (<drive>:\fnsw\lib\perf - Windows).

---

```
cp /fnsw/lib/perf/perf_mon.script /fnsw/local/sd/perf_mon.script
```

copy to <drive>:\fnsw\_loc\sd on a Windows server.

- 2 From the /fnsw/local/sd directory (<drive>:\fnsw\_loc\sd - for Windows), using vi or your preferred editor, edit the perf\_mon.script file to make the first command line of the script file to be **set listener true**. Below is the edited scrip file:

```
# stamp
#
set listener true
schedule 0 0:00:00 2:00:00
schedule 0 6:00:00 0:15:00
schedule 0 19:00:00 2:00:00
schedule 1 0:00:00 2:00:00
link 0 1
link 1 0
link 2 0
link 3 0
link 4 0
link 5 0
link 6 1
poll /fnsw/local/sd/1/perflog
echo done
# stamp d;lkfjpoj;r;wohf
```

- 3 Save and Exit the file.
- 4 Restart the FileNet software.

## Related Topics

[“perf\\_mon” on page 954](#)



---

## flat

The UNIX logo consists of the word "UNIX" in white, uppercase letters, centered within a red parallelogram shape that is wider at the top and tapers towards the bottom.

The flat tool is available only on a UNIX-based Image Services server.

### Description

The FileNet LAN Analysis Tool (flat) reads and decodes the output of network analyzer programs and network packet traces on UNIX platforms. It provides some FileNet-specific analysis capabilities not found in other commercially-available analyzers. The flat tool also analyzes traces captured from a variety of other tools, such as:

- Sniffer
- LANalyzer
- LANWatch
- iptrace (AIX/6000 platform only)
- snoop (Solaris Operating Environment platform only)

Your service representative dials into a server to analyze the trace data or an on-site representative can examine the trace. This enables the service representative to diagnose a wide variety of system-level problems.

### Use

Use flat to analyze trace data for system problem resolution.

## Syntax

flat [<trace\_file>]

**<trace\_file>** Name of a saved trace output data file. If you do not specify the trace file name at the time you invoke flat, you can use the flat **display (d)** command to specify the file. (See [“Trace Commands” on page 431](#).)

When you start flat, command mode becomes active and the **CMD>** prompt displays and waits for input.

## Command Quick Reference

The flat tool provides several commands, each one initiated by typing a single character. Some of these commands have different functions, depending on whether you're working in command or display mode. The following table briefly describes these commands in alphabetical order.

### Quick Reference of flat Commands

Enter	Command	Description
!	!-shellcmd	Runs a shell command
/	/ and ?	Searches forward for a packet containing a given hexadecimal string
?	/ and ?	Searches backward for a packet containing a given hexadecimal string
a	a-all conns	Symbolically formats RPCs for all connections
b	b-page up	Moves back (up) one screen and displays the next page of data
c	c-connections	Performs connection analysis when viewing trace data
c	c-calls	Formats RPCs for a given connection. This command is available after you select the connections option.
d	d-half page down	Moves down one-half screen when displaying data
d	d-isplay	Displays a packet trace in command mode

## Quick Reference of flat Commands, Continued

<b>Enter</b>	<b>Command</b>	<b>Description</b>
<b>e</b>	e-dit	Edits packet filter
<b>e</b>	e-rrors	Corrects errors. This command applies only when a connection has a protocol error (at user or transport level) or there is a timing or performance problem.
<b>f</b>	f-page down	Moves forward (down) one screen and displays the next page of data
<b>g</b>	g-goto	Goes to line n, where n=line number
<b>j</b>	j-next	Moves down 1 line
<b>k</b>	k-prev	Moves up 1 line
<b>n</b>	n-next	Moves down 1 packet
<b>o</b>	o-ld RPC format	Toggles between original flat and newer ELA format
<b>p</b>	p-rev	Moves up 1 packet in display mode
<b>p</b>	p-ackets	Performs packet analysis on each packet in the trace and formats the data when viewing trace data
<b>q</b>	q-uit	Returns to previous display mode
<b>q</b>	q-uit	Terminates flat and returns to the UNIX shell from command mode
<b>s</b>	s-save file	Writes selected data in ASCII format to a text file
<b>t</b>	t-time base	Sets the time base for the current packet to 0 and all subsequent packets are relative to time 0
<b>u</b>	u-half page up	Moves up one-half screen and display the data
<b>v</b>	v-verbose	Sets verbose mode, a one-page-per-packet display

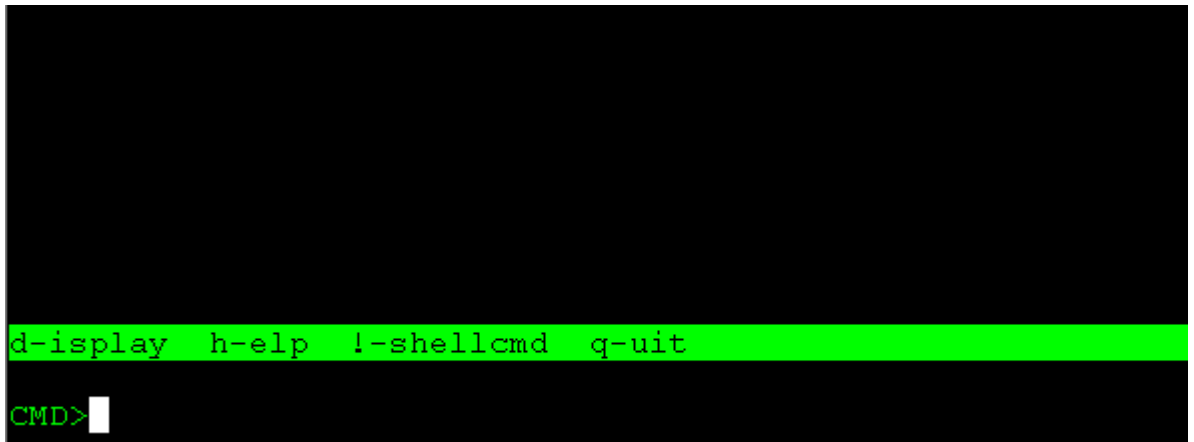
For more detailed descriptions of flat commands, see the following sections, organized by type of command:

- [“Basic Display Commands” on page 424](#)
- [“Miscellaneous Display Commands” on page 429](#)
- [“Trace Commands” on page 431](#)

## Interface Overview

You interface with flat through windows and by selecting context-sensitive commands from a menu bar in each window. Selecting a menu option presents a hierarchy of display modes (described later in this section).

Once active and in command mode, flat displays a **CMD>** prompt and a list of available commands in a menu bar at the bottom of the screen, similar to the following:

A screenshot of the flat command interface. The background is black. At the bottom, there is a green horizontal bar containing the text 'd-display h-help !-shellcmd q-uit' in a monospaced font. Below this bar, the text 'CMD>' is displayed in green, followed by a white cursor block.

```
d-display h-help !-shellcmd q-uit  
CMD>
```

You can request a list of flat commands with a brief description by entering **h** at the **CMD>** prompt:

```
FLAT -- FileNet Lan Analysis Tool

commands:
  d -- Display collected packets
  h -- Display help information
  e -- Edit packet filter
  i -- Display information about the collection
  l -- Load filter from file
  s -- Save filter to file
  r -- Start collecting packets
  q -- Quit
  ! -- Run a shell command

d-display h-help !-shellcmd q-uit

CMD> 
```

Enter the trace file name at the prompt as shown below:

```
FLAT -- FileNet Lan Analysis Tool

commands:
  d -- Display collected packets
  h -- Display help information
  e -- Edit packet filter
  i -- Display information about the collection
  l -- Load filter from file
  s -- Save filter to file
  r -- Start collecting packets
  q -- Quit
  ! -- Run a shell command

d-display h-help !-shellcmd q-uit

trace file /home/fnsw/sample_traces/telnettest.cap
```

**Tip** If you started flat with a specified trace name, the file name becomes the default trace file and automatically displays in the **trace file** prompt field when you enter the display command. You could enter a new trace file name or press Enter to accept the displayed trace file.

After you enter the trace file name, flat displays the terse packet display window and highlights the currently active entry and the available commands for the current mode:

```

FLAT -- FileNet Lan Analysis Tool

TIME          PLEN  DESTINATION  ORIGINATION  TYPE
22:52:07.016  60    10.1.71.200  10.1.71.206  ip    TCP  S
22:52:07.016  58    10.1.71.206  10.1.71.200  ip    TCP  SA
22:52:07.017  60    10.1.71.200  10.1.71.206  ip    TCP  A
22:52:07.017  69    10.1.71.206  10.1.71.200  ip    TCP  APM
22:52:07.038  69    10.1.71.200  10.1.71.206  ip    TCP  APM
22:52:07.039  63    10.1.71.206  10.1.71.200  ip    TCP  APM
22:52:07.051  72    10.1.71.200  10.1.71.206  ip    TCP  AP
22:52:07.051  69    10.1.71.206  10.1.71.200  ip    TCP  APM
22:52:07.054  60    10.1.71.200  10.1.71.206  ip    TCP  APM
22:52:07.035  54    10.1.71.206  10.1.71.200  ip    TCP  A
22:52:07.036  65    10.1.71.200  10.1.71.206  ip    TCP  AP
22:52:07.037  104   10.1.71.206  10.1.71.200  ip    TCP  AP
22:52:07.052  60    10.1.71.200  10.1.71.206  ip    TCP  A
22:52:07.052  64    10.1.71.206  10.1.71.200  ip    TCP  AP
22:52:07.056  60    10.1.71.200  10.1.71.206  ip    TCP  AP
22:52:07.056  57    10.1.71.206  10.1.71.200  ip    TCP  AP

k-prev  g-goto  v-verbose  c-connections  q-cmd  mode  !-she

sample_traces/telnettest.cap (snoop 2.0)  70 packets

```

See [“Basic Display Commands” on page 424](#) for a description of the commands that appear at the lower portion of the display.

## Basic Display Commands

The flat tool has several display modes—command mode, terse packet display mode, and connection display mode. Many commands have different meanings depending on the display mode you select. However, all display screens support a basic set of display commands you use to move through trace data, save or print a file, and execute commands from a shell.

**Note** Basic display commands operate on lines in the display in a manner similar to the commands you use with the **less** utility (see [“less” on page 692](#)). Although a currently highlighted line of the display could be a packet, a connection, or other component, the operation of all commands in the basic set is the same.

The following table lists the set of basic display commands with a description of the action taken when you select the command from the menu bar:

Basic Display Commands and Descriptions

Command	Description
j-next	Moves down 1 line. You cannot move past the last line in the trace.
k-prev	Moves up 1 line. You cannot move past the first line in the trace.
g-goto	Goes to line <b>n</b> , where <b>n</b> is the line number. The program prompts you for a line number that must be between 1 and the maximum number of lines displayed. If you specify a number out of the range of available entries, flat displays an error message.
u-half page up	Moves up one-half screen and displays the data. You cannot move past the beginning of the trace.
d-half page down	Moves down one-half screen and display the data. You cannot move past the end of the trace.



## Basic Display Commands and Descriptions, Continued

Command	Description
f-page down	Moves forward (down) one screen and displays the next page of data. You cannot move past the end of the trace.
b-page up	Moves back (up) one screen and displays the data. You cannot move past the beginning of the trace.
s-save file	Writes selected data in ASCII format to a text file. The program prompts you to enter a file name and a print command. Since flat deals with binary data files which you cannot print or display without a special viewer, use this command to create text files for printing, displaying, or attaching in electronic mail.
!-shellcmd	Runs a shell command from this display mode
q-uit	Returns to the previous display mode
/ and ?	Searches for a packet containing a given hexadecimal string. Use / to search forward and ? to search backward.

As you make selections, the commands in the menu bar change to reflect those allowed for the selected display mode.

When you enter the **c** (connections) command from connection mode level, the following window displays. Compare the available menu options to those in the previous window.

```

NN# SOURCE HOST DEST HOST #PKTS START TIME DURATN
1 10.1.71.202 10.1.71.200 94 23:09:36.055 20.953
2 10.1.71.202 10.1.71.200 1 23:09:46.005 0.000
3 10.1.71.200 10.1.71.202 13 23:09:46.009 2.043
4 10.1.71.200 10.1.71.202 10 23:09:48.052 -1.990
5 10.1.71.200 10.1.71.202 13 23:09:49.062 -1.987
6 10.1.71.200 10.1.71.202 10 23:09:49.049 -1.956
7 10.1.71.200 10.1.71.202 13 23:09:49.052 -1.967
8 10.1.71.200 10.1.71.202 10 23:09:49.055 -1.962
9 10.1.71.200 10.1.71.202 10 23:09:49.016 -1.986
10 10.1.71.200 10.1.71.202 10 23:09:49.003 3.030
11 10.1.71.200 10.1.71.202 10 23:09:52.014 0.013
12 10.1.71.200 10.1.71.202 10 23:09:52.027 0.013
13 10.1.71.200 10.1.71.202 10 23:09:52.038 -1.997
14 10.1.71.200 10.1.71.202 10 23:09:52.035 0.018
15 10.1.71.200 10.1.71.202 13 23:09:52.055 -1.976
16 10.1.71.200 10.1.71.202 10 23:09:53.047 -1.968

next k-prev g-goto p-packets c-all s q-uit a-ll conn:

le: sample_traces/wal_logon.cap 257 packets 17 connect:

```

All flat display screens support a basic set of commands you can use to move through the trace data, save and print the file, and execute shell commands. (See [“Basic Display Commands” on page 424.](#)) In

addition, flat has several display modes—command mode, terse packet display mode, and connection display mode. You can view these command modes as a tree structure.

As you move from one display mode level to the next, the commands available from the menu bar have different results. To return to a previous command display level, enter **q** (quit). A few commands do not change modes, but instead simply execute a function. (See [“Miscellaneous Display Commands” on page 429](#).)

---

**Tip** To minimize the volume of information displayed, use the **c**, **a**, **e** command sequence, as described below.

---

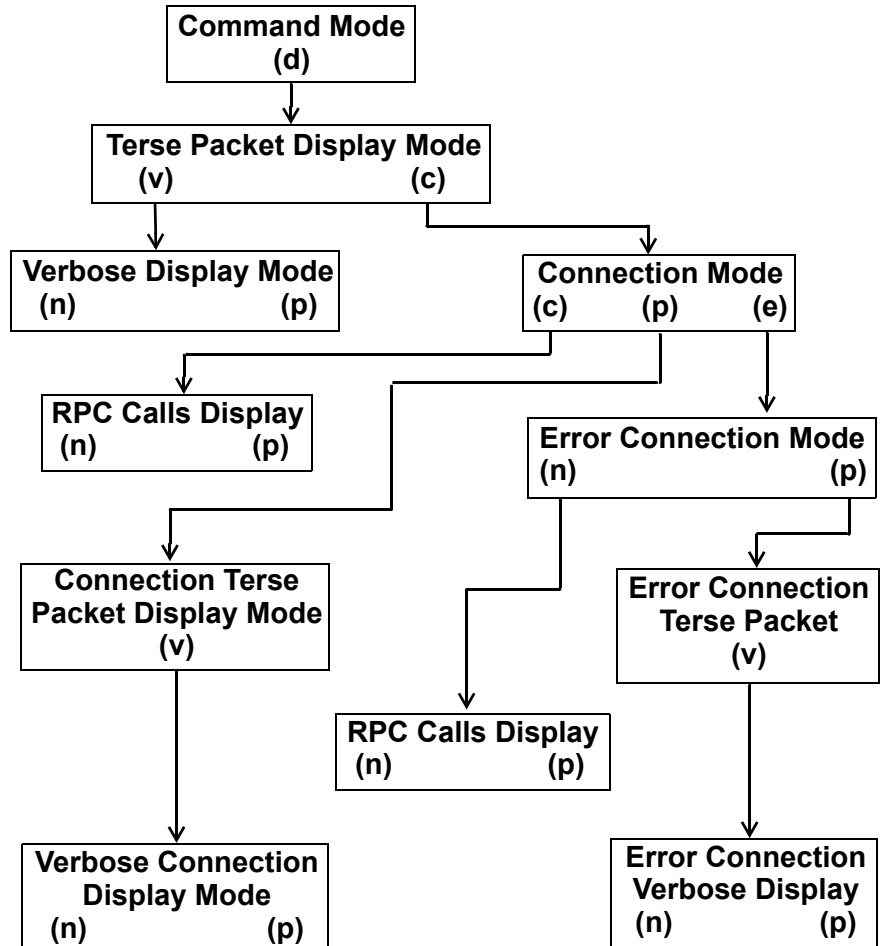
- 1** Display network connection (instead of packets) using the **c** (connections) command. This displays fewer objects to examine.
- 2** Analyze all connections at both the network transport and FileNet RPC levels using the **a** (all connections) command.
- 3** Display only the connections that contain connection errors using the **e** (error) command.

---

**Note** Experience with FileNet applications, the IS Toolkit libraries, and network protocols can help you more effectively analyze trace data.

---

The following diagram shows the relationship between display modes and the commands associated with each mode.



Note: Miscellaneous commands t, a, n, and p do not change modes.

### Flat Command Mode Hierarchy

## Miscellaneous Display Commands

Each display mode level in flat supports a set of basic commands and a set of context-sensitive commands. For basic command descriptions, see [“Basic Display Commands” on page 424](#).

The following miscellaneous commands do not change modes, but instead simply execute the function described in the table within one of the display modes:

Miscellaneous Display Commands and Descriptions

Command	Function Description
n-ext	Moves to the next object, which could be a packet, a call, or a connection. You cannot move past the last packet in the trace. This command is available from verbose display mode and verbose connection packet display mode.
p-rev	Moves to the previous object, which could be a packet, a call, or a connection. You cannot move past the first packet in the trace. This command is available from verbose display mode and verbose connection packet display mode.
t-time base	Sets the time base for the current packet to 0. All subsequent packets are relative to time 0. This command is available from terse packet display mode and connection terse packets display mode. The time base command displays packet-relative times. When you are in time base mode, press <b>t</b> again to restore the display to absolute time.

## Miscellaneous Display Commands and Descriptions, Continued

Command	Function Description
a-ll conns	Symbolically formats Remote Procedure Calls (RPCs) for all connections. This command is available from connection display mode only. Use this command to discover which RPCs were used, whether any were rejected or aborted, or whether any RPCs experienced transport-level or lower-level errors. The program name or procedure name of a rejected RPC is prefixed with an asterisk (*); a prefix of + indicates an aborted RPC. To check for RPCs that have had transport-level or lower-level errors, look for Courier connections that are not successfully formatted and which remain shown as type "Courier."
e-rrors	Displays errors. This option applies only when a connection has a protocol error (at user or transport level) or when the system has a timing or performance problem.

## Trace Commands

These flat commands perform viewing and display functions on the trace.

### Viewing and Display Commands

Command	Description
d-isplay	<p>Displays a packet trace. Flat prompts you for a file name after you enter <b>d</b>.</p> <p>By convention, Sniffer Ethernet trace file names have a suffix of <b>.enc</b>, while Sniffer Token Ring trace file names end with <b>.trc</b>. Snoop trace files have a <b>.cap</b> suffix. LANalyzer traces are often split into multiple files, with suffixes <b>.tr1</b>, <b>.tr2</b>, and so on. Specify only the <b>.tr1</b> suffix when requesting that flat examine a multi-file LANalyzer trace. This enables the flat tool to use information in the first file to determine how many other files are used. If any of these files are missing, flat displays an error.</p> <p>Packets in a trace are numbered from 1 to <b>n</b>, where <b>n</b> is the number of the last packet. Display mode formats trace data into an easy-to-read table, one row per packet, and displays the entries on a terminal screen. The table has column headers for packet number, time (in milliseconds), packet length (in bytes), source (“origination”) host, destination host, and protocol type.</p>
!-shellcmd	<p>Executes a shell command from within flat. You do not have to exit flat to issue shell commands. When the shell command completes, flat prompts you to press Enter to redisplay the screen.</p>
q-uit	<p>Terminates flat and returns to the shell. (Quit is an alias for the Escape key.)</p> <p>As you select commands from the flat <b>CMD&gt;</b> prompt, additional context-sensitive commands become available. Some options are only available with certain types of command selections (see <b><u>“Miscellaneous Display Commands” on page 429</u></b>).</p>

## Viewing and Display Commands, Continued

Command	Description
v-erbose	<p>Sets verbose mode, a one-page-per-packet display. It symbolically displays the fields in protocol headers and other packet data in both hexadecimal and ASCII formats.</p> <p>Move the cursor within the list to make a line item selection or use the <b>j</b>, <b>k</b>, or <b>g</b> commands to make your selection. Then enter <b>v</b> to display detailed information about the selected line item.</p>
c-onnections	<p>Performs connection analysis. The flat program examines each packet in the trace, identifying connections, then formats the data in either terse or verbose mode. The flat tool displays the connection number, source host, destination host, number of packets, and start time, duration, and type of the connection. If you use the <b>a</b> (all conns) command to first symbolically format the connections information, the terse display of connections also indicates a rejected or aborted RPC. To denote a rejected RPC, flat prefixes the program name or procedure name with an asterisk (*); the prefix for an aborted RPC is a plus sign (+).</p>
c-alls	<p>Formats RPCs for a given connection. This option is available only after selecting the connections option.</p>
p-ackets	<p>Performs packet analysis on each packet in the trace and formats the data. The flat tool displays the packet number, a time stamp, packet length, source host, destination host, and the packet type. This option is available only after selecting the connections option.</p>
q-cmd mode	<p>Returns to the previous command mode screen</p>
o-ld RPC format	<p>Toggles between original flat and newer ELA format</p>



## Samples

The samples below display the described flat commands. You could find it helpful to refer to the diagram [“Flat Command Mode Hierarchy” on page 428](#) as you examine the flow between display modes.

Normally, you use a certain set of commands to examine a trace. For every trace, we recommend that you first select the terse packet display mode, then connections, followed by all connections. Further examination of packets, connections, etc. is dictated by the problem you are trying to solve (for example, hardware vs. software problem, network vs. application problem, etc.).

### Basic Trace Examination

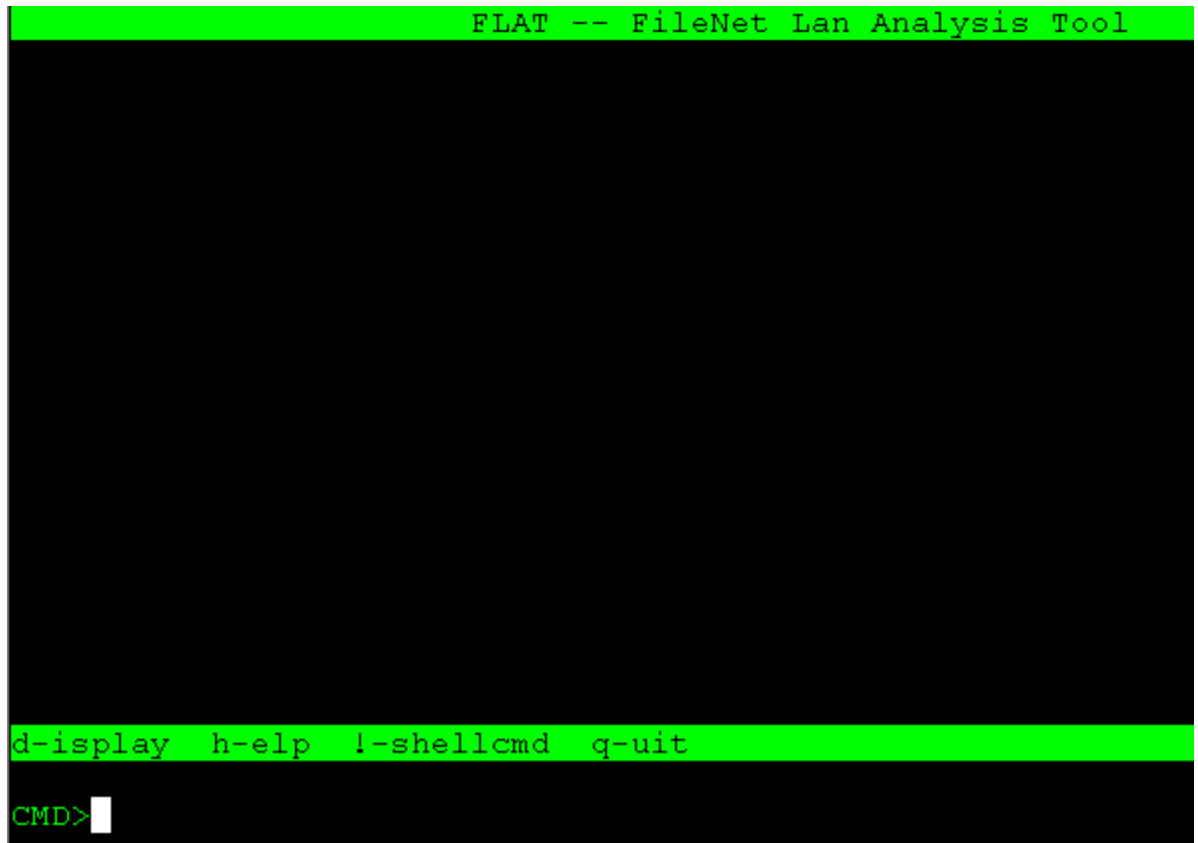
The following samples show a set of commands typically used to examine a trace.

---

**Note** If you enter the flat command without a trace file name, you must specify a trace file name with the **d** command. The requested trace data displays after you enter the file name.

---

The following illustration shows the initial screen in command mode:



```
FLAT -- FileNet Lan Analysis Tool

d-isplay h-elp !-shellcmd q-uit

CMD>
```

Note the commands available at the bottom of the screen and the command mode prompt, **CMD>**.

For example, to display a trace file, you would type **d**, then enter the file path and name at the prompt for trace file.

The flat tool displays the following screen, called the terse packet display mode. Note that if you had invoked flat using the **file path and name**, you would already be at the following display, as shown for the trace file wfd\_logon.cap, stored in the subdirectory sample\_traces.

```

FLAT -- FileNet Lan Analysis Tool
PKTNUM      TIME          PLEN  DESTINATION  ORIGINATION  TYPE
  1 17:03:08.050      68  10.1.73.254  10.1.71.200  ip    UDP
  2 17:03:08.053      84  10.1.71.200  10.1.73.254  ip    UDP
  3 17:03:08.056      68  10.1.73.254  10.1.71.200  ip    UDP
  4 17:03:08.058      62  10.1.71.200  10.1.73.254  ip    TCP
  5 17:03:08.058      58  10.1.73.254  10.1.71.200  ip    TCP
  6 17:03:08.059      60  10.1.71.200  10.1.73.254  ip    TCP
  7 17:03:08.060     120  10.1.71.200  10.1.73.254  ip    TCP
  8 17:03:08.037      54  10.1.73.254  10.1.71.200  ip    TCP
  9 17:03:08.063      62  10.1.73.254  10.1.71.200  ip    TCP
 10 17:03:08.063      94  10.1.73.254  10.1.71.200  ip    TCP
 11 17:03:08.064      62  10.1.73.254  10.1.71.200  ip    TCP
 12 17:03:08.064      60  10.1.71.200  10.1.73.254  ip    TCP
 13 17:03:08.065      60  10.1.71.200  10.1.73.254  ip    TCP
 14 17:03:08.065      54  10.1.73.254  10.1.71.200  ip    TCP
 15 17:03:08.000      54  10.1.73.254  10.1.71.200  ip    TCP
 16 17:03:08.000      60  10.1.71.200  10.1.73.254  ip    TCP
j-next k-prev g-goto v-verbose c-onnections q-cmd mode

File: sample_traces/wfd_logon.cap (snoop 2.0) 166 packets

```

The connections command is a very important command and differentiates flat from all commercially available network analyzers on the market.

The power of the connections command is its ability to reduce a trace of 165 objects (packets), in our example above, to a trace containing just 12 connections. Analysis and problem identification become much easier tasks. In a production environment, a similar trace file could contain thousands of packets, making it very difficult to detect anomalies.

The connections command output helps identify problems detected based on the number of packets, packet length, the duration, or the connection type. Gathering this information manually requires considerable networking expertise.

To see the connections detail for this trace file, select the **c** (connections) command from the terse packet display menu bar. The resulting display, called the connection mode display, is shown below:

```

FLAT -- FileNet Lan Analysis Tool

CONN# SOURCE HOST DEST HOST #PKTS START TIME DURATN
1 10.1.71.200 10.1.73.254 1 17:03:08.050 0.000
2 10.1.73.254 10.1.71.200 2 17:03:08.053 0.003
3 10.1.73.254 10.1.71.200 13 17:03:08.058 -1.942
4 10.1.73.254 10.1.71.200 13 17:03:14.015 0.002
5 10.1.73.254 10.1.71.200 10 17:03:14.017 -1.990
6 10.1.73.254 10.1.71.200 13 17:03:15.014 0.001
7 10.1.73.254 10.1.71.200 10 17:03:15.015 0.013
8 10.1.73.254 10.1.71.200 10 17:03:15.029 -1.990
9 10.1.73.254 10.1.71.200 10 17:03:15.041 -1.971
10 10.1.73.254 10.1.71.200 8 17:03:15.015 1.001
11 10.1.71.200 10.1.73.254 1 17:03:16.037 0.000
12 10.1.73.254 10.1.71.200 2 17:03:16.039 0.003
13 10.1.73.254 10.1.71.200 13 17:03:16.044 0.005
14 10.1.73.254 10.1.71.200 10 17:03:16.049 -1.960
15 10.1.73.254 10.1.71.200 10 17:03:16.010 0.006
16 10.1.73.254 10.1.71.200 10 17:03:16.016 1.041

j-next k-prev g-goto p-packets c-calls q-quit a-all conns

File: sample_traces/wfd_logon.cap 166 packets 19 connectio

```

You could use the next level of analysis to examine the packets and/or the calls using the **a**, **p**, and **c** commands. The **a** command applies to “all” connections, while the **p** and **c** commands operate on the currently highlighted entry. The following screens show the output from the **p** and **c** selections.

In the first example below, the **p** command displays packets for connection 4.

```

FLAT -- FileNet Lan Analysis Tool

PKTNUM      TIME          PLEN  DESTINATION  ORIGINATION  TYPE
  17 17:03:14.015    62  10.1.71.200  10.1.73.254  ip    TCP
  18 17:03:14.016    58  10.1.73.254  10.1.71.200  ip    TCP
  19 17:03:14.016    60  10.1.71.200  10.1.73.254  ip    TCP
  20 17:03:14.017   110  10.1.71.200  10.1.73.254  ip    TCP
  21 17:03:14.000    54  10.1.73.254  10.1.71.200  ip    TCP
  22 17:03:14.014    62  10.1.73.254  10.1.71.200  ip    TCP
  23 17:03:14.014    94  10.1.73.254  10.1.71.200  ip    TCP
  24 17:03:14.014    62  10.1.73.254  10.1.71.200  ip    TCP
  25 17:03:14.015    60  10.1.71.200  10.1.73.254  ip    TCP
  26 17:03:14.016    60  10.1.71.200  10.1.73.254  ip    TCP
  27 17:03:14.016    54  10.1.73.254  10.1.71.200  ip    TCP
  28 17:03:14.016    54  10.1.73.254  10.1.71.200  ip    TCP
  29 17:03:14.017    60  10.1.71.200  10.1.73.254  ip    TCP

j-next k-prev g-goto v-verbose q-cmd mode !-shell comma

File: sample_traces/wfd_logon.cap Connection: 4 13 packets

```

The next example has selected the **c** (calls) command when displaying a Snoop trace file named **wfd\_logon.cap** in the `sample_traces` subdirectory. This displays the formatted RPC calls associated with connection 4, a logon to the Network Clearing House, calling the List Objects command.

```

FLAT -- FileNet Lan Analysis Tool

CONN# SOURCE HOST   DEST HOST       #PKTS  START TIME  DURATN
  4  10.1.73.254  10.1.71.200     13  17:03:14.015  0.002

Message 0  packet 20  bytes 52 (0x34)  Msgtype call  tid 0x
  NCH.ListObjects
    pattern = RECORD  [
      organization = "FileNet\00",
      domain = "sunlite\00",
      object = "*\00"
    ]c
    property = 0xead7 (60119)c
    agent = RECORD  [
      credentials = 0 (0),
      verifier = 0 (0)
    ]

Message 2  packet 23  bytes 36 (0x24)  Msgtype bulk

j-next k-prev g-goto q-cmd mode l-shell command h-elp

File: sample_traces/wfd_logon.cap  166 packets  19 connecti

```

To examine a specific packet in detail, use the verbose command from the connection terse packet display mode. Enter **q** to exit the above RPC calls display mode and return to the connection mode display.

The following screens show the sequence from connection mode to display packet information in verbose mode for the file telnettest.cap.

For the first example, we selected **g** (go to) and entered a **4** at the prompt to select a connection.

```

PKTNUM      TIME          PLEN  DESTINATION  ORIGINATION  TYPE
   1 22:52:07.016    60  10.1.71.200  10.1.71.206 ip    TCP
   2 22:52:07.016    58  10.1.71.206  10.1.71.206 ip    TCP
   3 22:52:07.017    60  10.1.71.200  10.1.71.206 ip    TCP
   4 22:52:07.017    69  10.1.71.206  10.1.71.206 ip    TCP
   5 22:52:07.038    69  10.1.71.200  10.1.71.206 ip    TCP
   6 22:52:07.039    63  10.1.71.206  10.1.71.206 ip    TCP
   7 22:52:07.051    72  10.1.71.200  10.1.71.206 ip    TCP
   8 22:52:07.051    69  10.1.71.206  10.1.71.206 ip    TCP
   9 22:52:07.054    60  10.1.71.200  10.1.71.206 ip    TCP
  10 22:52:07.035    54  10.1.71.206  10.1.71.206 ip    TCP
  11 22:52:07.036    65  10.1.71.200  10.1.71.206 ip    TCP
  12 22:52:07.037   104  10.1.71.206  10.1.71.206 ip    TCP
  13 22:52:07.052    60  10.1.71.200  10.1.71.206 ip    TCP
  14 22:52:07.052    64  10.1.71.206  10.1.71.206 ip    TCP
  15 22:52:07.056    60  10.1.71.200  10.1.71.206 ip    TCP
  16 22:52:07.056    57  10.1.71.206  10.1.71.206 ip    TCP

j-next k-prev g-goto v-verbose c-connections q-cmd mode

File: sample_traces/telnettest.cap (snoop 2.0) 70 packets

```



The next example results from selecting **v** to display connection **4** in verbose mode. As you can see, this displays detailed information on the selected packet.

```

PKTNUM      TIME          PLEN  DESTINATION  ORIGINATION  TYPE
  4 22:52:07.017    69  10.1.71.206  10.1.71.200  ip    TCP

00000:      dest_address      02608C2C8557
00006:      source_address    0800208105B6
0000c:      packet_type       0800 (IP)
0000e:      ip.hdr_len_vers   45
0000f:      ip.type_of_svc    00
00010:      ip.len            0037
00012:      ip.id             73BC
00014:      ip.frag_off       4000
00016:      ip.time_to_live   FF
00017:      ip.protocol       06 (tcp)
00018:      ip.checksum       646C
0001a:      ip.src.addr       0A0147C8
0001e:      ip.dst.addr       0A0147CE
00022:      ip.tcp.src_port   0017 (telnet)

j-next k-prev g-goto n-ext p-revious q-display !-shell

File: sample_traces/telnettest.cap (snoop 2.0) 70 packets

```

**Tip** The number of packets associated with a connection is usually small—between 7 and 15. When you examine a trace, scan the **#PKTS** column for large values. Large values could indicate a problem that needs examination.

To display details for the next example, we selected n (next packet), displaying details for packet 5.

```

FLAT -- FileNet Lan Analysis Tool

PKTNUM      TIME          PLEN  DESTINATION  ORIGINATION  TYPE
-----
  5 22:52:07.038    69  10.1.71.200  10.1.71.206  ip    TCP

00000:      dest_address      0800208105B6
00006:      source_address    02608C2C8557
0000c:      packet_type       0800 (IP)
0000e:      ip.hdr_len_vers   45
0000f:      ip.type_of_svc    00
00010:      ip.len            0037
00012:      ip.id             179A
00014:      ip.frag_off       0000
00016:      ip.time_to_live   3C
00017:      ip.protocol       06 (tcp)
00018:      ip.checksum       C38F
0001a:      ip.src.addr       0A0147CE
0001e:      ip.dst.addr       0A0147C8
00022:      ip.tcp.src_port   11DB

j-next  k-prev  g-goto  n-next  p-previous  q-display  !-shell

File: sample_traces/telnettest.cap (snoop 2.0)  70 packets

```

Enter **q** (quit) to return to connection terse packet display mode.

In terse display mode, you can use the `t` (time base) command to examine packets relative to a base packet. The following display has selected packet `1` as the base packet for a time-based analysis.

```

FLAT -- FileNet Lan Analysis Tool

PKTNUM      TIME          PLEN  DESTINATION  ORIGINATION  TYPE
  1  0:00:00.000      60  10.1.71.200  10.1.71.206  ip
  2  0:00:00.000      58  10.1.71.206  10.1.71.200  ip
  3  0:00:00.001      60  10.1.71.200  10.1.71.206  ip
  4  0:00:00.001      69  10.1.71.206  10.1.71.200  ip
  5  0:00:00.022      69  10.1.71.200  10.1.71.206  ip
  6  0:00:00.023      63  10.1.71.206  10.1.71.200  ip
  7  0:00:00.035      72  10.1.71.200  10.1.71.206  ip
  8  0:00:00.035      69  10.1.71.206  10.1.71.200  ip
  9  0:00:00.038      60  10.1.71.200  10.1.71.206  ip
 10  0:00:00.019      54  10.1.71.206  10.1.71.200  ip
 11  0:00:00.020      65  10.1.71.200  10.1.71.206  ip
 12  0:00:00.021     104  10.1.71.206  10.1.71.200  ip
 13  0:00:00.036      60  10.1.71.200  10.1.71.206  ip
 14  0:00:00.036      64  10.1.71.206  10.1.71.200  ip
 15  0:00:00.040      60  10.1.71.200  10.1.71.206  ip
 16  0:00:00.040      57  10.1.71.206  10.1.71.200  ip

j-next  k-prev  g-goto  v-verbose  c-connections  q-cmd m

File: sample_traces/telnettest.cap (snoop 2.0)  70 packe

```

Entering **t** again returns the display to absolute time, as shown in the following example.

PKTNUM	TIME	PLEN	DESTINATION	ORIGINATION	TYPE
1	22:52:07.016	60	10.1.71.200	10.1.71.206	ip TCP
2	22:52:07.016	58	10.1.71.206	10.1.71.200	ip TCP
3	22:52:07.017	60	10.1.71.200	10.1.71.206	ip TCP
4	22:52:07.017	69	10.1.71.206	10.1.71.200	ip TCP
5	22:52:07.038	69	10.1.71.200	10.1.71.206	ip TCP
6	22:52:07.039	63	10.1.71.206	10.1.71.200	ip TCP
7	22:52:07.051	72	10.1.71.200	10.1.71.206	ip TCP
8	22:52:07.051	69	10.1.71.206	10.1.71.200	ip TCP
9	22:52:07.054	60	10.1.71.200	10.1.71.206	ip TCP
10	22:52:07.035	54	10.1.71.206	10.1.71.200	ip TCP
11	22:52:07.036	65	10.1.71.200	10.1.71.206	ip TCP
12	22:52:07.037	104	10.1.71.206	10.1.71.200	ip TCP
13	22:52:07.052	60	10.1.71.200	10.1.71.206	ip TCP
14	22:52:07.052	64	10.1.71.206	10.1.71.200	ip TCP
15	22:52:07.056	60	10.1.71.200	10.1.71.206	ip TCP
16	22:52:07.056	57	10.1.71.206	10.1.71.200	ip TCP

j-next k-prev g-goto v-verbose c-connections q-cmd mode

When you have completed your analysis, enter **q** (quit) to exit to the command mode display (**CMD>** prompt). Enter **q** (quit) again to exit flat.

Detailed examination can continue to very low levels of the network management components. Your service representative can assist you with detailed analysis.

---

## Files

The flat tool is versatile in its ability to decode and format specific information. A packet description file, `pkt.desc`, contains packet descriptions that flat uses to decode packets. You can make modifications to the description file. In addition, files with the naming convention of `x.y` contain descriptions of Courier RPCs for a program, where the hexadecimal program number is `x` and the program version is `y`.

The `pkt.desc` file and Courier RPC description files are contained in a default directory, `/fnsw/local/flat`. However, you can specify an alternate directory to contain the packet and Courier RPC description files by setting the `FLAT_DIR` environment variable. The following listing shows the files contained in the default directory:

```
corona(root)/> ls /fnsw/local/flat
0.1          8003401.1   8003404.1   8003415.1   800341a.1   pkt.desc
2.2          8003402.1   800340a.1   8003417.1   80034fe.1
8003400.1    8003403.1   8003414.1   8003418.1   80034ff.1
```

For example, `8003402.1` is the file that contains the RPC description for INX (which is Courier program number `0x8003402`), version 1. (The information is also available in the `/fnsw/etc/serverConfig` file.)

Use a combination of your modified `pkt.desc` file and a program file to decode and format trace data for analysis of a specific problem or to limit the amount of information that flat retrieves from the trace file.

## Checklist

Before you use flat, be aware of the following:

- You should use flat under the direction of your service representative.
- The flat tool is not supported on the Windows Server platforms.
- You can obtain trace data from network analyzer programs or from trace utilities as directed by your service representative.
- Many commands have different functions depending on the command level from which you invoke them.

## Procedure

Perform the following steps at the direction of your service representative:

- 1 Run the appropriate trace program, directing the output to a file.
- 2 Run flat against the trace file, using display mode commands to format and analyze the data.

## Related Topics

Use the UNIX **man ipttrace** command to display the related on-line manual pages.

## fn\_edit

The fn\_edit program invokes a graphical user interface (GUI) utility called the FileNet System Configuration Editor. This program creates and modifies the system configuration database.

For an introduction to fn\_edit, refer to the *System Configuration Overview*. For detailed instructions, use the program's online help system. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## fn\_ldif\_xfer

### Description

UNIXWIN

The standards surrounding LDAP (Lightweight Directory Access Protocol - a directory service protocol that runs over TCP/IP) have advanced. This tool provides a better performing conversion utility that increases the openness of the FileNet Image Services software. This tool translates industry standard LDIF (Lightweight Directory Interchange Format) files into the FileNet proprietary XML format.

This tool gives customers the ability to use their LDAP tools to generate the LDIF files for use with FileNet software by translating files into the proper format to be imported into Image Services and Content Services.

### Use

This utility will convert LDAP objects in LDIF format into FileNet proprietary XML format. The **LDAP\_import** tool is then used to import the file attributes into the Image Services security database. All objects are mapped one to one. This tool provides a batch (add) mode during import as opposed to the previous requirement of having all users and groups residing in a single file. Customers now have the freedom to create LDIF files any way they desire and the translator needs to only translate FileNet-related objects and attributes. The tool works in a three step process.

- **1** - A parsing of the input file occurs, which accomplishes several things: creates a temporary users file, a temporary groups file, an entry search table and a members linked list.



- **2** - Goes through the members linked list and for each member, finds its entry in the table, retrieves its type and inserts it into a temporary file.
- **3** - Finally, the tool merges all temporary files into a single FileNet proprietary XML file.

---

**Note** The creation of temporary files requires disk space in the current directory. Please ensure that adequate disk space is available before running this utility.

---

## Syntax

**WIN**

```
fn_ldif_xfer /i<ldif file> /o<output xml file> /t<server type>
             /s<server name> /a<ip address>
             [ /f<configuration file> ] [ /u<attribute name> ]
```

**UNIX**

```
fn_ldif_xfer -i<ldif file> -o<output xml file> -t<server type>
             -s<server name> -a<ip address>
             [ -f<configuration file> ] [ -u<attribute name> ]
```

---

**Note** Options are **not** case sensitive. Also, a / is used before an option on Windows systems and a - is used before an option on UNIX systems.

---

**[?]** Help. This will bring up the help screen.

**i<ldif file>** input LDIF file

**o<output xml file>** output FileNet XML file

**t<server type>** nov, msft, sun or other

**s<server name>** the LDAP server where the LDIF data is extracted from

**a<ip address>** the IP address of the server specified in the **s** option

**[f<configuration file>]** configuration file for tother option

**[u<attribute name>]** optional used ID attribute

The following table shows the relationships between the LDAP attributes and the FileNet XML objects:

#### LDAP to XML Attribute Mappings

FileNet XML Object Type	FileNet XML Object Attribute	MS Active Directory Attribute Name	Sun Java System Attribute Name	Novell NDS Attribute Name
<user>	<dn>	dn	dn	dn
<user>	<id>	userPrincipalName	uid	cn
<user>	<cn>	name	cn	cn
<user>	<description>	description	description	description
<group>	<dn>	dn	dn	dn
<group>	<id>	cn	cn	cn
<group>	<cn>	cn	cn	cn
<group>	<description>	description	description	description
<member>	<groupName>	dn	dn	dn
<member>	<memberName>	member	uniquemember	member
<member>	<memberType>	Not applicable	Not applicable	Not applicable

**Note** Sun Java™ System was formally known as Sun ONE Directory Server.

- The **tother** option supports LDAP vendors other than Active Directory, ONE and NDS. However, the **tother** option must be used with

the **f<configuration file>** option allowing you the freedom of defining custom LDAP attribute mappings. The following is an example of an ASCII-based configuration file.

userdn	dn
userid	cn
usercn	cn
userdescription	description
groupdn	dn
groupid	cn
groupcn	cn
groupdescription	description
membergroup	dn
membername	member
objectclass	objectclass
userobjectclass	user
groupobjectclass	group

Each line consists of two string tokens separated by spaces or tabs. The first token is the FileNet XML object attribute descriptor which should never be modified. The second token is the custom LDAP attribute field. Both token types are not case sensitive.

- The following table displays the object identifiers for each of the main LDAP vendors.

Main LDAP Vendor Object Identifiers

Object Type	MS Active Directory Object Identifier	Sun Java System Object Identifier	Novell NDS Object Identifier
user object	objectclass = person or objectclass = user	objectclass = person or objectclass = user	objectclass = person or objectclass = user or objectclass = inetorgperson
group object	objectclass = group	objectclass = groupofuniquenames	objectclass = groupofnames

## Generating LDIF Files

**Important** It is **important** to know how to correctly generate your LDIF files. When generating LDIF files, it is crucial that all group and user objects are fully exported, because:

- The FileNet software needs to know if a member is a user or a group.
- The FileNet software needs other attributes.

There is very definitely a correct and an incorrect way to do this. In the example below, the command is incomplete and the resulting output is inadequate because it does not indicate if the members are users or groups. It displays a partial export of a group called ThreeStooges using the **ldifde** tool.

---

**Note** There are many ways to generate LDIF data, and the **ldifde** tool is used here solely as an example. You could use any tool you wish.

---

#### Example of Inadequate LDIF File

---

```
>ldifde -f output.ldf -r "(&(objectclass=group)(cn=Three Stooges))"
```

```
dn: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
changetype: add
member: CN=Curley Fine,CN=Users,DC=ldap,DC=eng
member: CN=Moe Fine,CN=Users,DC=ldap,DC=eng
member: CN=Larry Fine,CN=Users,DC=ldap,DC=eng
member: CN=Bad!User,CN=Users,DC=ldap,DC=eng
memberOf: CN=test1group,OU=Groups,DC=ldap,DC=eng
memberOf: CN=grp1,OU=OU1,DC=ldap,DC=eng
cn: Three Stooges
groupType: -2147483646
instanceType: 4
distinguishedName: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
objectCategory: CN=Group,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: group
objectGUID:: 6XCih/E56kONEe87OZVm5g==
objectSid:: AQUAAAAAAAAUVAAAASyy8GkTduD2Ci6YoeicAAA==
name: Three Stooges
sAMAccountName: Three Stooges
sAMAccountType: 268435456
uSNChanged: 88788
uSNCreated: 88492
whenChanged: 20050725165211.0Z
whenCreated: 20050719225330.0Z
```

---

If you process the inadequate LDIF file, the following error is generated:

```
Warning: member type unknown, Group DN= ..., Member DN= ...
```

The example below illustrates the proper way to export a single group using a complete command, where each member object is fully exported.

#### Example of Complete LDIF File

```
>ldifde -f output.ldf -r "(|(&(objectclass=user)
(memberof=cn=Three Stooges,OU=groups,dc=ldap,dc=eng))
(&("objectclass=group)(cn=Three Stooges)))"
```

```
dn: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
changetype: add
member: CN=Curley Fine,CN=Users,DC=ldap,DC=eng
member: CN=Moe Fine,CN=Users,DC=ldap,DC=eng
member: CN=Larry Fine,CN=Users,DC=ldap,DC=eng
member: CN=Bad!User,CN=Users,DC=ldap,DC=eng
memberOf: CN=testlgroup,OU=Groups,DC=ldap,DC=eng
memberOf: CN=grp1,OU=OU1,DC=ldap,DC=eng
cn: Three Stooges
groupType: -2147483646
instanceType: 4
distinguishedName: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
objectCategory: CN=Group,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: group
objectGUID:: 6XC1h/E56kONEe870ZVm5g==
objectSid:: AQUAAAAAAAAUVAASyy8GkTduD2Ci6YoeicAAA==
name: Three Stooges
sAMAccountName: Three Stooges
sAMAccountType: 268435456
```

---

```
uSNChanged: 88788
uSNCreated: 88492
whenChanged: 20050725165211.0Z
whenCreated: 20050719225330.0Z

dn: CN=Larry Fine,CN=Users,DC=ldap,DC=eng
changetype: add
memberOf: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
accountExpires: 9223372036854775807
badPasswordTime: 0
badPwdCount: 0
codePage: 0
cn: Larry Fine
countryCode: 0
displayName: Larry Fine
givenName: Larry
instanceType: 4
lastLogoff: 0
lastLogon: 0
logonCount: 0
distinguishedName: CN=Larry Fine,CN=Users,DC=ldap,DC=eng
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: user
objectGUID:: u3v5AjL8sk+bgowPp1Jo9w==
objectSid:: AQUAAAAAAAAUVAAAASyy8GkTduD2Ci6YoeycAAA==
primaryGroupID: 513
pwdLastSet: 127662872680468750
name: Larry Fine
sAMAccountName: lfine
sAMAccountType: 805306368
sn: Fine
userAccountControl: 512
userPrincipalName: lfine@ldap.eng
uSNChanged: 88500
uSNCreated: 88496
whenChanged: 20050719225428.0Z
whenCreated: 20050719225427.0Z
```

```
dn: CN=Moe Fine,CN=Users,DC=ldap,DC=eng
changetype: add
memberOf: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
accountExpires: 9223372036854775807
badPasswordTime: 0
badPwdCount: 0
codePage: 0
cn: Moe Fine
countryCode: 0
displayName: Moe Fine
givenName: Moe
instanceType: 4
lastLogoff: 0
lastLogon: 0
logonCount: 0
distinguishedName: CN=Moe Fine,CN=Users,DC=ldap,DC=eng
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: user
objectGUID:: JNjpsEOCbEmePDRH3RO/YQ==
objectSid:: AQUAAAAAAAAUVAASy8GkTduD2Ci6YofCcAAA==
primaryGroupID: 513
pwdLastSet: 127662873013281250
name: Moe Fine
sAMAccountName: mfine
sAMAccountType: 805306368
sn: Fine
userAccountControl: 512
userPrincipalName: mfine@ldap.eng
uSNChanged: 88506
uSNCreated: 88502
whenChanged: 20050719225501.0Z
whenCreated: 20050719225501.0Z

dn: CN=Curley Fine,CN=Users,DC=ldap,DC=eng
changetype: add
memberOf: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
```



```
accountExpires: 9223372036854775807
badPasswordTime: 0
badPwdCount: 0
codePage: 0
cn: Curley Fine
countryCode: 0
displayName: Curley Fine
givenName: Curley
instanceType: 4
lastLogoff: 0
lastLogon: 0
logonCount: 0
distinguishedName: CN=Curley Fine,CN=Users,DC=ldap,DC=eng
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: user
objectGUID:: bd/jQ1tFKUm6f0DLYSATXQ==
objectSid:: AQUAAAAAAAAUVAASyy8GkTduD2Ci6YofScAAA==
primaryGroupID: 513
pwdLastSet: 127662873684218750
name: Curley Fine
sAMAccountName: cfine
sAMAccountType: 805306368
sn: Fine
userAccountControl: 512
userPrincipalName: cfine@ldap.eng
uSNChanged: 88512
uSNCreated: 88508
whenChanged: 20050719225608.0Z
whenCreated: 20050719225608.0Z

dn: CN=Bad!User,CN=Users,DC=ldap,DC=eng
changetype: add
memberOf: CN=Three Stooges,OU=Groups,DC=ldap,DC=eng
accountExpires: 9223372036854775807
badPasswordTime: 0
badPwdCount: 0
codePage: 0
```

```
cn: Bad!User
countryCode: 0
displayName: Bad!User s. User
givenName: Bad!User
initials: s
instanceType: 4
lastLogoff: 0
lastLogon: 0
logonCount: 0
distinguishedName: CN=Bad!User,CN=Users,DC=ldap,DC=eng
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: user
objectGUID:: dEEp5EfxJkeIEwodJT/8pQ==
objectSid:: AQUAAAAAAAAUAAAAASyy8GkTduD2Ci6Yo7gQAAA==
primaryGroupID: 513
pwdLastSet: 0
name: Bad!User
sAMAccountName: Bad!User
sAMAccountType: 805306368
sn: User
userAccountControl: 512
userPrincipalName: Bad!User@ldap.eng
uSNChanged: 72069
uSNCreated: 4452
whenChanged: 20040616155932.0Z
whenCreated: 20030915224630.0Z
```

---

## Examples

The following command converts a Sun Java System LDIF file to a FileNet XML file. This is a Windows example.

```
fn_ldif_xfer /imy.ldf /ofilenet.xml /tsun /sldapserver /a2.10.105.7
```

The following command converts an Active Directory LDIF file to a FileNet XML file, but uses the sAMAccountName field as the user id. This is a Windows example.

```
fn_ldif_xfer /imy.ldf /ofilenet.xml /tmsft /usAMAccountName  
/sldapserver /a5.105.17.10
```

## Checklist

Before using `fn_ldif_xfer`, there are several predefined rules that you must follow since the tool is designed for support use only:

- `fn_ldif_xfer` is intended for use on all UNIX and Windows platforms running Image Services 4.0 SP4 or higher.
- Make sure you have the LDAP server name and IP address available.
- You must be a member of the `fnadmin` or `fnop` group to run `fn_ldif_xfer`.
- Translations will only occur on Users, Groups and Members. No other objects will be translated.
- For each translatable object, only FileNet **required** attributes will be translated.

- You are allowed to provide an optional attribute for the FileNet user ID attribute (/u). The application default is show in the table below:

Directory Service Product Default Attributes

Directory Service Product	Default Attribute
Microsoft Active Directory	userPrincipalName
Sun Java System	uid
Novell NDS	cn

- No changes are made to the current XML schema.
- All general logging and error logging are output to the following:

**WIN**

\fnsw\_loc\logs\ldap\fn\_ldif\_xferYYYYMMDDlog.txt

**UNIX**

/fnsw/local/logs/ldap/fn\_ldif\_xferYYYYMMDDlog.txt

## LDIF Format Sample File

For specific details on LDIF, refer to RFC 2849 on the Internet using any search engine (e.g., Google).

LDIF files are ASCII-based with the following general rules:

- Objects are separated by a carriage return line feed (new line).
- One <attribute\_name>:<value> pair per line.
- Lines starting with a # sign indicate comments.
- Any non-empty line could be folded (continued) by inserting a line separator and a space.

## Sample LDIF File

---

```
# This is a sample LDIF file

dn: CN=Test T. User,CN=Users,DC=ldap,DC=eng
changetype: add
accountExpires: 9223372036854775807
badPasswordTime: 0
badPwdCount: 0
codePage: 0
cn: Test T. User
countryCode: 0
displayName: Test T. User
givenName: Test
initials: T
instanceType: 4
lastLogoff: 0
lastLogon: 0
logonCount: 0
distinguishedName: CN=Test T. User,CN=Users,
  DC=ldap,DC=eng
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: user
objectGUID:: fDey+Mh27k2YKVVrkqTUrg==
objectSid:: AQUAAAAAAAAUAAAAASyy8GkTduD2Ci6YoAQUAAA==
primaryGroupID: 513
pwdLastSet: 0
name: Test T. User
sAMAccountName: prewindowsname
sAMAccountType: 805306368
sn: User
userAccountControl: 512
userPrincipalName: userlogonname@ldap.eng
uSNChanged: 72150
uSNCreated: 4509
whenChanged: 20040616155935.0Z
```

```
whenCreated: 20030915224631.0Z

dn: CN=Bad!Group,CN=Users,DC=ldap,DC=eng
changetype: add
cn: Bad!Group
groupType: -2147483646
instanceType: 4
distinguishedName: CN=Bad!Group,CN=Users,DC=lda
  p,DC=eng
objectCategory: CN=Group,CN=Schema,CN=Configuration,DC=ldap,DC=eng
objectClass: group
objectGUID:: aLlyLAmLckKjeTSwCAkgrw==
objectSid:: AQUAAAAAAAAUVAAAASyy8GkTduD2Ci6YoAgUAAA==
name: Bad!Group
sAMAccountName: Bad!Group
sAMAccountType: 268435456
uSNChanged: 4514
uSNCreated: 4512
whenChanged: 20030915225148.0Z
whenCreated: 20030915225148.0Z
```

---

## Procedure

No specific procedure is required.

## Related Topics

[“ldap\\_exp” on page 662](#)

[“LDAP\\_import” on page 671](#)

# fn\_msg

## Description

The fn\_msg tool retrieves information that describes the error that occurred, including the shared library that detected the error. The first field of the information returned displays the shared library. In the sample output above, the error tuple 80,0,2 was issued by the DOC shared library. A brief description of the error is also supplied, if available.

## Use

Use fn\_msg to decode error message tuples.

## Syntax

```
fn_msg <errorspec>
```

**<errorspec >** Specifies one or two error codes. If you specify two error codes, separate the two codes with a hyphen.

An error code could be entered as three decimal numbers separated by commas, a decimal number suffixed with a period, or a hexadecimal number. For example, the following commands are equivalent:

```
fn_msg 126, 0, 103
fn_msg 2113929319.
fn_msg 7e000067
```



## Sample Output

The example below requests information on error code 80,0,2, received when attempting to locate a document. The fn\_msg tool returns information about the message tuple.

```
corona(root)/> fn_msg 80,0,2  
<DOC,0,2> Document not found by DOC.
```

## Checklist

Before you use fn\_msg, be aware that message information might not be available for all error tuples.

## Procedure

No specific procedure is required.

## Related Topics

See the *System Messages Handbook*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## fn\_perm

WIN

The **fn\_perm** tool is only available on the Windows Server platform and not available on the UNIX platforms.

### Description

The **fn\_perm** tool is a command line tool used to set up file and directory permissions on Windows servers where the set permissions option is not available.

### Use

This tool is predominately used in eProcess or Visual WorkFlo-only Image Services installations, which are installed using a mini-installer that puts down a subset of the full IS software. The tool can also be used when IS service packs are installed on those types of systems. It is not used during normal IS software installations and updates.

### Syntax

```
fn_perm <drive letter of location of \fnsw and/or \fnsw_loc>
```

If the **fnsw** and **fnsw\_loc** directories are located on different drives, **fn\_perm** needs to be run twice using a different drive letter argument each time.

**fn\_perm** does not generate any output on the screen or in a log file.

## fn\_pso\_driver

The fn\_pso\_driver tool is new in the IS 4.0 Itanium® Edition and IS 4.0 SP5 releases. It is not included or supported in earlier releases of Image Services.

### Description

The fn\_pso\_driver tool is used exclusively during the Programmable Security Object recovery process, and allows you to convert all standard SQL scripts and system scripts into customized scripts.

#### Important

---

This tool is highly sensitive, and its misuse could cause your IS system to behave unexpectedly. Please confer with your service representative before using this tool.

---

### Use

This tool generates PSO customized scripts.

### Syntax

**UNIX**

```
fn_pso_driver [-l | -d | -c | -t | -? ]
```

**WIN**

```
fn_pso_driver [/l | /d | /c | /t | /? ]
```

- l** List files to be converted
- c** Generate PSO customized scripts  
(If you omit this option, the tool generates default FileNet scripts.)
- d** Disable verbose mode

**t <string>** Database type, where <string> can be one of the following:

mssql oracle db2 none

(This option overrides the default RDB type)

? Help screen

## Procedure

Refer to the description of fn\_pso\_podf\_admin for PSO recovery procedures.

## Related Topics

[“fn\\_pso\\_podf\\_admin” on page 469](#)

[“fn\\_pso\\_switch” on page 474](#)

## fn\_pso\_pdf\_admin

The fn\_pso\_pdf\_admin tool is new in the IS 4.0 Itanium Edition and IS 4.0 SP5 releases. It is not included or supported in earlier releases of Image Services.

### Description

The fn\_pso\_pdf\_admin tool is used exclusively during the Programmable Security Object recovery process, and allows you to directly update the master Programmable Object Data File (PODF).

#### Important

---

This tool is highly sensitive, and its misuse could cause your IS system to behave unexpectedly. Please confer with your service representative before using this tool.

---

FileNet security objects include four operating system objects and several relational database objects. When these objects are configured properly, they ensure that FileNet software is robust and secure.

### Use

If it's absolutely necessary to change one or more FileNet security object, then use this tool along with PSO guidelines to perform the task. This tool forcibly updates the master PODF.

### Syntax

**UNIX**

```
fn_pso_pdf_admin [-d <string>] | -h | -?
```

**WIN**

```
fn_pso_pdf_admin [/d <string>] | /h | /?
```

**d** <string> Database type, where <string> can be one of the following:

mssql oracle db2 none

**h** Help screen

**?** Help screen

## Procedure

To change the name of the FileNet software user **fns**w, you must be logged in with **root** or **administrator** privileges.

Follow the appropriate procedure for your platform:

- [“UNIX” on page 470](#)
- [“Windows” on page 472](#)

## UNIX

If any error occurred when using the **fn\_pso\_switch** tool, the master PODF would not be changed and you should still be able to restart Image Services using the previous PODF data. If this is the case, follow these steps:

- a Resolve the error.
- b Log in as **root** user and run **fn\_setup**.
- c Restart Image Services.
- d Run the **fn\_pso\_switch** tool again.

If there were no errors, but Image Services would not start, you would have to change the master PODF data back to the previous state. Image Services should then start again. (If necessary, contact your service representative.) If this is the case, follow these steps:

**In a non-cluster environment:**

- a Resolve the error.
- b Run **fn\_pso\_podf\_admin** tool and enter the original PODF data.
- c Run **fn\_pso\_driver -c** to regenerate all script files.
- d Log in as **root** user and run **fn\_setup**.
- e Restart Image Services.
- f Run the **fn\_pso\_switch** tool again, or contact your service representative.

**In a cluster environment:**

- a Resolve the error.
- b Run **fn\_pso\_podf\_admin** tool and enter the original PODF data on node 1.
- c Run **fn\_pso\_driver -c** to regenerate all scripts on node 1.
- d Log in as **root** user, run **fn\_setup** on node 1.
- e Restart Image Services on node 1.

- f Run the **fn\_pso\_switch** tool again, or contact your service representative.

## Windows

If any error occurred when using the **fn\_pso\_switch** tool, the master PODF would not be changed and you should still be able to restart Image Services using the previous PODF data. If this is the case, follow these steps:

- a Resolve the error.
- b Restart Image Services.
- c Run the **fn\_pso\_switch** tool again.

If there were no errors, but Image Services did not start, then you should change the master PODF data back to the previous state so Image Services will start again. Then contact your service representative. If this is the case, follow these steps:

- a Resolve the error.
- b Run **fn\_pso\_podf\_admin** tool and enter the original PODF data.
- c Run **fn\_pso\_driver /c** to regenerate all script files.
- d Reconfigure your "IS ControlService" service.
- e Restart Image Services.
- f Run the **fn\_pso\_switch** tool again, or contact your service representative.



## Related Topics

[“fn\\_pso\\_driver” on page 467](#)

[“fn\\_pso\\_switch” on page 474](#)

## fn\_pso\_switch

The fn\_pso\_switch tool is new in the IS 4.0 Itanium Edition and IS 4.0 SP5 releases. It is not included or supported in earlier releases of Image Services.

### Description

The fn\_pso\_switch tool enables you to change the user name of the standard FileNet software user, **fns**w.

#### Important

---

The task of changing any security object should be done with great caution. Before you actually change the **fns**w user name, take a few moments to analyze your system, and if you can avoid changing this security object, you should do so. If changing security objects is mandatory, please familiarize yourself with Programmable Security Objects in the *Image Services Systems Administrator's Handbook*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21\*\*](#).

---

### Syntax

**UNIX**

fn\_pso\_switch [ -l ]

**WIN**

fn\_pso\_switch [ /l ]

**l (list)**

List the current user name.

**<no option>**

Launch the procedure to change the FileNet software user name.

## Procedure

To change the name of the FileNet software user **fns**w, you must be logged in with **root** or **administrator** privileges.

- 1 At a command prompt, enter:

**fn\_pso\_switch**

The following messages display:

Checking permissions...  
Verifying IS permission...

- 2 Then you are prompted for the Authorized IS user name and password:

Authorized IS user name:  
Authorized IS password:

Enter the name of the SysAdmin user and the appropriate password.

- 3 When the log in is successful, you see:

Logging in '<server>' as 'SysAdmin...'  
Permission granted!

Enter Command ('m' for menu):

- 4 To see the menu of commands, enter **m** at the prompt.

Command	Description
<b>m/?</b>	display command options
<b>c</b>	change the FileNet user name

Command	Description, Continued
<b>d</b>	display the current settings
<b>e</b>	execute
<b>s</b>	display IS status, release information, and RDB information
<b>t</b>	toggle the password display (DB2 only) (This option is currently disabled.)
<b>q/x</b>	quit

The **c** command allows you to change **fnsw** to any other name and saves it in the local buffer. You could run the **c** option as often as you wish until you are satisfied with the new setting. The change is not saved until you use the **e** option

- 5 For example, to change the **fnsw** user name to **george**, you would enter:

**c george**

If you think this isn't quite right, you can enter a different name:

**c robert**

Someone might guess that name, so perhaps...

**c tenelif**

Only someone who knows that this name is FileNet spelled backwards would guess it.

---

**Tip** Choose a name that is memorable, but not obvious.

---

- 6 Save this user name by entering the **e** command.

- 7 You can verify the current user name setting by entering the **d** command to display its current value.
- 8 When you've finished, enter either the **q** or **x** command to quit the fn\_pso\_switch tool.
- 9 If any errors occur during the fn\_pso\_switch process, refer to the recovery procedures described with [\*\*“fn\\_pso\\_podf\\_admin” on page 469\*\*](#)

## Related Topics

[\*\*“fn\\_pso\\_driver” on page 467\*\*](#)

[\*\*“fn\\_pso\\_podf\\_admin” on page 469\*\*](#)

## fn\_setup

The **fn\_setup** program attempts to set the permissions for all files under /fnsw and /fnsw/local directories using a permission\_table that is updated with each new Image Services release.

For information on fn\_util and its commands, refer to the *System Configuration Overview*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## fn\_util

The fn\_util starts, stops, creates, and updates the MKF and RDBMS databases, calling fn\_oracle, fn\_db2, and fn\_mssql to perform RDBMS configurations.

For information on fn\_util and its commands, refer to the *System Configuration Overview*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## fnddcfg

**WIN**

This **fnddcfg** tool is only available on the Windows Server platform and not available on the UNIX platforms.

### Syntax

```
fnddcfg [-u] [-v] [-f <group>]
```

If no argument is specified, fnddcfg will configure FnScsidd

**-u** Unconfigures FnScsidd.

**-v** Runs in verbose mode

**-f <group>** Specifies the group in SYSTEM\CurrentControlSet\Control\Service-GroupOrder that the FnScsi class should start after.

### Procedure

After FnScsidd is configured or unconfigured, the server needs to be rebooted to put the configuration change into effect.

## fndev

 WIN

This **fndev** tool is only available on the Windows Server platform and not available on the UNIX platforms.

### Syntax

```
fndev [-u | -d] [-v] <dev name> [<port> <path> <target> <lun>]\
```

If no argument is specified, **fndev** will list all SCSI optical disk drives and auto changers detected by the FileNet Windows Server kernel mode class driver.

- u <dev name>** The device will be displayed unmarked in the Windows Server registry. If the device is not in the registry, it will be added unmarked along with the specified SCSI port, path, target and lun, if applicable. Both cases cause the device NOT to be claimed for FileNet use on the next reboot.
- d <dev name>** The device will be deleted from the Windows Server registry.
- <dev name>** The device will be displayed marked in the Windows Server registry. If the device is not in the registry, it will be added marked along with the specified SCSI port, path, target and lun, if applicable. Both cases cause the device to be claimed for FileNet use on the next reboot.
- v** Runs in verbose mode. Information concerning optical disk drives and auto changers are stored under subkey "SYSTEM\CurrentControlSet\Services\FnScsidd\Parameters" and are relative to the HKEY\_LOCAL\_MACHINE in the Windows Server registry.



# FNL\_disp

## Description

The FNL\_disp tool displays interlocks (ILKs) and process control records (PCRs). ILKs and PCRs could be designated by current position, next or previous, absolute index, or pointer. The displayed data is a snapshot of the running system. Be aware that the real data could be changing as you view the snapshot.

## Use

FNL\_disp was designed as a tool for use only by program developers to analyze problems in FileNet process control. Such problems typically require examination of shared memory, semaphores, and interlocks. Occasionally, your service representative could request that you run FNL\_disp to gather information for the problems described above.

---

**Note** Do not use FNL\_disp unless requested to do so by your service representative. You can use the ipc\_tool to examine process control problems.

---

## Syntax

FNL\_disp

Enter commands at the FNL\_disp> prompt.

## Commands

In the command descriptions below, replace items within angle brackets (<>) with an appropriate integer. Items within square brackets

( [ ] ) are optional; items within braces ( { } ) are required. For example, the following syntax indicates that A is required and must be followed by one of the alternatives, B or C:

A { B | C }

In the following syntax example, A is required but selection of either B or C is optional:

A [ B | C ]

- q** Quit the program
- ?** Display the help text
- g** Display the globals

**i** { . | + | - | <dec int> | \* <hex int> }

Display an interlock.

The following parameters display an interlock.

.	current
+	next
-	prior
<dec int>	decimal integer
*<hex int>	address as a hexadecimal integer (Do not prefix <hex int> with leading 0x.)

**p { . | + | - | <dec int> | \*<hex int> }**

Displays a Process Control Record. (See the **i** (interlock) command above for parameter descriptions.)

**h { <hex int> ... <hex int> | <hex int>/<dec int> }**

Displays shared memory. Displays hexadecimal longword and ASCII dump of address range <hex int> ... <hex int> or of <dec int> longwords. (Do not prefix <hex int> with leading 0x.)

Use the scrolling functions at the bottom of the display to move forward and backward through the data.

---

**Tip** You must type the scrolling option character then press Enter.

---

**l <dec int>**

Sets the number of lines per page to <dec int> for the **h** (hex) command display.

---

**s { + | ++ | - | - - } {pid <n> | p { r | c } [ i <n> ] | i c [ l ] }**

Searches for processes or interlocks using the positional parameters described below:

- + forward
- ++ forward, skipping current
- backward
- - backward, skipping current

Searches processes for:

pid <n> A specific process, where <n> is the process ID number  
or

p { r | c } A specific process r or c

where

r is the reader of the current interlock (or interlock <n>), or  
c is the last claimer of the current interlock (or interlock <n>)

Searches interlocks for:

i <n> A specific interlock where <n> is the interlock number

or

i c The interlock for which the last claimer is the current process

or

i l The locked binary interlock for which last claimer is the current process

---

## Sample Output

The examples below use FNL\_disp commands to display the globals, the current interlock, and the next process control record. The last command, **q**, exits the FNL\_disp program.

---

```
corona(root)/fns> FNL_disp
FNL_disp>g
GLOBALS @ 0x50000674:
    avail_pros_p           0x500120C8
    avail_pros_num        100
    pros_fvx              128
    num_pros_chunks       1
    pros_elts_per_chunk   128
    next_p                 0x00000000
    fast_p                 0x00000000
    cs_ilk_oflo_flag       0
    pros_rec_dv[0] = 0x50011878
    avail_ilk              159
    ilk_fvx                256
    num_ilk_chunks        2
    ilk_elts_per_chunk    128
    ilk_rec_dv[0] = 0x50013E80
    ilk_rec_dv[1] = 0x500E8DDC
FNL_disp>
FNL_disp>i .

interlock_rec[0] @ 0x50013E80:
    lockint                0
    claim_count            0
    claimq_p               0x00000000
    cs_count               0
    shareq_p               0x00000000
    validation              0x00000000
    last_claim_pid         0
```

---

```
    handle_addr          0x00000000
    next_pros_ilk_p      0x00000000
    alloc_prosx          pros_rec[0]
    init_cc              0
FNL_disp>p +
pros_rec[1] @ 0x500118C4:
  myinx                 1
  pid                   11773 (alive)
  args                  /fnsw/lib/perf/perf_mon -f/fnsw
  next_p                0x00000000
  prev_p                0x00000000
  wait_ilk_p            0x00000000
  wait_type             0 (UNKNOWN)
  trying                0 (FALSE)
  pros_owned_ilk_p:    0x00000000
  cs_ilk [0]:           0x00000000
  cs_ilk [1]:           0x00000000
  cs_ilk [2]:           0x00000000
FNL_disp>q
corona(root)/fnsw>
```

---

## Checklist

Before you use FNL\_disp, be aware of the following:

- Typically, you use ipc\_tool to obtain debugging information about interlocks and process control records. However, your service representative could request that you run FNL\_disp for other reasons.
- The real data could be changing as you view the snapshot taken by FNL\_disp.
- You can safely run FNL\_disp while your system is active. Since FNL\_disp does not obtain locks during operation, you can also safely terminate FNL\_disp prior to its completion. To prematurely terminate FNL\_disp, use one of the following methods:
- If ddexim is running in the foreground, press the Control+c key sequence to abort ddexim. If running in the background, enter one of the following commands depending upon the server's operating system, supplying the process ID (<pid>) for ddexim:
  - On UNIX platforms, use one of the following commands where <pid> is the process ID of the FNL\_disp process.

**AIX**

**kill -30 <pid>**

**HPUX  
SOL**

**kill -16 <pid>**

**WIN**

- On Windows Server platforms, use the **pview** command or the Task Manager to terminate the FNL\_disp process.

See your Windows Server manuals for information about pview and the *System Administrator's Handbook* for information about Task Manager functions. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## Procedure

No specific procedure is required.

## Related Topics

[\*\*“ipc\\_tool” on page 610\*\*](#)



# fnlogon

## Description

**UNIX**

The fnlogon tool is available only on a UNIX-based Image Services server.

The fnlogon tool provides a security interface to the following FileNet tools: CSM\_exim, CSM\_tool, ddexim, DOC\_tool, MKF\_tool, oddump, PRI\_tool, SEC\_tool, and WQS\_tool. You can run some tools without using fnlogon if you are logged on as the root or fnsd user or if you are logged on as a member of the fnadmin group with SysAdmin privileges.

At successful logon, the tool you want to access validates your logon information.

## Use

Use fnlogon to logon to the FileNet security interface when you need to use tools that require this interface.

When you enter fnlogon at the command line, a series of prompts request a FileNet logon name, a FileNet logon password, a security service name, and the program name you want to run.

Enter your FileNet logon and password. At the prompt for security service name, press Enter to access the local system. If you want to access a networked system, enter the name of the security system for the networked system (for example, Security-System:<domain>:FileNet). We do not recommend that you log on to a remote security system.

**Tip** We generally recommend that you do not specify a security service name. If you do not specify a security service name, fnlogon connects you to the local domain's security service. Connecting to the local domain's security service eliminates difficulties if you subsequently run applications that are not designed for remote operation.

---

At the program prompt, you can enter the tool you want to access or press Enter. If you do not specify a tool name at the program prompt, fnlogon runs your default shell and a system prompt appears. At the system prompt, you can enter **q** (or **quit**) to logoff the security system or enter the tool name you want to access. The requested tool issues calls to validate the logon security information that fnlogon passed to it. When you quit the tool, you can run another tool that requires fnlogon without re-running fnlogon itself. However, if you entered a tool name at the program prompt, when you subsequently exit that tool, you also exit the fnlogon program. To run any other tool that requires fnlogon, you must re-run the fnlogon tool.

## Syntax

fnlogon

## Example

In the example below, fnlogon issues prompts to which you must respond.

```
$ fnlogon

FileNet user name: SysAdmin
FileNet password:
FileNet security service (CR = local service):
Program (CR = default shell): SEC_tool
```

In this example, the user pressed the Enter key at the FileNet security service prompt to connect to the local security service (recommended). The SEC\_tool was selected at the program prompt.

You could also respond to the program prompt by pressing the Enter before you enter a tool name from the system prompt.

```
$ fnlogon

FileNet user name: SysAdmin
FileNet password:
FileNet security service (CR = local service):
Program (CR = default shell):
$ SEC_tool
SEC_tool>
```

If you access a tool as described in the example directly above, you could quit the tool and enter another tool name at the system prompt without re-running fnlogon. Enter **exit** to completely exit from fnlogon.

The following example shows an exit first from the SEC\_tool with the **quit** command (after which the system prompt displays), then from the

fnlogon program with the **exit** command, and finally from the system, using the **logout** command.

```
<SEC_tool>quit
$ exit

fnlogon: logging off...
$ logout
```

## Checklist

Before you use fnlogon, be aware of the following:

- If you default to the shell by pressing Enter at the program prompt, you are inside another shell from which you must exit (by entering **exit**) before you can log out.
- The Control+d key sequence also exits fnlogon.

## Procedure

- 1 Enter **fnlogon** at the system prompt.
- 2 Respond to the prompts.

## Related Topics

See the “Security Administration” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

See individual tools listed in [“Description” on page 489](#).

# gaddr

## Description

**AIX**

The gaddr tool is supported on Image Services for AIX/6000 systems only.

The gaddr command displays the host addresses of those hosts accessible to the glogin and gsh tools. The system stores the host addresses in files in the directory /usr/FileNet/glogin. The gloginsvr program, which starts automatically during FileNet software initialization, keeps these host addresses up to date.

If you invoke gaddr with no arguments, the displayed information includes:

- Host name
- Server number
- Network address for each server with a nonzero host number
- Time and date of the last modification to each host address file

If you supply host names, gaddr displays information about only those hosts.

An option is available that identifies those host addresses which appear in more than one host file and another option removes address files for named hosts. The gaddr tool broadcasts requests to remove the address files from remote machines as well.

## Use

The gaddr tool displays the addresses of the host machines that are accessible with gsh. Use gaddr to determine host configuration information or to spot certain configuration problems such as duplicate addresses.

## Syntax

```
gaddr [-d] [-r] [<file name> ...<file name>]  
      [<hostname> ... <hostname>]
```

- d** Identifies and displays duplicate addresses. Use this option to investigate potential configuration problems caused by duplicate addresses.
- r** Removes named files and broadcasts the remove request to remove the address files on remote machines

- <file name>** One or more file names to remove with the -r option
- <host name>** One or more host names to display

## Sample Output

In the example below, gaddr is run first without options, then with the -d option to search for duplicate addresses. The last example requests information for only two hosts, costa1 and costaNLS.

---

**Note** In the following sample output, ellipses (...) indicates that only a portion of the entire listing is included in this document.

---

The output consists of the host name, the server number of the host, and the network address for each server with a nonzero host number. The date and time stamp represents when the last modification to the address file was made.

---

```

corona<fnsw> gaddr
...
costal1          1 00000001.42608c2f2dba Tue Dec 10 12:57:44 1996
costal10         1 00000001.02608c2cdf3a Tue Dec 10 12:57:41 1996
costal11         1 00000001.08005afc5d84 Wed Dec 11 14:52:20 1996
costal12         1 00000001.10005afa4e65 Tue Dec 10 12:59:02 1996
costalAPP        1 00000001.42608c2f6022 Tue Dec 10 12:57:47 1996
costa5           1 00000001.02608c2cb5d2 Tue Dec 10 15:06:49 1996
costa5tr         1 00000001.02608c2cb5d2 Wed Nov 27 10:35:58 1996
costa6           1 00000001.08005acd5da7 Tue Dec 10 12:57:38 1996
costa6app        1 00000001.02608c2f5f9d Tue Dec 10 12:58:33 1996
costa7           1 00000001.08005a4d19ed Wed Nov 27 10:35:47 1996
costa8           1 00000001.08005a4d13d8 Wed Nov 27 10:35:56 1996
costa9           1 00000001.02608c2f2ba1 Wed Nov 27 10:35:47 1996
costaAIX         1 00000001.02608c2fbdd2 Tue Dec 10 12:57:57 1996
costaCONV        1 00000001.02608c2f5f9d Tue Dec 10 12:58:15 1996
costaNLS         1 00000001.02608c2cb07f Tue Dec 10 12:58:06 1996
...
corona> gaddr -d
...
SSE_TRAIN        1 00000000.080034000303
SSE_dan          1 00000000.080034000303
p45              51 00000000.080034000413
opus             1 00000000.080034000413
alf              2 00000000.080034000426
seg3m           1 00000000.0800340009ad
pcprint         1 00000000.0800340009ad
SSE_Train       6 00000000.080034000b1c
costa5tr        1 00000001.02608c2cb5d2
costa5           1 00000001.02608c2cb5d2
...
corona> gaddr costal costaNLS
costal1          1 00000001.42608c2f2dba Tue Dec 10 12:57:44 1996
costaNLS         1 00000001.02608c2cb07f Tue Dec 10 12:58:06 1996

```

---



## Checklist

Before using gaddr, be aware of the following:

- The -r option removes specified host address files from remote machines as well as local machines.
- To get online help, enter

**gaddr -**

## Procedure

No specific procedure is required.

## Related Topics

[\*\*“glogin” on page 565\*\*](#)

[\*\*“gsh” on page 570\*\*](#)

# gcp

## Description

The gcp tool copies files between systems. Each file or directory argument is either a remote file name of the form **rhost:path** or a local file name containing no **:path**. If a full path name is not specified for a remote file or directory, a default path name is interpreted relative to a directory on the remote host as follows: The gcp tool uses the string value of the REMOTEDIR environment variable, if set; otherwise is uses /tmp.

You do not have to have an account at the remote host to use gcp. Files created on a remote host belong to the user running gcp if the host has an account with the same name. If one does not exist, the owner of the file is "daemon." Requests to read files on remote hosts are subject to security checking relative to the same account; otherwise, the file must have read permission enabled for all users.

When the copy operation completes, use the ls (list) command to verify that you received the file. You can also use the sum command to examine the byte count of the received file. The sum command is generally used to determine if a file that has been copied or sent over transmission lines is an exact copy of the original. Compare this byte count with the byte count of the file on the source system to verify that they are equal. An example of the sum command follows:

```
corona(root)/> sum /fnsw/local/tmp/diane/chkpgsmp1  
58254      66 /fnsw/local/tmp/diane/chkpgsmp1  
corona(root)/>
```

If you want the name of the received file to be the same name as the source file, enter that name in the <file name2> parameter. However, if you enter the name of the receiving file incorrectly, gcp creates a new file of that name. No message displays telling you that the new file has been created. This could result in what appears to be a “lost” file. To avoid this, you can use a period (.) in the <file name2> parameter instead of a file name. Then gcp automatically names the received file the same name as the source file. In the example below, we use a period for <file name2> and gcp names the received file /fnsw/local/tmp/diane/chkpgsmp1. Then we use the ls (list) command to verify that the received file has the correct name.

```
corona(root)/> ls /fnsw/local/tmp/diane
log12094      testlog
corona(root)/> gcp norco:/tmp/chkpgsmp1 /fnsw/local/tmp/diane/.
corona(root)/> ls /fnsw/local/tmp/diane
chkpgsmp1    log12094     testlog
```

## Use

Use gcp to copy files between two systems on the same or a different local area network.

## Syntax

```
gcp [ options ] <file name1> [ <file name2> | <directory> ]
```

- <file name1>** The local or remote file to copy from (the source file).  
The format of a remote file name is rhost:path where rhost is the remote host name and path is the full path name to a remote file or directory. (See [“Checklist” on page 502](#) for the default.)
- <file name2>** The local or remote file to copy to (the target file).  
The format of a remote file name is rhost:path where rhost is the remote host name and path is the full path name to a remote file or directory. (See [“Checklist” on page 502](#) for the default.) You can also use a period (.) to force gcp to name the target file the same as the source file.
- <directory>** The directory to receive the copied files. (See [“Checklist” on page 502](#) for the default.)
- <options>** Valid options for special handling are listed below:

Option	Description
<b>-h</b>	Print the RCS header
<b>-i</b>	Prompt with the name of the file whenever the copy overwrites an existing file of the same name. Enter <b>y</b> to continue and overwrite the file. Press any other key to prevent overwriting the file.
<b>-l</b>	Use long block size to read and write data. Default block size is 1024 bytes.
<b>-s</b>	Use short block size to read and write data. Default size is 512 bytes less the packet overhead

Option, Continued	Description
<b>-v</b>	Verbose mode, which continuously displays the transferred byte count. This option can slow the progress of the copy operation but helps you monitor progress for large files.
<b>-n#</b>	Sets destination network number (specified by #). As a result, gcp can copy files through a router to a different local area network (LAN).

## Examples

In the following example, the file named `chkpgsmp1` is copied from the remote host named `norco` into a directory `/fnsw/local/tmp/diane` on the receiving system. You can verify that the file was received by using the `ls` command to list the contents of the directory that received the file.

```
corona(root)/> gcp norco:/tmp/chkpgsmp1 /fnsw/local/tmp/diane/chkpgsmp1
corona(root)/> ls /fnsw/local/tmp/diane
chkpgsmp1
```

Use a period (`.`) for the target file name and `gcp` automatically names the received file the same name as the source file.

```
corona(root)/> gcp norco:/tmp/chkpgsmp1 /fnsw/local/tmp/diane/.
```

## Checklist

Before you use gcp, be aware of the following:

- If you specify the file name or directory name in the gcp command, and do not specify a full path name for a remote file or directory, the default path is interpreted relative to a directory on the remote host as follows: If you set the REMOTEDIR environment variable, gcp uses that string value as the default path; otherwise it uses /tmp.
- A copy operation is not performed if neither source nor target files exist on the current system.
- Before you run gcp, gcpshr must be running on the remote host.

## Procedure

No specific procedure is required.

## Related Topics

Is command (See the online manual pages or the “Commands” chapter of your *System Administrator’s Handbook*.) To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

sum command (Use the appropriate UNIX **man** command to display the related on-line manual pages.)

# GDBcheckdb

## Description

GDBcheckdb is a tool called internally by FileNet startup software to check for the logon status of the relational database management system (RDBMS) before logging on to the RDBMS.

GDBcheckdb terminates if it encounters an error.

## Use

FileNet software calls GDBcheckdb internally during RDBMS startup. Occasionally, you can run this command manually for problem determination. For example, if your system is encountering RDBMS logon errors, you can use the GDBdebug and GDBcheckdb tools to examine the errors as part of a problem determination strategy. GDBcheckdb ensures that the RDBMS can be logged on to, which is typically the first verification to perform in the debugging process before you proceed with the analysis. (See [“GDBdebug” on page 505.](#))

## Syntax

```
GDBcheckdb [-l | -w]
```

- w** Retries the logon every 15 seconds. Mutually exclusive with the **-l** option.
- l** Verifies that the FileNet database can be logged on to. One attempt is made to log on. If the logon is successful, GDBcheckdb completes. If unsuccessful, GDBcheckdb logs an error and exits. Mutually exclusive with the **-w** option.

## Checklist

Before you use GDBcheckdb, be aware that this tool is not intended to be run manually. Use this tool from the command line only during problem determination of RDBMS logon failures.

## Procedure

To check for the logon status of the RDBMS during problem determination, follow these steps:

- 1 Use GDBdebug to turn on debugging.
- 2 Start the relational database management system.

You can use a number of methods such as:

- fn\_util starttrdb command
- FileNet scripts (fn\_oracle, fn\_sql, or fn\_db2)
- native RDBMS procedures

- 3 Enter **GDBcheckdb** at the command line.

Call your service representative if you need assistance with problem determination.

## Related Topics

[\*\*“GDBdebug” on page 505\*\*](#)



# GDBdebug

## Description

The GDBdebug tool traces relational database operations and writes trace output to a file that you can browse with a text editor.

---

**Note** GDBdebug is intended for use **only** at the direction of your service representative.

---

Trace options are set with flags, using hexadecimal values for the events to be traced. Each hexadecimal value represents an entry point in the generic database (GDB) program.

The default file to receive trace output is one of the following:

`/fnsw/local/logs/gdb_logs/GDBdbg<pid>` for UNIX platforms

`<drive>:\fnsw_loc\logs\gdb_logs\GDBdbg<pid>` for Windows  
Server platforms

where `<pid>` is the process ID of the process calling the GDB functions.

### Changing the Default Path and File Name

You can change the location of the default trace file by setting the `GDB_DEBUG_FILE` environment variable to another directory path and file name or you can redirect the output to the standard output device. (The `GDB_DEBUG_FILE` environment variable applies only to the processes started in the current shell.)

**CAUTION**

Trace files can be very large. Use GDBdebug only when necessary and turn the trace off as soon as possible after debugging information is collected.

---

For example, to set the GDB\_DEBUG\_FILE environment variable, follow these steps:

- a On UNIX servers, enter the appropriate command for your shell:

```
export GDB_DEBUG_FILE=/tmp/<filename> Bourne/Korn shell
```

```
setenv GDB_DEBUG_FILE /tmp/<filename> C shell
```

On Windows Servers, go to Control Panel > System > Advanced tab > Environment Variables.

- b Shutdown and restart the Image Services software.

**Deadlock Checking**

GDBdebug supports deadlock checking and retry logic for GDB functions. Trace functions allow callers to determine if the error they received was a retryable deadlock error or not. When deadlock errors occur, a half-second delay is introduced to allow the competing process to progress and give up any of the involved locks. You can reset the default delay with GDBdebug -d. If all available retries fail, GDBdebug writes an informational message to the event log. It allows up to 20 retries; however, you can reset the number of retries on a server-wide basis with GDBdebug -r.

GDBdebug includes a deadlock generation facility for testing of deadlock retry logic. This function is only used by support personnel. For more information, call your service representative for instructions and information.

## Use

Use GDBdebug at the direction of your service representative to collect trace information when you are investigating problems in the generic database (GDB) subsystem.

The most common use of GDBdebug is to trace an application program. The trace results are most accurate when only a single client is accesses the FileNet system.

## Syntax

To print the complete syntax for GDBdebug, enter GDBdebug with no parameters at the command line.

```
GDBdebug {-g | -s <value> | -d <#> | -r <#> }
```

- g** Retrieves and displays the value of the current debug flag setting as well as the current deadlock delay value and the maximum number of deadlock retry attempts
- s <value>** Sets the debug flag with the hexadecimal number specified by <value>. See the online help statement for GDBdebug or [“GDBdebug Values” on page 508](#).

---

**Tip** The most commonly-used setting is FFFF (all bits turned on). Call your service representative to obtain the correct hexadecimal flag setting for the type of trace necessary.

---

Setting the value to zero (0) turns off the trace.

- d <#>** Sets the debug deadlock delay time to the value specified by #, where # is in 100ths of a second. For example, 20 = 1/5 second; 300 = 3 seconds.
- r <#>** Sets the maximum number of debug deadlock retry attempts. The default value is 20.

## GDBdebug Values

The following table lists the hexadecimal bit settings and describes the function to be traced for each setting:

Bit Setting	GDB Function
0x00000001	GDB_logon
0x00000002	GDB_logoff
0x00000004	GDB_alloc_stmt
0x00000008	GDB_free_stmt
0x00000010	GDB_prepare
0x00000020	GDB_in_bind
0x00000040	GDB_out_bind
0x00000080	GDB_exec
0x00000100	GDB_fetch
0x00000200	GDB_desc
0x00000400	GDB_begin_trans
0x00000800	GDB_end_trans
0x00001000	GDB_stat
0x00002000	GDB_errmsg
0x00004000	GDB_cancel_query
0x00008000	Other functions
0x00010000	DBMS entry calls
0x00020000	Print full blob
0x40000000	Deadlock testing

## Example

The following examples show how to obtain the current setting for the GDBdebug flag and how to reset the flag if directed to do so by your service representative. In this example, the flag setting is displayed and shows that the trace is off (debug: 0x0). The flag is set to hexadecimal value '10' and the flag is once again displayed to verify the setting. Finally, the flag is reset to turn off tracing and again verified for the correct setting:

```
> GDBdebug -g
debug: 0x0
deadlock delay: 50 (secs/100)
deadlock max retries: 20
> GDBdebug -s 10
> GDBdebug -g
debug: 0x10
deadlock delay: 50 (secs/100)
deadlock max retries: 20
> GDBdebug -s 0
> GDBdebug -g
debug: 0x0
deadlock delay: 50 (secs/100)
deadlock max retries: 20
```

## Checklist

Before you use GDBdebug, be aware of the following:

- You should use GDBdebug only when directed by your service representative.
- GDBdebug trace files can be very large. Use GDBdebug only when necessary and only long enough to obtain required output.

## Procedure

Call your service representative for the procedure to run GDBdebug and the correct hexadecimal debug value setting for the specific problem being investigated.

See also the procedure under [\*\*“GDBcheckdb” on page 503.\*\*](#)

# GDB\_exim

## Description

GDB\_exim provides generic export and import functions for the purpose of moving data between relational database managements systems (RDBMSs) or between servers.

---

**CAUTION** GDB\_exim is not designed to be used by the customer. Complete procedures for running GDB\_exim are available only from your service representative.

---

GDB\_exim provides the following main functions:

- Export data from a database to external files
- Import data from external files to a database
- Transfer data from one server to another via tape files, magnetic disk files, or over a network

GDB\_exim can add or delete columns from indexes during its processing. When you add columns to the index, GDB\_exim automatically prevents duplicate columns. You can specify indexes as nonclustered. If you select columns for deletion that did not exist originally, GDB\_exim does not generate an error.

If GDB\_exim is interrupted during export or import, it can be restarted. Information about interrupted jobs is kept in restart files. These files allow the operation to continue at the point of interruption. Since they have already been processed, the original input files are largely ignored at restart and input is read from the restart files instead.

**Note** If the export or import is taking place over a network, the job might have to be reinitiated if the sending and receiving systems detect a discrepancy in the point of restart.

---

### Command Files

GDB\_exim performs functions based on information found in command files. Command files use a definition language to specify what to export or import and what changes to make between RDBMSs and what changes to make (for example, adding or dropping columns or changing storage parameters).

The export function reads the contents of the commands file and creates a definitions file that describes the exported data which the data files contain. Commands in the export file control the sequencing of the operation. Each step in the operation has a name. The step name can be used for restart, if necessary, and export/import synchronization. The name becomes the root of the file names that are passed from export to import (via definitions files and data files).



## Rules Files

The import function reads the commands file and rules file, which provide a higher degree of control by overriding the definitions files. Since definitions files represent the “old” database, the rules files provide a way to describe the differences to be reflected in the “new” database. A standard rules file is normally used, however, you can modify this file to meet customization requirements or special circumstances.

You can add or delete columns from indexes by specifying index properties in the rules file. If a column you are attempting to add already exists, GDB\_exim automatically prevents the addition of a duplicate column. You can specify that indexes be nonclustered (for Oracle environments).

## Job File Names

The following table describes the job file names that GDB\_exim uses.

Job File Name	Description
<jobname>.exp	The export command file names the steps of the job and the databases, owners, and tables to be exported. You create this text file and it can be any descriptive name (see <a href="#">Sample.exp</a> below). GDB_exim uses the .exp file as an input parameter to export RDBMS.
<stepname>.def	This definitions file specifies the details of tables, columns, and indexes as reflected in the source database. GDB_exim export creates this file.
<stepname>.dat	This data file contains the data of the exported tables. GDB_exim export creates this file.
<stepname>.nnn	This data file contains sections of tables that are exported by ranges specified by nnn, where nnn can be 001 to 999. Restarts can occur at the beginning of a range. GDB_exim export creates this file.

Job File Name	Description
<jobname>.imp	The import command file names the steps of the import job. You create this file (see <b>Sample.imp</b> below). Except when running the network option of GDB_exim, the file contains the import options. If you are running the network option, specify the import options in the <jobname>.exp file so that export can pass them to the import job via the network.
<jobname>.rul	The rules file supplies any modifications to the definitions that could be required to successfully import the data. Rules are typically needed when importing across RDBMSs or to make changes to the data (such as adding or deleting columns, specifying nondefault data types, and specifying null/non-null.) You can modify this file if necessary (see <b>Sample.rul</b> below). This rules file will cover almost everything you want different from the exported definitions, or to override the system defaults.
<jobname>.rse	The export restart file contains information about an export job in progress. GDB_exim export creates this file.
<jobname>.rsi	The import restart file contains information about an import job in progress. GDB_exim import creates this file.
<jobname>.elg	The export log file contains much of the screen output generated by the running export job. These files could be useful for problem determination if the job fails. GDB_exim export creates this file.
<jobname>.ilg	The import log file contains much of the screen output generated by the running import job. These files could be useful for problem determination if the job fails. GDB_exim import creates this file.

## User-defined Job File Examples

### Sample.exp

In this sample .exp file, GDB\_exim -e will export all tables owned by f\_sw under the “indexdb” database and generate a pair of export1 files: export1.def and export1.dat.

```
options (disk '.')
step export1:
export
(
  database indexdb
  (
    owner f_sw
    (
      table *
    )
  )
)
```

### Sample.imp

In this sample .imp file, verify that you copy a pair of files (export1.def and export1.dat) from the server that you exported to the server that you want to import data to. As the first line of the example indicates, Import will first drop the old, existing tables before creating the new tables.

```
options (disk '.', existing_tables drop)
step export1 % export1 matches with the export one in sample.def
```

## Sample.rul

In this sample .rul file, data will be imported from another RDBMS (mssql/db2) and automatically change the tablespace name to be SITE\_SYS\_TS to fit with Oracle. This is optional and you can always change the schema (.def) file to test the import. .

```
oracle
(
  database indexdb
  {
    owner f_sw
    [
      table * (location SITE_SYS_TS)          % Refers to Oracle tablespace
    ]
  }
)
```

## Use

You can use GDB\_exim to move data in your RDBMS databases between different RDBMSs or to move data when converting from one release of your RDBMS to another. You can also use GDB\_exim to move data between databases on different servers.

## Syntax

---

**Tip** Enter the following command to display the syntax:  
**GDB\_exim -**

---

GDB\_exim <options>

where <options> are the following:

- e <jobname>** Exports data using the command file specified by <jobname> in the format <jobname>.exp. Mutually exclusive with the -i or -l options.
- i <jobname>** Imports data using the command file specified by <jobname> in the format <jobname>.imp. Mutually exclusive with the -e or -l options.
- l** Import via network (listen). Must be specified alone or with the -p option. Mutually exclusive with the -e or -i options.

---

**Note** You must specify one of the options: -e, -i, or -l.

---

- n** When specified with import, shows only contents of the data file and definitions files. No data is imported. When specified with export, estimates table size in the database. No data is exported.
- p <portnum>** Specifies a network port number to use if the default port is unsuitable.
- t <capacity>** Number of megabytes per tape volume. This option is required for export only. The default value is 200.

## Checklist

Before you use GDB\_exim, be aware of the following:

- GDB\_exim is not designed for customer use.
- You must contact your service representative for complete procedures and for assistance in creating command files or modifying the rules file.
- If GDB\_exim is interrupted during an export or import operation, the job can be restarted.

## Procedure

Before you run GDB\_exim, call your service representative for complete procedures. You must create the required command and definitions files. You can modify the rules file for import operations. Use GDB\_exim only at the direction of your service representative.

## Related Topics

See the *Index and WorkFlo Database Contents Manual* for information about the index database tables. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## get\_rdbms

### Description

The get\_rdbms tool gathers performance and configuration information about your RDBMS. The information is a snapshot of index and Workflo Queue database configuration and statistics; rollback and data dictionary activity; session and system-wide activity; space use; table and tablespace information; index information; and Oracle Server Manager reports.

---

**Note** For IBM DB2, this tool is not supported.

---

#### UNIX

The get\_rdbms tool runs on the AIX/6000, HP-UX, and Solaris Operating Environment platforms only.

You can gather information on all categories of RDBMS information or optionally select individual report types. Once you select your options, the get\_rdbms script gathers the appropriate information and writes the formatted information to the output file.

The get\_rdbms tool writes output to the directory you specify or to the default directory /fnsw/local/logs/perf. The output file name format is:

rdbms.<n>.<mmdd>

where <n> is the local station number and <mmdd> is the current month and day (for example, Jun15).

## Use

Use `get_rdbms` to collect a snapshot of your RDBMS performance and configuration statistics in a single file.

You can collect all reports or individual reports by category. For example, if you suspect individual tables are fragmented in the database, select the `get_rdbms` space use option and examine the DBA Extents information in the report.

## Syntax

```
get_rdbms [-o <outfile>] [-od <outdir>] [-a] | [-i] [-p] [-s] [-t] [-I]
```

- o <outfile>** Specifies the output file. The default is `rdbms.<n>.<mmdd>` where `<n>` is the local station number and `<mmdd>` is the current month and day.
- od <outdir>** Specifies the output directory. The default is `/fnsw/local/logs/perf`.
- a** Creates all reports (index, performance, space, table and Oracle Server Manager). This option is mutually exclusive with individual report types.
- i** Creates index reports
- p** Creates performance reports
- s** Creates space use reports
- t** Creates table reports
- I** Creates Oracle Server Manager parameter reports for Oracle



---

**Tip** The option for parameter reports is specified as a capital **I**, not a lower case **L**.

---

## Sample Output

The following sample output is from `get_rdbms -I` for Oracle Server Manager parameter reports (output is truncated for readability):

---

```
get_rdbms -I - Fri Sep 3 10:14:23 PDT 1999
>>> S Q L D B A      R E P O R T S <<<

Oracle Server Manager Release 3.0.4.0.0 - Production

(c) Copyright 1997, Oracle Corporation.  All Rights Reserved.

Oracle8 Enterprise Edition Release 8.0.4.0.0 - Production
PL/SQL Release 8.0.4.0.0 - Production

SVRMGR> Connected.
SVRMGR> Total System Global Area          6568364 bytes
Fixed Size                               38940 bytes
Variable Size                             4448656 bytes
Database Buffers                          2048000 bytes
Redo Buffers                               32768 bytes
SVRMGR> NAME                                TYPE      VALUE
-----
always_anti_join                           string    NESTED_LOOPS
audit_file_dest                             string    ?/rdbms/audit
audit_trail                                 string    NONE
b_tree_bitmap_plans                         boolean   FALSE
background_core_dump                        string    full
background_dump_dest                        string    ?/rdbms/log
bitmap_merge_area_size                      integer  1048576
```

```
blank_trimming                boolean FALSE
cache_size_threshold          integer 100
ccf_io_size                   integer 65536
checkpoint_process            boolean FALSE
cleanup_rollback_entries      integer 20
close_cached_open_cursors     boolean FALSE
SVRMGR>
Server Manager complete.
```

---

## Procedure

- 1 Run get\_rdbms with desired options.
- 2 Use a text editor to view the output file.

## Checklist

Before you use get\_rdbms, be aware of the following:

- To display online help for the command syntax, enter get\_rdbms followed by a hyphen:

**get\_rdbms -**

- get\_rdbms checks to see if there is a password set in the environment for f\_maint. If one has not been set, you will be prompted for the f\_maint password.

## Related Topics

**[“sgs” on page 1183](#)**

See your RDBMS documentation for information about Oracle Server Manager.

## getreports

**UNIX**

The getreports tool is available only on a UNIX-based Image Services server. To generate reports on a Windows server, use perf\_report.

### Description

The getreports tool is a UNIX shell script that runs multiple instances of the perf\_report tool to collect the necessary soft reports for each type of server. The getreports tool:

- Calls the perf\_report tool to examine the type of server from the information found in the reports file
- Creates appropriate reports based on server type
- Builds the reports

The perf\_mon utility collects system performance statistics on the current Image Services server at configurable intervals (by default, every 15 minutes from 6:00 AM to 7:00 PM, and every two hours during other times). It then stores the data in the perflog file.

The perf\_report tool uses the appropriate report definition file to format the report data it retrieves from the perflog file. It then creates the output report files with a naming convention that includes the server type, server number, report type, and date.

The getreports tool stores the reports as text files in /fnsw/local/logs/perf. If the directory does not exist, getreports creates it. Each time you run getreports, it uses perf\_report to generate a new set of report files, then stores the files in the same directory.

You can view the reports using a UNIX command, such as `less` or `more`, or by opening it in a text editor, such as `vi`.

The report file naming convention is:

`<server type><station number>.<report type>.MonDD.txt`

**server type** One of the following designations:

<code>cmb</code>	Combined server
<code>inx</code>	Root/Index server or Index server only
<code>doc</code>	Document server
<code>os</code>	Root/Storage Library server or Storage Library server only
<code>app</code>	Application server

**server number** Number of the server

If `getreports` cannot obtain the server number for a particular server, it uses server number 99.

**report type** Report type. Full reports have one of the following designations:

<code>ful</code>	Full report
<code>abs</code>	Full absolute report

The `perf_report` tool also generates subsystem-specific reports. For a complete list of predefined soft reports, see [\*\*“Predefined Soft Reports” on page 973.\*\*](#)

**MonDD** Month and day the report is generated

**txt** Extension added for compatibility with editors on a Windows platform or for viewing on any PC

As an example, the name of an output report file generated for server 1, a Combined server, reporting document services performance information for December 10 is "cmb1.ds.Dec10.txt"

---

**Note** See the perf\_mon and perf\_report tools for a full description of all the report types.

---

The getreports tool also creates status messages. It writes status messages to a log file, rptn.stat.<mmdd>.txt, where **n** is a report number and <mmdd> is the date of the report in month and day format. For example, **rpt1.stat.Dec10.txt** is the file containing status information from December 10. It writes log files to the /fnsw/local/logs/perf directory.

The information written to the status message log file indicates the following:

- Date and time that getreports was started and/or abnormally terminated
- Configuration file used to generate reports
- System for which reports are being generated
- Types of reports generated
- File information on the report files created

The following example shows a getreports status file:

---

```

elcapitan(fnsw)/fnsw/local/logs/perf> less rpt1.stat.Aug31
Starting getreports Tue Aug 31 10:25:12 PDT 1999
ReportsDir=/fnsw/local/logs/perf/
REPORTS GENERATED

```

Name	Title	File
asum	AIX Processor Summary Report	cmb1.asum.Aug31.txt
vmm	AIX Virtual Memory Activity	cmb1.vmm.Aug31.txt
atcp	AIX TCP Net Activity	cmb1.atcp.Aug31.txt
css	CSS Summary Report	cmb1.css.Aug31.txt
sv	Service activity	cmb1.sv.Aug31.txt
bs	Batch Entry Report	cmb1.bs.Aug31.txt
rbs	Batch Entry Server Report	cmb1.rbs.Aug31.txt
transdb_io	Transient DB I/O Tuning Report	cmb1.transdb_io.Aug31.txt
transdb_trans	Transient DB Transaction Tuning	cmb1.transdb_trans.Aug31.txt
csm	Cache Services Report	cmb1.csm.Aug31.txt
doc	Document Services Report	cmb1.doc.Aug31.txt
inx	Index Services Report	cmb1.inx.Aug31.txt
nch	Network Clearinghouse Report	cmb1.nch.Aug31.txt
ds	Document Services Summary	cmb1.ds.Aug31.txt
pf	Image Prefetch Activity Report	cmb1.pf.Aug31.txt
ol	OSAR Load Summary	cmb1.ol.Aug31.txt
permdb_io	Permanent DB I/O Tuning Report	cmb1.permdb_io.Aug31.txt
permdb_trans	Permanent DB Transaction Tuning	cmb1.permdb_trans.Aug31.txt
prt	Print Activity Report	cmb1.prt.Aug31.txt
wfl	WorkFlo Activity Report	cmb1.wfl.Aug31.txt
sq	SQL Services Report	cmb1.sq.Aug31.txt
sec	Security Services Report	cmb1.sec.Aug31.txt
secdb_io	Security DB I/O Tuning Report	cmb1.secdb_io.Aug31.txt
secdb_trans	Security DB Transaction Tuning	cmb1.secdb_trans.Aug31.txt

```

The following reports were generated:
-rw-rw-r-- 1 fnsw  fnsur      2203 Aug 31 10:25 /fnsw/local/logs/perf/rpt1
.stat.Aug31
Exiting getreports Tue Aug 31 10:25:12 PDT 1999

```

---

## Use

Use getreports to generate performance reports based on the server type.

Use optional switch settings to limit output from getreports. To display a list of optional switch settings, enter the getreports command followed by a space and two hyphens (--), as shown below:

```
elcapitan(fnsw)/fnsw/local/logs/perf> getreports --
getreports [-a] [-v] [-r reportdir] [perfargs]

-a          all reports all servers
-v          verbose mode
-r reportdir produce output in reportdir
            (Default is /fnsw/local/logs/perf)
perfargs    perf_report arguments
```

You can further limit your output by typing a hyphen followed by one of the perf\_report arguments. You can set the perf\_report argument variables from the command line.

See [“perf\\_report” on page 968](#) for a detailed discussion of the arguments. Some of the more commonly-used arguments are:

- -br to set a date on which to start statistics averaging
- -er to set a date on which to end statistics averaging
- -b to set the average interval begin time
- -e to set an average interval end time
- -d to set the maximum number of daily reports

As an example, to limit reports to one day, enter:

### **getreports -d1**

The default is all report types for the day you invoke the getreport command (in this case, August 31). The following output displays a list of generated reports that result from the -d1 switch setting.

```
elcapitan(fnsw)/fnsw/local/logs/perf> ls *Aug31*
cmb1.asum.Aug31.txt          cmb1.pf.Aug31.txt
cmb1.atcp.Aug31.txt          cmb1.prt.Aug31.txt
cmb1.axns.Aug31.txt          cmb1.rbs.Aug31.txt
cmb1.bs.Aug31.txt            cmb1.sec.Aug31.txt
cmb1.csm.Aug31.txt           cmb1.secdb_io.Aug31.txt
cmb1.css.Aug31.txt           cmb1.secdb_trans.Aug31.txt
cmb1.doc.Aug31.txt           cmb1.sql.Aug31.txt
cmb1.ds.Aug31.txt            cmb1.sv.Aug31.txt
cmb1.inx.Aug31.txt           cmb1.transdb_io.Aug31.txt
cmb1.nch.Aug31.txt           cmb1.transdb_trans.Aug31.txt
cmb1.ol.Aug31.txt            cmb1.vmm.Aug31.txt
cmb1.permdb_io.Aug31.txt     cmb1.wfl.Aug31.txt
cmb1.permdb_trans.Aug31.txt  rpt1.stat.Aug31
```



## Syntax

getreports [-a] [-v] [-r <reportdir>] [perfargs]

- a** Generates all report types for all servers
- v** Displays output in verbose (detailed) mode
- r <reportdir>** Produces output in the specified directory  
The default file is /fnsw/local/logs/perf.
- perfargs** perf\_report arguments  
Refer to [“perf\\_report” on page 968](#) for a list of arguments and descriptions.

## Example

If run on a server with batch services, getreports calls perf\_report -a which generates the reports for a Batch server including the following "bs" report type. The following example shows a Batch Entry Report. (For samples of other report types, see ["Sample Reports" on page 1000.](#))

```
Batch Entry Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan
```

	CPU		CreateBatc		Open Batch		CommitBatc		CreatImage		WriteImage		Read Image		Create Doc		Update Doc			
--	%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA		
10:00	72	1	3	24	25	0.0	26	0.0	2	0.0	2284	0.0	2291	0.2	14	0.0	2426	0.0	16	0.0
09:45	95	0	1	4	11	0.0	0	*	0	*	249	0.0	248	0.2	0	*	0	*	0	*
.																				
02:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
01:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
SUM:					36		26		2		2533		2539		14		2426		16	
AVG:	98	0	0	2	2	0.0	1	0.0	0	0.0	127	0.0	127	0.2	1	0.0	121	0.0	1	0.0

The report heading includes the report type, date, and time. The prefix for each entry is the time in hh:mm format. In this example, the time shows that perf\_mon had collected the data in 15 minute intervals.

To format the group and column headings for the Batch Services Report (shown in the previous example), perf\_report used the report bs definition in the reports file (located in /fnsw/lib/perf). For more details, see ["Sample Reports" on page 1000.](#)

## Checklist

Before you use getreports, remember to use optional switch settings to limit the report output types, servers, time, and so on.

For example, to limit the output to one day, use the -d1 switch. The output files created apply to one day only. Refer to [“perf\\_report” on page 968](#) for a description of performance report arguments.

## Procedure

No specific procedure is required.

## Related Topics

[“perf\\_mon” on page 954](#)

[“perf\\_report” on page 968](#)

## getstamps

UNIX

The getstamps tool is available only on a UNIX-based Image Services server.

### Description

The getstamps tool is a UNIX shell script that collects stamps for all subsystems on the Image Services server on which you run the program. A stamp displays the system and developer release numbers, subsystem, release type, and SCR number for a set of files.

The getstamps tool enables a technician to generate a database file on the SCR server, which contains the most current software fixes available for a specific UNIX platform and Image Services software release. The technician can then compare the existing SCR database file with the stamps currently installed on a customer's server using either of the following two methods:

- **Intervention:** A technician requests the customer to run the **getstamps -stamps** option at the customer site to generate a stamps report. The customer forwards the stamps report electronically to the technician who runs the **getstamps -compare** option against the customer's stamps report.
- **Customer Site:** A field technician at the customer site remotely logs onto the FTP site and copies the current database file and getstamps tool to the customer's server. The field technician can then run the **getstamps -all** option to generate the stamps report and run the comparison in one step.

By comparing the most current database with the stamps identified on the customer's systems, you can determine whether you need to install any missing fix packs on each system at the customer site.

## Use

Use getstamps before attempting other troubleshooting methods to determine if the customer's Image Services system has the most current fix packs installed on each server.

## Syntax

```
getstamps [ -mkdb | -stamps | -compare | -all ]
```

---

**Note** This command requires a space between the command and its option.

---

**-mkdb** <platform>  
<version\_number> Creates a database of the latest System Change Requests (SCRs) on a FileNet server.

---

**Note** This option is available only to support staff with user access to the System Change Request (SCR) server.

---

**-stamps** Generates the file that contains the stamps for subsystems on the Image Services server. If you run this option on the customer's Image Services server, you should follow it with the **-compare** option.

**-compare** Compares the stamps on the customer's server with the latest fixes identified by available SCRs. You should run this option after generating the stamps report file with the **-stamps** option.

**-all** Performs both the **-stamps** and **-compare** options in one step.

## Sample Output

This section shows the screens displayed when using this tool to verify the current installation of a customer server named “happy.”

### Displaying Help Text

To list the help text showing the proper command syntax, enter the **getstamps.sh** command without any arguments, as shown in the following example.

```
./getstamps.sh

getstamps: Get stamp info; Compare stamps with current SCRs

USAGE: getstamps -mkdb PLATFORM VERSION # Make DB of latest SCRs on cantillo
       getstamps -stamps                # Get site stamps
       getstamps -compare                # Compare site stamps/latest scrs
       getstamps -all                    # One-stop get/compare

EXAMPLE: 1) On cantillo, run "getstamps -mkdb hp9 342"
         2) Copy "scrs.hp9.342.db" to /fnsw/local/tmp on customer system
         3) Finally, run "getstamps -all" on customer's system(s)

NOTES:
* Includes IMS, COLD and WAL SCRs.
* The "-compare" can either be done directly in /fnsw/local/tmp on the
  customer's machine or remotely at FileNet.
```

## Generating the Stamps

To generate the stamps, you must first copy the `getstamps.sh` file to the customer's server, then enter the **getstamps.sh** command with the **-stamps** option, as shown in the following example.

```
./getstamps.sh -stamps
Getting basic system info...
Fetching IMS stamp info, please be patient...
Fetching WAL/Unix stamp info, please be patient...
Done capturing FileNet stamp information:
See list file /fnsw/local/tmp/stamps.happy.lst.
```

## Viewing the Stamps

To view the information contained in the generated stamps file, you could use the **less** command, as shown in the following example.

```
less stamps.happy.lst =>
#
# System Stamp info
#
# System hostname: happy
# System OS: HP-UX happy B.10.20 A 9000/847 1427304221 two-user license
# Stamp Date: Fri, Sep 17, 1999 10:54:15 AM
#
# IMS STAMPS:
-rwsr-s---  1 fnsw      fnusr      192360 Feb  6  1998 /fnsw/bin/ARC_copy
  system 3.4.1.46(3) (userid 8, Thu Jan 29 18:22:13 1998)
  developer 3.4.1.0.0 (userid 8, Thu Jan 29 18:16:57 1998)
  SubSys: ar, Rel_type: rel_hpux, SCR#: 41938, mode: 100755, size: 192360
-rwsr-s---  1 fnsw      fnusr      32972 Feb  6  1998 /fnsw/bin/ARC_create
  system 3.4.1.46(3) (userid 8, Thu Jan 29 18:22:15 1998)
  developer 3.4.1.0.0 (userid 8, Thu Jan 29 18:17:16 1998)
  SubSys: ar, Rel_type: rel_hpux, SCR#: 41938, mode: 100755, size: 32972
...

```



## Running the Compare Command

To compare the system stamps with the current SCR database, change to the directory where you copied the database file, then enter the **getstamps.sh** command with the **-compare** option, as shown in the following example.

```
happy(root)/fnsw/local/tmp> /tommy/paul/proj/getstamps/getstamps.sh -compare =>
Comparing /fnsw/local/tmp/scrs.hp9.341.db with /fnsw/local/tmp/stamps.happy.lst.
```

## Viewing the Compare File

To view the compare file, you could use the **less** command, as shown in the following example.

---

```
less compare.happy.lst =>
#
# SCRs/Stamps Comparison
#
# Report Date: Wed, Sep 29, 1999 07:40:58 AM
#
# SCRs Database: /fnsw/local/tmp/scrs.hp9.341.db
# Mkdir Date: Tue Sep 28 22:24:46 PDT 1999
#
# Stamps list: /fnsw/local/tmp/stamps.happy.lst
# Stamp Date: Wed, Sep 29, 1999 07:40:58 AM
#
# IMS MODULES:
# Latest      Site      Module
# SCR         Stamp    Name
# -----    -
93280        48772    /fnsw/bin/Xcold_rpt
93279        66274    /fnsw/lib/shobj/libINX1.sl
93274        NULL     permission_table: NOT FOUND
```

---

92371	42083	/fnsw/bin/ds_init
90900	61633	/fnsw/lib/shobj/libDTI.sl
90883	43587	/fnsw/bin/fn_util
90874	42942	/fnsw/lib/shobj/libNMI.sl
89901	63166	/fnsw/lib/shobj/libBESl.sl
89899	48780	/fnsw/lib/shobj/libWQSl.sl
89885	65488	/fnsw/bin/fn_build
88924	42083	/fnsw/bin/ds_notify
88911	65436	/fnsw/lib/shobj/libARM.sl
88903	NULL	libBROR.sl_shobj: NOT FOUND
88903	NULL	EBR_orreset_bin: NOT FOUND
88902	42008	/fnsw/lib/shobj/libDOCb.sl
86153	42083	/fnsw/lib/shobj/libDIA.sl
82468	60096	/fnsw/bin/cold_3770
82468	60096	/fnsw/bin/Xcold_ver
82468	60096	/fnsw/bin/Xcold_imp
80643	42083	/fnsw/lib/shobj/libODU.sl
80333	42008	/fnsw/lib/shobj/libDOCr.sl
80333	42008	/fnsw/bin/DOCs
79276	56247	/fnsw/lib/shobj/libODX.sl
79261	42061	/fnsw/bin/nch_tool
79261	52258	/fnsw/lib/shobj/libNCH.sl
78250	42042	/fnsw/bin/dclview
76322	NULL	Xcold_rpt_xvt.h_bin: NOT FOUND
76322	NULL	Xcold_rpt_bin: NOT FOUND
76322	NULL	Xcold_rpt.url_bin: NOT FOUND
76322	NULL	Xcold_rpt.uil_bin: NOT FOUND
76322	NULL	Xcold_rpt.uid_bin: NOT FOUND
75281	60853	/fnsw/lib/shobj/libOSIl.sl
75279	NULL	erm.msg: NOT FOUND
75279	NULL	erm.cat: NOT FOUND
75279	63014	/fnsw/bin/dtp
75229	62411	/fnsw/lib/shobj/libPRIl.sl
74738	42042	/fnsw/bin/dbupgrade
71926	41972	/fnsw/lib/shobj/libBRrt.sl
71926	41972	/fnsw/lib/shobj/libBRbt.sl
71926	41972	/fnsw/bin/backup_demon
71916	61648	/fnsw/lib/shobj/libSysV.sl
71916	61648	/fnsw/bin/killfns
71916	61648	/fnsw/bin/ipc_tool
71146	NULL	erm.msg: NOT FOUND

---

71146	NULL	erm.cat: NOT FOUND
71142	42083	/fnsw/bin/oddump
70404	42028	/fnsw/lib/shobj/libGDBO.sl
70402	42042	/fnsw/bin/INXs
70394	41964	/fnsw/lib/shobj/libDTM.sl
70332	42941	/fnsw/bin/fn_mkf_convert
70263	64671	/fnsw/bin/fn_oracle
69509	41968	/fnsw/lib/shobj/libBRPC.sl
69500	42083	/fnsw/bin/dsched
68631	48772	/fnsw/bin/Xcold_log
67828	51328	/fnsw/bin/WQS_tool
67825	63160	/fnsw/lib/shobj/libCSM1.sl
67822	64684	/fnsw/lib/shobj/libGDB.sl
67819	NULL	/fnsw/lib/shobj/libxvtxmhb458.sl
67819	NULL	/fnsw/lib/shobj/libxvtxmhb458.sl
67802	41997	/fnsw/bin/Xapdprog
66283	66283	/fnsw/lib/shobj/libBESr.sl
66283	66283	/fnsw/lib/shobj/libBES.sl
66283	66283	/fnsw/bin/BESs
66274	66274	/fnsw/bin/INX_tool
65494	65494	/fnsw/bin/ddexim
65459	65459	/fnsw/lib/shobj/libSECr.sl
65459	65459	/fnsw/lib/shobj/libSECl.sl
65459	65459	/fnsw/lib/shobj/libSEC.sl
65459	65459	/fnsw/bin/SECs
65459	65459	/fnsw/bin/SEC_tool
65459	65459	/fnsw/bin/SEC_debug
65269	64689	/fnsw/bin/Xslc
64671	64671	/fnsw/bin/fn_dataset_config
64586	64586	/fnsw/bin/MasterSnmpd
63926	63926	/fnsw/lib/shobj/libSOD.sl
63909	63909	/fnsw/bin/XPR_print
60850	60850	/fnsw/lib/shobj/libDMU.sl
60099	60099	/fnsw/lib/shobj/libPSM.sl
60099	60099	/fnsw/lib/shobj/libPSC.sl
60095	60095	/fnsw/bin/Xbjc
60068	60068	/fnsw/lib/shobj/libPSI.sl
57834	57834	/fnsw/lib/perf/eladisp
57834	NULL	SCT.cor: NOT FOUND
57827	57827	/fnsw/bin/OCOR_Listen
57060	57060	/fnsw/lib/shobj/libHLT.sl

---

```

54516      54516      /fnsw/lib/perf/reports
54516      54516      /fnsw/lib/perf/perf_mon
54516      NULL        elacodes: NOT FOUND
54504      54504      /fnsw/lib/perf/perf_report
52986      52986      /fnsw/lib/shobj/libSC.sl
52262      52262      /fnsw/lib/shobj/libTPI.sl
52262      52262      /fnsw/lib/shobj/libCOR.sl
52261      52261      /fnsw/lib/shobj/libSNM.sl
52261      52261      /fnsw/bin/fn_trapd
52261      52261      /fnsw/bin/fn_snmpd
52258      52258      /fnsw/lib/shobj/libNCH1.sl
52250      52250      /fnsw/bin/fn_edit
52245      52245      /fnsw/bin/fbc_commit
52201      52201      /fnsw/bin/Xsec_admin
51784      51784      /fnsw/bin/SEC_daemon
51719      NULL        Xsysmon.csc: NOT FOUND
51719      NULL        Xsysmon.csa: NOT FOUND
51707      51707      /fnsw/bin/Xsysmon
50849      50849      /fnsw/lib/shobj/libCDB.sl
50849      50849      /fnsw/bin/fn_migrate
50358      50358      /fnsw/bin/SC_convert
49359      49359      /fnsw/lib/shobj/libCSM.sl
49352      49352      /fnsw/lib/shobj/libSSU.sl
49352      49352      /fnsw/lib/shobj/libSSD.sl
49352      49352      /fnsw/lib/shobj/libSMM.sl
48772      48772      /fnsw/bin/Xcold_tem
48772      48772      /fnsw/bin/Xcold_menu
48772      48772      /fnsw/bin/Xcold_job
48772      48772      /fnsw/bin/Xcold_ch
48752      48752      /fnsw/bin/Xpupsec
48730      48730      /fnsw/bin/PRI_tool
48222      48222      /fnsw/lib/perf/log_extract
48222      48222      /fnsw/lib/perf/log_dir
48222      48222      /fnsw/lib/perf/log_create
44711      44711      /fnsw/bin/bes_clean
44562      44562      /fnsw/bin/doccopy
43717      43717      /fnsw/lib/shobj/libDBU.sl

```

## # WAL MODULES:

```

# Latest      Site      Module
# SCR        Stamp    Name

```

---

```
# -----
89876      38033    /fnsw/client/shobj/libINXr.sl
87088      38999    /fnsw/client/bin/wal_purge
86152      38041    /fnsw/client/shobj/libFFI.sl
86152      NULL     FFI.defs: NOT FOUND
73818      34980    /fnsw/client/shobj/libBESr.sl
70287      38999    /fnsw/client/bin/wal_ipc
70287      35118    /fnsw/client/bin/wal_daemon
70287      38999    /fnsw/client/shobj/libSysV.sl
61640      NULL     fn_msg: NOT FOUND
61640      NULL     libERM.sl: NOT FOUND
61634      38001    /fnsw/client/shobj/libDTI.sl
59401      35136    /fnsw/client/bin/wal_cc
59354      38967    /fnsw/client/shobj/libIMS.sl
57830      38984    /fnsw/client/shobj/libSECr.sl
57830      39011    /fnsw/client/shobj/libSEC.sl
57830      NULL     WALSEC_debug: NOT FOUND
57049      38987    /fnsw/client/shobj/libCOR.sl
52996      NULL     libELA.sl: NOT FOUND
52987      35045    /fnsw/client/shobj/libSC.sl
52978      35050    /fnsw/client/shobj/libNCHr.sl
52978      35050    /fnsw/client/shobj/libNCH.sl
52977      35052    /fnsw/client/shobj/libTPI.sl
48120      38944    /fnsw/client/shobj/libSSU.sl
48120      38944    /fnsw/client/shobj/libSMM.sl
46238      35118    /fnsw/client/shobj/libFCL.sl
45221      38017    /fnsw/client/shobj/libPRIr.sl
41919      37649    /fnsw/client/shobj/libFP.sl
39496      35106    /fnsw/client/shobj/libASH.sl
```

---

## Analyzing the Compare File

As shown in the example above, the compare file lists the most recent SCR next to the latest stamp found on each module installed on the customer's Image Services server. The compare file sorts this information in chronological order (by SCR number).

You can easily scan this list to determine if and how far the site stamps for each module are behind the most current SCR. For example:

- Look at the top of the list to see how far out of sync the site stamps are from the latest SCRs.
- Scan down the list to find where the SCR numbers match to determine at what point the SCRs become in sync.

From examining the start of the Image Services stamps report in this example, you can see that the modules at the top of the list are very out of date.

```
# IMS MODULES:
# Latest      Site      Module
# SCR         Stamp    Name
# -----    -
93280        48772    /fnsw/bin/Xcold_rpt
93279        66274    /fnsw/lib/shobj/libINX1.sl
93274        NULL     permission_table: NOT FOUND
67802        41997    /fnsw/bin/Xapdprog
66283        66283    /fnsw/lib/shobj/libBESr.sl
66283        66283    /fnsw/lib/shobj/libBES.sl
66283        66283    /fnsw/bin/BESs
66274        66274    /fnsw/bin/INX_tool
...

```

As you scan further down the list in this example, you can see that the modules become synchronized around SCR 66283. Everything from that point down should match.

```
...
67802      41997      /fns/bin/Xapdprog
66283      66283      /fns/lib/shobj/libBESr.sl
66283      66283      /fns/lib/shobj/libBES.sl
66283      66283      /fns/bin/BESs
66274      66274      /fns/bin/INX_tool
...
```

## Procedure

- 1 Obtain the current SCR database file for the customer's server. Its name identifies the server platform and Image Services release number using the following format:

scrs.<platform>.<release\_number>.db

where:

<platform> is the platform identifier, as defined in the table below  
<release\_number> is the release number, such as 3.5.0, 3.6 or 4.0.0

Platform	Identifier	Sample File Name
IBM AIX	aix	scrs.aix.3.6.db
HPUX 10.20	hp9	scrs.hp9.3.6.db
HPUX 11.x	hp11	scrs.hp11.3.6.db
Windows	nt	scrs.nt.3.6.db
Sun Solaris	sun	scrs.sun.3.6.db

- 2 Copy the getstamps and scrs.<platform>.<release\_number>.db files to the following directory on the customer system:

/fnsw/local/tmp

### Warning

If you use kermit to transfer these files, you must change the name. For example: **mv scrs.hp9x350xdb scrs.hp9.db**

- 3 Run the **getstamps -all** command on the customer's server.



## Checklist

Before you use getstamps, be aware of the following:

- You must generate the stamps report on the customer's system. We recommend that you generate the report on ALL Image Services servers in a dual or multi-server configuration.
- Be sure to create a BINARY copy when transferring files between a Windows Server and a UNIX platform.

## Related Topics

[“stamp” on page 1234](#)

[“stampro” on page 1236](#)

---

# getstatus

## Description

**UNIX**

The getstatus tool is available only on a UNIX-based Image Services server.

The getstatus tool is a script that collects a variety of information on a FileNet system. The information is a snapshot of the state of the system in terms of active processes, network statistics, and database statistics, and a listing of some configuration files.

The getstatus tool retrieves configuration information from various configuration files, for example, /fnsw/local/sd/1/as\_config.s, /fnsw/local/sd/1/as\_config.g, data definition language (ddl) files for MKF databases, and /fnsw/local/sd/1/print\_config for print services. As an option, you can eliminate collecting and displaying the configuration files.

The getstatus tool writes output to the default directory /fnsw/local/logs/perf. The output file name format is:

```
stats.<n>.<mmdd>
```

where

<n> is the server number

<mmdd> is the month and day of the report.

For example, the following file contains statistics gathered on September 1 for server ID 1:

```
stats.1.Sep1
```

If you run `getstatus` and an output file already exists, `getstatus` forces you to overwrite the existing file or exit the program. The following example shows the dialog:

```
elcapitan(fns)/fns/local/logs/perf> getstatus -s
/fns/local/logs/perf/stats.1.Sep1 already exists
Do you wish to overwrite it <(y)/n> ?
```

If you reply **n** to the dialog, `getstatus` terminates. If you want to create a new file without overwriting the existing file, rename the existing file and rerun `getstatus`.

## Use

Use `getstatus` to collect a snapshot of FileNet system performance statistics and configuration settings in a single file.

If the databases are active at the time `getstatus` runs, the report data might not be completely current. The output includes a warning to inform you of this situation. For the most current statistics, run `getstatus` with the databases shut down.

The following example shows a partial report generated by running the `getstatus -eperm` command:

```

System: elcapitan
getstatus
Server Profile Report

Reporting for station 1 on Wed Sep 1 14:37:00 PDT 1999 ...
Invoked as getstatus -eperm

PROCESS STATUS LISTING FOR STATION 1
-----
  F S UID  PID  PPID  C PRI NI ADDR  SZ  RSS  WCHAN  TTY  TIME CMD
303 A  0    0    0 120 16 -- 3c0f 12 12          - 3:11 swapper
200003 A  0    1    0  6 63 20 280a 368 176          - 0:14 /etc/init
303 A  0  516  0 120 127 -- 4010  8  8          - 11832:45 kproc
303 A  0  774  0  0 16 -- 4411 12  8          - 0:08 kproc
303 A  0 1032  0  0 36 -- 5c17 16 16          - 0:06 kproc
303 A  0 1290  0  0 37 -- 6018 64 56          * - 1:40 kproc
40201 A  0 2112  0  0 60 20 4471 16  8          - 0:00 kproc
240001 A  0 2436  1  0 60 20 1144 88 36  EVENT - 3:10 /usr/sbin/syncd 60
40303 A  0 2604  1  0 39 -- 5936 16 16          - 0:00 kproc
40001 A  0 2934  1  0 60 20 715c 408 288          - 0:18 /usr/dt/bin/dtlogin -daemon
240001 A 80 3128 4140  0 60 20 15a5 1480 1000          - 0:09 /usr/dt/bin/dtsession
240001 A 80 3556 24300  0 60 20 2769 540 12          - 0:00 /bin/bsh /bin/bsh /bin/bsh
40401 A  0 3644  1  0 60 20 5e1 472 12  EVENT - 0:00 /usr/lib/errdemon
240001 A  0 3878 2934  0 60 20 2168 5304 2524  EVENT - 5:58 /usr/lpp/X11/bin/X -x abx -x dbe
-x GLX -D /usr/lib/X11/rgb
40001 A  0 4140 2934  0 60 20 697a 592 68          - 0:00 dtlogin <:0> -daemon
240001 A 80 4730 24024  0 60 20 381 304 40          - 0:00 /usr/dt/bin/dtexec -open 0 -ttprocid
2.tmj4o 01 24300 134217
240001 A  0 4910 5164  0 60 20 25a9 136 12  EVENT - 0:00 /usr/lpp/info/bin/infod
240001 A  0 5164  1  0 60 20 15e5 300 12          - 0:00 /usr/sbin/srcmstr
240001 A  0 5426  1  0 60 20 7ddf 248 212  EVENT - 0:01 /usr/sbin/cron
240001 A  0 5676  1  0 60 20 29ea 72 20  EVENT - 0:00 /usr/ccs/bin/shlap
240401 A  0 6022  1  0 60 20 328c 40 12  EVENT - 0:00 /usr/sbin/uprintfd
240001 A  0 6256 5164  0 60 20 4a52 224 88          - 0:00 /usr/sbin/syslogd
240001 A  0 6452 5164  0 60 20 6659 724 440          - 0:00 sendmail: accepting connections on
port 25
...

```

## Syntax

```
getstatus [-v] [-d] [-s] [-y] [-a] [-r reportdir] [-E]
[-e<mkf | perf | perm | net>] [-nconfig] [-p<password>]
```

- v** Generates report in verbose mode
- d** Displays commands that will be run but does not generate reports
- s** Displays a snapshot without service reports
- y** Forces overwrite of an existing output file
- r <reportdir>** Writes output to the directory specified by <reportdir>, which is the full path name of the directory to receive the report file. Default directory is /fnsw/local/logs/perf.
- E** Runs all extra statistical output options (-enet for network, -emkf for MKF, -eperf for performance configuration, and -eperm for permanent database)
- enet** Displays extra network output only  
Runs netstats, a standard UNIX tool. For netstats options and format, see the netstats manual pages for your platform.
- emkf** Displays extra MKF output only (runs MKF\_perf)
- eperf** Displays extra output on the performance configuration only and concatenates the perf\_mon.script file to the getstatus output
- eperm** Displays extra MKF statistics on the permanent database only  
Runs MKF\_stats on the MKF permanent database.
- nconfig** Do not display a listing of configuration settings

**-p<password>** getstatus has a hard-coded rdbms password for f\_maint. If the hard-coded one is not the real password, you can now put in the password.

---

**Note** To display online help for command syntax, enter **getstatus - -**

---

---

## Example

You can specify the `-d` option to display the commands that `getstatus` will run but without generating reports, as shown in the example below:

---

```
elcapitan(fnsw)/usr/fnsw> getstatus -d
                        System: elcapitan
                        getstatus
                        Server Profile Report

Reporting for station 1 on Thu Sep 2 10:06:13 PDT 1999 ...
Invoked as getstatus -d

DISPLAY ONLY MODE NO DATA WILL BE GATHERED

CMD:      ps gl, ps guw, ps gv
CMD:      PPMOI
CMD:      netstat -m, netstat -in
CMD:      netstat -v, netstat -m, netstat -in
CMD:      iostat
CMD:      vmstat 1 5, vmstat -s
CMD:      svmon
CMD:      df
CMD:      nfsstat
CMD:      lsvg -p <lv>, lspv -l <pv>
CMD:      lscfg
CMD:      CSM_tool
CMD:      dbp -s
CMD:      MKF_stats -d /fnsw/dev/1/transient_db0
CMD:      Checking transient.ddl files
CMD:      Checking as_config.s files
CMD:      Checking as_config.g files
CMD:      Checking init.ora
CMD:      Checking DocClass Indexes
CMD:      WQS_tool
End getstatus report for elcapitan station 1
```

---

The following example spanning the next few pages is a partial report provided only to show the major subheadings of information collected by getstatus:

```
System: elcapitan
getstatus
Server Profile Report
```

```
Reporting for station 1 on Thu Sep 2 10:09:40 PDT 1999 ...
Invoked as getstatus
```

```
PROCESS STATUS LISTING FOR STATION 1
```

```
-----
  F S UID  PID  PPID  C PRI NI ADDR  SZ  RSS  WCHAN  TTY  TIME CMD
303 A  0    0    0 120 16 -- 3c0f 12  12          -   3:35 swapper
200003 A  0    1    0  0 60 20 280a 368 180          -   0:19 /etc/init
303 A  0    516  0 120 127 -- 4010  8  8          - 12997:47 kproc
303 A  0    774  0  0 16 -- 4411 12  8          -   0:08 kproc
...
USER      PID %CPU %MEM  SZ  RSS  TTY STAT  STIME  TIME COMMAND
root      516 99.2  0.0   8   8   - A      Aug 24 12997:47 kproc
fnsw     52778  0.1  1.0 1268 1476  - A      14:18:57  0:39 BESs
root     3878  0.0  1.0 5304 2444  - A      Aug 24  6:20 /usr/lpp/X11/bin/X -x
abx -x dbe -x GLX -D /usr/lib/X11//rgb -T -forc
fnsw     72598  0.0  1.0 1260 1468  - A      14:22:03  0:27 BESs
...
  PID  TTY STAT  TIME PGIN  SIZE  RSS  LIM  TSIZ  TRS %CPU %MEM COMMAND
    0   - A   3:35   8    12    12   xx   0    0    0 0.0 0.0 swapper
    1   - A   0:19   75   344   180 32768  25   36    0 0.0 0.0 /etc/init
  516   - A 12997:47  0    8     8   xx   0    0 99.2 0.0 kproc
  774   - A   0:08   0    12    8    xx   0    0 0.0 0.0 kproc
 1032   - A   0:07   10   16   16   xx   0    0 0.0 0.0 kproc
 1290   - A   1:47   0    64   56   xx   0    0 0.0 0.0 kproc
 2112   - A   0:00   0    16    8    xx   0    0 0.0 0.0 kproc
 2436   - A   3:28 2231  88   36   xx   2    4 0.0 0.0 /usr/sbin/syncd
60
...
PPMOI ON STATION 1
-----
```



QPerf - Display RPC queue performance statistics.[qperf|qp]  
 Quit - Terminates the Operator Interface session.[quit|q]

#	Name	Number	Vers	MaxProc	No Proc	No Culs
0	NCHs	2	2	12	0	0
	Pid	Status	Conns	Avg Time	Min Time	Max Time
70454	available/idle	1368	0.0035	0.0016	0.0295	
80890	available/idle	5	0.0138	0.0035	0.0475	
	Totals	1373	0.0035	0.0016	0.0475	
1	CSMs	134231040	1	12	0	0
	Pid	Status	Conns	Avg Time	Min Time	Max Time
75710	available/idle	2234	0.0862	0.0013	60.3077	
82516	available/idle	938	0.1001	0.0015	3.3917	
37056	available/idle	558	0.1279	0.0027	1.7081	
30200	available/idle	254	0.1215	0.0022	0.8726	
62628	available/idle	20	0.2304	0.0123	0.8204	
76930	available/idle	10	0.4790	0.0210	1.8399	
74422	available/idle	4	0.7674	0.1711	1.7254	
66476	available/idle	1	0.5868	0.5868	0.5868	
65850	available/idle	1	0.5536	0.5536	0.5536	
	Totals	4020	0.1001	0.0013	60.3077	
2	DOCs	134231041	1	12	0	0

...

NETWORK STATISTICS FOR STATION 1

Kernel malloc statistics:

\*\*\*\*\* CPU 0 \*\*\*\*\*

By size	inuse	calls	failed	free	hiwat	freed
32	306	268664	0	462	640	8
64	168	1835	0	24	320	0
128	118	127809	0	106	160	52
256	532	3959879	0	92	384	0
512	265	247482	0	7	40	161
1024	118	234478	0	46	100	5563
2048	384	18642	0	4	100	0
4096	2	21569	0	110	120	146

8192	6	786	0	3	10	0
16384	1	1145	0	20	24	7
32768	1	1	0	0	1023	0

By type            inuse            calls failed    memuse    memmax    mapb

Streams mblk statistic failures:

0 high priority mblk failures

0 medium priority mblk failures

0 low priority mblk failures

Name	Mtu	Network	Address	Ipkts	Ierrs	Opkts	Oerrs	Coll
lo0	16896	link#1		300593	0	301147	0	0
lo0	16896	127	127.0.0.1	300593	0	301147	0	0
lo0	16896	:::1		300593	0	301147	0	0
en0	1500	link#2	0.60.94.e9.2a.ea	7315921	0	1431230	0	0
en0	1500	10.2	10.2.52.110	7315921	0	1431230	0	0

NETWORK STATISTICS FOR STATION 1

ETHERNET STATISTICS (ent0) :

Device Type: IBM 10/100 Mbps Ethernet PCI Adapter (23100020)

Hardware Address: 00:60:94:e9:2a:ea

Elapsed Time: 9 days 2 hours 16 minutes 55 seconds

Transmit Statistics:

Packets: 1431229

Bytes: 227118601

Interrupts: 24602

Transmit Errors: 0

Packets Dropped: 0

Max Packets on S/W Transmit Queue: 42

S/W Transmit Queue Overflow: 0

Current S/W+H/W Transmit Queue Length: 1

Broadcast Packets: 558

Multicast Packets: 2

No Carrier Sense: 0

Receive Statistics:

Packets: 10881457

Bytes: 2098484155

Interrupts: 10880858

Receive Errors: 0

Packets Dropped: 0

Bad Packets: 0

Broadcast Packets: 9399940

Multicast Packets: 6259

CRC Errors: 0

```

DMA Underrun: 0
Lost CTS Errors: 0
Max Collision Errors: 0
Late Collision Errors: 0
Deferred: 48392
SQE Test: 0
Timeout Errors: 0
Single Collision Count: 25858
Multiple Collision Count: 8624
Current HW Transmit Queue Length: 1

DMA Overrun: 0
Alignment Errors: 0
No Resource Errors: 0
Receive Collision Errors: 0
Packet Too Short Errors: 0
Packet Too Long Errors: 0
Packets Discarded by Adapter: 0
Receiver Start Count: 0

```

## General Statistics:

```

-----
No mbuf Errors: 0
Adapter Reset Count: 0
Driver Flags: Up Broadcast Running
Simplex AlternateAddress 64BitSupport

```

## IBM 10/100 Mbps Ethernet PCI Adapter (23100020) Specific Statistics:

```

-----
Chip Version: 25
RJ45 Port Link Status : up
Media Speed Selected: 10 Mbps Half Duplex
Media Speed Running: 10 Mbps Half Duplex
Receive Pool Buffer Size: 384
Free Receive Buffers: 384
No Receive Buffers: 0
Inter Packet Gap: 96
Adapter Restarts due to IOCTL commands: 1
Packets with Transmit collisions:
  1 collisions: 25858      6 collisions: 0          11 collisions: 0
  2 collisions: 7376     7 collisions: 0          12 collisions: 0
  3 collisions: 1183     8 collisions: 0          13 collisions: 0
  4 collisions: 64       9 collisions: 0          14 collisions: 0
  5 collisions: 1       10 collisions: 0         15 collisions: 0
Excessive Deferrals: 0

```

## Kernel malloc statistics:

```

***** CPU 0 *****
By size      inuse      calls failed    free    hiwat    freed

```

32	306	268667	0	462	640	8
64	168	1835	0	24	320	0
128	118	127809	0	106	160	52
256	532	3959885	0	92	384	0
512	265	247491	0	7	40	161
1024	118	234478	0	46	100	5563
2048	384	18642	0	4	100	0
4096	2	21569	0	110	120	146
8192	6	786	0	3	10	0
16384	1	1145	0	20	24	7
32768	1	1	0	0	1023	0

By type            inuse            calls failed    memuse    memmax    mapb

Streams mblk statistic failures:

0 high priority mblk failures

0 medium priority mblk failures

0 low priority mblk failures

Name	Mtu	Network	Address	Ipkts	Ierrs	Opkts	Oerrs	Coll
lo0	16896	link#1		300593	0	301147	0	0
lo0	16896	127	127.0.0.1	300593	0	301147	0	0
lo0	16896	:::1		300593	0	301147	0	0
en0	1500	link#2	0.60.94.e9.2a.ea	7315921	0	1431230	0	0
en0	1500	10.2	10.2.52.110	7315921	0	1431230	0	0

IO STATISTICS FOR STATION 1

```
-----
tty:      tin      tout  avg-cpu:  % user  % sys  % idle  % iowait
          0.0      2.9           0.4    0.4    97.9    1.3
```

Disks:	% tm_act	Kbps	tps	Kb_read	Kb_wrtn
hdisk0	0.5	3.3	0.5	300209	2288707
hdisk1	1.0	7.2	1.3	2534708	3121910
cd0	0.0	0.8	0.0	601724	0

VM STATISTICS FOR STATION 1

```
-----
kthr      memory          page          faults          cpu
-----
```

---

```

r  b  avm  fre  re  pi  po  fr  sr  cy  in  sy  cs  us  sy  id  wa
0  1  66912  537  0  0  0  0  1  0  127  426  49  0  0  98  1
0  1  66912  537  0  0  0  0  0  0  116  393  31  0  0  99  0
0  1  66912  537  0  0  0  0  0  0  113  49  24  0  0  99  0
0  1  66912  537  0  0  0  0  0  0  119  96  29  0  0  99  0
0  1  66912  537  0  0  0  0  0  0  113  66  34  0  0  99  0

```

```

15275657 total address trans. faults
  219075 page ins
  575901 page outs
    6067 paging space page ins
    28586 paging space page outs
      0 total reclaims
4167395 zero filled pages faults
  23777 executable filled pages faults
1289982 pages examined by clock
   20 revolutions of the clock hand
304366 pages freed by the clock
  72405 backtracks
    0 lock misses
    733 free frame waits
    0 extend XPT waits
   64077 pending I/O waits
  636665 start I/Os
  636665 iodones
39094259 cpu context switches
99906885 device interrupts
    0 software interrupts
    0 traps
335071513 syscalls

```

## SVM STATISTICS FOR STATION 1

```

-----
/usr/bin/svmon: Permission denied.

```

## DISK FREE SPACE ON STATION 1

```

-----
Filesystem      512-blocks      Free %Used      Iused %Iused Mounted on
/dev/hd4         81920          61936  25%         1452    8% /
/dev/hd2        2031616        726672  65%        19477   8% /usr
/dev/hd9var     16384          13304  19%          209   11% /var

```

```

/dev/hd3          49152      25888    48%      130      3% /tmp
/dev/hd1          16384      15776     4%        22      2% /home
/dev/lv00         98304      14184    86%       684     6% /usr/welcome_arcade
/dev/lv01         49152      5400     90%       380     7% /usr/welcome
/dev/lv02         458752    195832   58%      1326    3% /fnsw
/dev/lv03         409600    157640   62%      1743    4% /fnsw/local
/dev/fn_oracle_ct0 16384      14248   14%        18     1% /fnsw/local/oracle/control0
/dev/fn_oracle_ct1 16384      14248   14%        18     1% /fnsw/local/oracle/control1
/dev/lv06         704512    268464   62%      2805    4% /usr/ora/733
/dev/lv07         933888    224392   76%      1753    2% /usr/ora/804
diskfarm01:/dvt_wkspac/dvt/3.6/rel_aix/dvt/bin 7987200 1568528 81% 7489
4% /fnsw/dvt/bin
diskfarm01:/dvt_wkspac/dvt/3.6/rel_aix/dvt/shobj 7987200 1568528 81%
7489 4% /fnsw/dvt/shobj

```

## NFS STATISTICS FOR STATION 1

-----

## Server rpc:

Connection oriented:

```

calls      badcalls    nullrecv    badlen      xdrcall     dupchecks   dupreqs
0           0            0           0           0           0           0

```

Connectionless:

```

calls      badcalls    nullrecv    badlen      xdrcall     dupchecks   dupreqs
0           0            0           0           0           0           0

```

## Server nfs:

```

calls      badcalls    public_v2   public_v3
0           0            0           0

```

Version 2: (0 calls)

```

null      getattr    setattr     root        lookup      readlink    read
0 0%      0 0%       0 0%        0 0%        0 0%        0 0%      0 0%
wrcache  write      create      remove      rename      link        symlink
0 0%      0 0%       0 0%        0 0%        0 0%        0 0%      0 0%
mkdir    rmdir     readdir     statfs
0 0%      0 0%       0 0%        0 0%

```

Version 3: (0 calls)

```

null      getattr    setattr     lookup      access      readlink    read
0 0%      0 0%       0 0%        0 0%        0 0%        0 0%      0 0%

```

```

write      create      mkdir      symlink    mknod      remove     rmdir
0 0%      0 0%      0 0%      0 0%      0 0%      0 0%      0 0%
...

```

---

```

LSVG/LSPV CONFIGURATION FOR STATION 1
-----

```

```

rootvg:

```

PV_NAME	PV STATE	TOTAL PPs	FREE PPs	FREE DISTRIBUTION
hdisk0	active	537	273	81..00..00..84..108

```

fnvg:

```

PV_NAME	PV STATE	TOTAL PPs	FREE PPs	FREE DISTRIBUTION
hdisk1	active	537	325	03..00..107..107..108

```

hdisk0:

```

LV NAME	LPs	PPs	DISTRIBUTION	MOUNT POINT
hd5	1	1	01..00..00..00..00	N/A
fn_oracle_ct1	1	1	01..00..00..00..00	/fnsw/local/oracle/control1
lv03	25	25	25..00..00..00..00	/fnsw/local
hd6	64	64	00..64..00..00..00	N/A
lv00	6	6	00..06..00..00..00	/usr/welcome_arcade
lv01	3	3	00..03..00..00..00	/usr/welcome
lv02	28	28	00..28..00..00..00	/fnsw
fn_oracle_ct0	1	1	00..01..00..00..00	/fnsw/local/oracle/control0
hd1	1	1	00..01..00..00..00	/home
hd3	3	3	00..03..00..00..00	/tmp
hd9var	1	1	00..01..00..00..00	/var
hd8	1	1	00..00..01..00..00	N/A
hd4	5	5	00..00..05..00..00	/
hd2	124	124	00..00..101..23..00	/usr

```

...

```

```

LSCFG CONFIGURATION FOR STATION 1
-----

```

```

INSTALLED RESOURCE LIST

```

The following resources are installed on the machine.

+/- = Added or deleted from Resource List.

\* = Diagnostic support not available.

```

Model Architecture: chrp

```

---

Model Implementation: Multiple Processor, PCI bus

```

+ sysplanar0      00-00      System Planar
+ mem0            00-00      Memory
+ proc0           00-00      Processor
+ L2cache0       00-00      L2 Cache
* pmc0           00-00      n/a
* pci1           00-fee00000  PCI Bus
* pci0           00-fef00000  PCI Bus
* isa0           10-58      ISA Bus
+ fda0           01-D1      Standard I/O Diskette Adapter
+ fd0            01-D1-00-00  Diskette Drive
* siokma0        01-K1      Keyboard/Mouse Adapter
+ sioka0         01-K1-00    Keyboard Adapter
+ kbd0           01-K1-00-00  PS/2 keyboard
+ sioma0         01-K1-01    Mouse Adapter
+ mouse0         01-K1-01-00  3 button mouse
+ siota0         01-Q1      Tablet Adapter
+ paud0          01-Q2      Ultimedia Integrated Audio
...

```

#### CACHE STATISTICS

-----  
Type '?' for help

Cache Id	Name	% locked	% full	% free
1	page_cache1:elcapitan:FileNet	0	10	90
3	bes_cache1:elcapitan:FileNet	0	0	99
4	sys_print_cache1:elcapitan:FileNet	0	0	100
5	app_print_cache1:elcapitan:FileNet	0	0	100
*	Physical space summary	0	2	98

Statistics for cache #1, name = 'page\_cache1:elcapitan:FileNet'

min_cache_sectors	20480	locked_sectors	0	inuse_sectors	1868
max_cache_sectors	20480	locked_objects	0	inuse_objects	948
free_sectors	18612	ageable	T	refcnts	F
self_cleaning	F				

...

-----  
Physical space summary

reserved_sectors	51200	locked_sectors	0	inuse_sectors	1868
------------------	-------	----------------	---	---------------	------



```
max_cache_sectors 102399  locked_objects  0          inuse_objects  948
free_sectors      100531  largest_fs_blk 99777
```

```
-----
Prefetch duration.....300 secs
Migrate duration.....300 secs
Refresh duration.....120 secs
Cache daemon threshold.....80%
Locked object threshold.....85%
Writes critical threshold...90%
```

## DOCUMENT BUFFER STATISTICS

-----  
Directory buffers:

```
  Avg_wait_msec:      0,  Max_wait_msec:      0
  Num_avl_bufs:       4,  Min_avl_bufs:       2
```

## Page buffers:

```
  Avg_wait_msec:      0,  Max_wait_msec:      1
  Num_avl_bufs:      16,  Min_avl_bufs:       7
```

## Descriptor buffers:

```
  Avg_wait_msec:      0,  Max_wait_msec:      9,  Cache hits:      0
  Num_avl_bufs:      16,  Min_avl_bufs:     12,  Cache allocs:   459
```

## TRANSDB MKF STATISTICS ON STATION 1

```
-----
MKF statistics for /fnsw/dev/1/transient_db0:
```

```
WARNING: /fnsw/dev/1/transient_db0 is NOT shutdown
Statistics may not be up to date completely.
```

```
File 0: /fnsw/dev/1/transient_db0 0 20480
```

Block type	Count	% nonvir.	% of Total
Virgin	17616	0.00	86.02
Empty	2546	88.87	12.43
Data	49	1.71	0.24
Index	223	7.78	1.09
Description	44	1.54	0.21
Empty list head	1	0.03	0.00
Large rec. head	1	0.03	0.00
Restart import	0	0.00	0.00

```
=====
```

---

```
Nonempty          319          11.13          1.56

Max number of blocks in data base = 20480
Nonvirgin blocks in data base      = 2865    13.99 %
```

## TRANSIENT.DDL PARAMETERS

-----

--transient Document Server Database DDL

## PARAMETERS

...

## AS\_CONF.G PARAMETERS

-----

...

## ORACLE CONFIGURATION

-----

```
control_files = /fnsw/local/oracle/control0/ctl.ora,/fnsw/local/oracle/control1/
ctl.ora
db_name = indexdb
nls_territory = AMERICA
db_block_buffers = 1000
db_block_size = 2048
db_files = 20
db_file_multiblock_read_count = 8
dml_locks = 100
log_archive_start = FALSE
...
```

## WORKFLO QUEUE PERFORMANCE ANALYSIS

-----

Type '?' for help

&lt;WQS\_tool&gt;Total Open: 0 Critical Lock: NOT LOCKED (0)

=====

&lt;WQS\_tool&gt;&lt;WQS\_tool&gt;

End getstatus report for elcapitan station 1

## Checklist

Before you use getstatus, be aware of the following:

- When run against a large database, the -eperm option could take considerable time to complete. For example, getstatus -eperm takes approximately ten minutes to complete for a 600 MB permanent database.
- If the databases are active when getstatus runs, report data might not be completely accurate. For the most accurate statistics, shut down the databases before you start getstatus.
- The getstatus output varies from platform to platform.
- To obtain RDBMS status, see [\*\*“getstamps” on page 532.\*\*](#)

## Procedure

No specific procedure is required.

## Related Topics

[“CSM\\_tool” on page 204](#)

[“dbp” on page 252](#)

[“getstamps” on page 532](#)

[“MKF\\_stats” on page 795](#)

[“perf\\_mon” on page 954](#)

[“PPMOI” on page 1030](#)

[“spacerpt” on page 1203](#)

[“WQS\\_tool” on page 1325](#)

# glogin

## Description

The glogin tool connects your terminal on the current local system to the remote system you specify and logs you in to that remote station with the same user name you are currently using on your local system. You can specify a different login user name by using the -l option when you run glogin.

The remote system name must be one that exists in /etc/uucpname, or a decimal station number, or both the name and number.

The server for the glogin program, gloginsvr, provides a remote login facility for FileNet users. The gloginsvr is started during FileNet software initialization and must be running before you can use glogin.

AIX

The glogin tool is supported on Image Services for AIX/6000 systems only.

## Use

Use glogin to log in to a remote system from your current local system. For example, you can examine files on a remote system or check job status on a remote system while working on your local system.

## Syntax

glogin <remote host name> [ -l <user name > ]

**<remote host name>**

System name of the remote host to be logged into, or a decimal station number, or both the name and number, separated by a colon. The default station number is 1 and the default system name is that of the local system.

**-l <user name>**

The alternate user name that should be logged into the remote system.

## Example

In the example below, a user on the norco system issues the glogin command requesting connection to a remote host named corona. The user enters the glogin command at norco's system prompt (norco%). Upon successful login to the remote host, the remote system (corona) displays its system prompt [corona(root/)>]:

```
norco% glogin corona  
corona(root/)>
```

## Checklist

Before using glogin, be aware of the following:

- Do not use the ~. (tilde + period) key combination to log out of a remote session. To terminate a glogin session, use the logout command or the Control+d key combination.

If you use the ~. key combination, glogin displays the following message:

```
glogin: Do you want abnormal termination? [ny]
```

Respond by pressing any key other than y or n to reestablish glogin operation. If you invoked the dialout command from glogin, you can pass the ~. (tilde, period) through glogin by typing ~~. (tilde, tilde, period) to terminate dialout.

- The glogin tool is supported on Image Services for AIX/6000 systems only.

## Procedure

No specific procedure is required.

## Related Topics

[“dialout” on page 295](#)

[“gaddr” on page 493](#)

# gls

## Description

The gls tool is a subset of the **ls** tool and lists directories and, optionally, files on remote systems.

### AIX

The gls tool is supported on Image Services for AIX/6000 systems only.

## Use

Use gls to list the directories of remote systems. The gls tool accepts only one name argument of the form **system:path**. If the path is invalid on the specified system or is not specified for a remote file or directory, gls interprets the path relative to a directory on the remote host as follows: The string value of the REMOTEDIR environment variable is used if it has been set for the gls user; otherwise, /tmp is used as the default path.

## Syntax

```
gls [ -l | -C ] [ -a ] [ <system name>:[<path>] ]
```

- l Produces a directory list in single column format. If the output is to a pipe or a file instead of a terminal screen or window, the -l option is automatically set.

---

**Note** -l is mutually exclusive with -C.

---



- C Produces a directory list in multi-column format when the output is to a pipe or a file instead of a terminal screen or window.

**Note** -C is mutually exclusive with -l.

- a Lists all files, including those with names beginning with a period (.)

**<system name>** Remote system name and full path name of the requested directory.  
**:<path>** Default path is /tmp if you do not specify the path name and the REMOTEDIR variable is not set for the gls user.

## Sample Output

```
corona(root)> gls costa3:/bin/a*
acctcom  aclput  adfutil  anno     ar       asa      atq      axeb
acledit  adb     admin    apply    arcv    at       atrm
aclget   addbib  ali      apropos  as      ate      awk
```

## Checklist

Before using gls, be aware of the following:

- The requested path must be valid on the specified system.
- The default path is /tmp if you do not specify the path name and the REMOTEDIR variable is not set for the gls user.
- The gls tool is supported on Image Services for AIX/6000 systems only.

## Procedure

No specific procedure is required.

---

# gsh

## Description

The gsh (global shell) tool runs commands on a remote system. The default user name is that which you used to log on to the local system. You can change the name using a gsh command option.

AIX

The gsh tool is supported on Image Services for AIX/6000 systems only.

## Use

From your local system, use the gsh tool to run commands and interactive programs on a remote system.

Use the gaddr command to determine what remote systems are available to you.

## Syntax

```
gsh [ -l <user name> ] <remote host> [ <command> ]
```

**-l <user name>** Changes the local user name to that specified in <user name>

**<remote host>** Name of remote host on which to run the command

The remote host must be either a name in /etc/uucpname, a decimal station number, or both the name and number in the format **rhost:station#**.

**<command>** The command you want to run

## Example

### **gsh amber ls -l**

In this example, the list command (ls -l) runs on the remote host called amber and the output produced by the list command returns to the local terminal display.

## Checklist

Before using gsh, be aware that gloginsvr provides a remote login facility for FileNet users. The gloginsvr is started during FileNet software initialization and must be running on the local system before you can use gsh.

## Procedure

No specific procedure is required.

## Related Topics

**[“gaddr” on page 493](#)**

---

# ident

## Description

The `ident` tool identifies a file by searching the specified source (text) files or object code files for all occurrences of a keyword pattern. The `ident` tool displays the output at the terminal or window from which you issued the command.

The keyword pattern can identify author, date, header, or other information. Keyword patterns are normally inserted into a source (text) file or object code file automatically by commands, such as the Revision Control System (RCS) checkout command, or programs. For example, if a C program in `file.c` (the source file) inserts header information as the following string array:

```
char rcsid[ ] = "$Header: file.c,o last updated 96/10/19 $"
```

and `file.c` is compiled into `file.o` (the object file), you can use `ident` to display the header information from both `file.c` and `file.o`, as shown below:

```
ident file.c file.o
file.c:
$Header: file.c,o last updated 96/10/19 $
file.o:
$Header: file.c,o last updated 96/10/19 $
```

You can also insert keyword patterns manually into a source file using a text editor. You can then use `ident` to search for these patterns within a file.

**Note** If the keyword pattern is in a comment line, ident only finds the pattern in the source file, not in the object code file.

---

## Use

Use the ident tool to locate and list a file by a specified date, author, header, locker, log, revision level, source, or state. For example, you can find header information for a file or group of files indicating the date the file was last updated or generated.

You can also use the stamp tool to obtain information about a file.

## Syntax

```
ident <file1> ...<fileN> [ $<pattern keyword>: ...$]
```

**<file>** One or more file names to be searched

**<pattern keyword>** The text string for which ident searches, where <pattern keyword> is one of the following: Author, Date, Header, Locker, Log, Revision, Source, State

## Sample Output

In the example below, `ident` first extracts header information from the source code files `main.c` and `main2.c`. Then, using a wildcard file search character (`*`), `ident` extracts header information from all source and object files starting with the letter `m`.

```
% ident main.c
main.c:
    $Header: main.c,v 1.1 96/10/19 11:44:07 tom Exp $
    $Header: main.c,v 1.1 96/10/19 11:44:07 tom Exp $
% ident main2.c
main2.c:
    $Header: This line was manually changed in vi by Tom $
    $Header:$
% ident m*
main:
    $Header: main.c,v 1.1 96/10/19 11:44:07 tom Exp $

main.c:
    $Header: main.c,v 1.1 96/10/19 11:44:07 tom Exp $
    $Header: main.c,v 1.1 96/10/19 11:44:07 tom Exp $

main2:
    $Header:$

main2.c:
    $Header: This line was manually changed in main2.c using an editor $
    $Header:$
```

## Checklist

Before you use ident, be aware of the following:

- You can use ident to extract information from both source (text) files and object code files.
- You can also use the stamp tool to obtain file identification information.

## Procedure

No specific procedure is required.

## Related Topics

[“stamp” on page 1234](#)

# import\_cdb

## Description

This tool imports the latest Configuration Database (.cdb) file into the IS configuration directory. This tool is designed to be the last step in the process of cloning the IS system configuration of multiple IS systems from a single master configuration database file and is done as a part of a fresh install of Image Services. The other tool associated with this process is the [export\\_cdb](#) tool.

For more information on the cloning process, see [“export\\_cdb” on page 409](#).

---

**Note** All the path names documented for this tool are shown in UNIX format (e.g., /fnsw/local/sd/conf\_db/import). For Windows Server path names, substitute the /fnsw/local/ with <drive>:\fnsw\_loc\ and use a back slash (\) instead of a forward slash (/) throughout the rest of the path (e.g., <drive>:\fnsw\_loc\sd\conf\_db\import).

---

## Use

You use this tool to import the .cdb file from /fnsw/local/sd/conf\_db/import directory to create a new .cdb file in the /fnsw/local/sd/conf\_db directory. The tool always selects the latest version of the .cdb file to import.

As part of the import, the utility automatically updates the domain and server information, the IP address, and the SSN number in the .cdb file to reflect the new system.



The import\_cdb tool should be used in conjunction with the export\_cdb tool.

## Syntax

```
import_cdb [-s <source_prefix>]
```

### Options

**-s <source\_prefix>** Optionally allows the user to specify a prefix for the source file other than the default “ims” prefix.

---

**Note** When you import using the optional -s parameter, you are still creating a new revision for the default IS configuration database file (ims\_xx.cdb). The xx in the file name will be one revision higher.

---

## Procedure

- 1 On the IS server being used as the master server, use fn\_edit to define the configuration database parameters. This has probably been done and the latest cdb file is established.
- 2 On the IS server being used as the master server, run **export\_cdb** to make a copy of the master IS server’s configuration database file by entering the following command:

### **export\_cdb**

In the above example, the export file would be created in the following directory structure: /fnsw/local/sd/conf\_db/export/ims\_exp\_xx.cdb, where xx is the revision number of the .cdb file.

If you want to use the optional **-s <source\_prefix>** parameter, <source\_prefix> is the value you use to associate the file with the clone server the configuration file is being exported to (for example, california). In this -s example, the export file would be created in the following directory structure: /fnsw/local/sd/conf\_db/export/california\_exp\_xx.cdb, where xx is the revision number of the .cdb file.

For more information on the **export\_cdb** tool, see [“export\\_cdb” on page 409](#)

- 3 Manually move or copy the newly created file from the master IS server to the clone IS server by placing it in the following location on the clone IS server:

```
/fnsw/local/sd/conf_db/import/ims_exp_xx.cdb
```

- 4 On the IS server being used as the clone server, run the following command to import the configuration database file and create a .cdb file:

### **import\_cdb**

Again if you want to use the optional **-s <source\_prefix>** parameter, <source\_prefix> is the value you use to associate the file with the clone server the configuration file is being exported to (for example, california).

## **Related Topics**

[“export\\_cdb” on page 409](#)

[“migrate\\_cdb\\_changes” on page 709](#)

# initbackup

## Description

The initbackup command sets up the environment for backups with a call to initfnsw backup. This includes running logproc to log errors, OCOR\_listen to monitor for network requests and replies, MKF\_startup to start the network clearinghouse database (NCH\_db0), and NCH\_daemon (on any server providing a remote NCH service).

## Use

In IDMIS 3.3.0 and later, the preferred method of initializing Image Services software is with the FileNet Task Manager. However, you can still run initbackup from the command line if you need to.

Use initbackup to prepare a server's processes for backup. Run initbackup on each server to prepare the environment prior to running a script to perform an offline backup.

## Syntax

```
initbackup
```

---

## Sample Output

The following sample is output from execution of the initbackup script.

---

```
Terminating FileNet software on Root/Index/Osar server...
Shutdown oracle data base.

SQL*DBA: Release 7.0.12.1.0 - Production on Thu Jan 13 09:38:41 1994

Copyright (c) Oracle Corporation 1979, 1992. All rights reserved.

ORACLE7 Server Release 7.0.12.1.0 - Production
With the procedural option
PL/SQL Release 2.0.14.0.1 - Production

Connected.
ORACLE instance shut down.
ORACLE instance started.
Database mounted.
Database opened.
Total System Global Area          4398928 bytes
      Fixed Size                   34344 bytes
      Variable Size                3922216 bytes
Database Buffers                   409600 bytes
Redo Buffers                        32768 bytes
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL*DBA complete.
Shutting down permanent data base.
Database is shut down.
Shutting down transient data base.
Database is shut down.
Shutting down security database
Database is shut down.
Termination of FileNet software completed.
MKF_clean                          19787
Starting NCH_daemon...
Starting OCOR_Listen...
```

---

## Checklist

Before you use initbackup, be aware of the following:

- Using the Task Manager Backup button is the preferred method of establishing the correct environment for running backup.
- initbackup calls the initfnsw command with the backup option.

## Procedure

- 1 Shut down all servers.
- 2 On each server, enter the following command:

**initbackup**

## Related Topics

### **[“endbackup” on page 401](#)**

initfnsw command with the backup option

See the “Backup” chapters of the following documents for additional information on preparing for backup:

- *System Administrator’s Companion for UNIX*
- *System Administrator’s Companion for Windows Server*
- *Enterprise Backup/Restore User’s Guide*

To download IBM FileNet documentation from the IBM support page, see **[“Accessing IBM FileNet Documentation” on page 21](#)**.

# initfnsw

## Description

The initfnsw program performs different initialization actions depending on the option you specify. The following is a description of the actions performed with each initfnsw option:

### **initfnsw backup**

After FileNet software is stopped, places FileNet software in the appropriate state to perform a backup.

### **initfnsw start**

Starts FileNet software and FileNet databases.

### **initfnsw stop**

Stops FileNet databases and FileNet software.

### **initfnsw restart**

Stops FileNet software, then starts it.

### **initfnsw restore**

Stops FileNet software, then starts FileNet processes necessary for a restore.

### **initfnsw status**

Displays the Image Services software status.

**initfnsw terminate**

Stops FileNet software and terminates the TM\_daemon. The terminate option is normally used to shut down the FileNet system before a software upgrade.

**Note** The following three options have to do with the fact that backup of magnetic storage requires the Image Services software to be shutdown. The addition of these three scriptable commands to pause and resume magnetic storage input and output and service network requests eliminates the need for this required shutdown on IS servers running AIX. These commands make the downtime of these devices as short as possible, eliminating increased response times and possible failures. It is expected that the customer, partner, or Professional Services will provide scripts to use these commands and interact with the RDBMS and the magnetic disk. The script should pause the IS software, pause the RDBMS, perform the storage split, resume the RDBMS, and resume IS.

---

**initfnsw pause (AIX only)**

Pauses magnetic disk input/output activity.

**initfnsw resume (AIX only)**

Resumes magnetic disk input/output activity.

**initfnsw ispaused (AIX only)**

Reports to stdout whether or not input/output activity is currently paused.



## Use

This command provides a command line method of initiating tasks more easily run through the FileNet Task Manager.

- Use `initfnsw backup` to prepare a server to be backed up.
- Use `initfnsw` to start, stop, and restart FileNet software.

### WIN

The `initfnsw start`, `restart`, and `stop` commands have no effect on the `TM_daemon` that runs as a service in the Image Services for Windows Server. Use the Services applet of the Windows Control Panel to start, restart, or stop the `TM_daemon` independently of the Image Services software. For more information, refer to the “Getting Started” chapter of the *System Administrator’s Companion for Windows Server*.

To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

## Syntax

---

**Tip** You can display `initfnsw` syntax by entering **`initfnsw`** with no parameters at the command line.

---

The `initfnsw` syntax is:

```
initfnsw [-y] [-m] [-h <hostname>] {start | stop | backup | restart |
status | terminate | pause | resume | ispaused}
```

**-m** Maintains software state when errors are encountered

If you specify this option, FileNet software does not terminate if errors are encountered during an `initfnsw` operation.

**-h <hostname>** Issues request to the host specified by <hostname>

The default host name is the local host.

The TM\_daemon must be running on the specified host. If you do not specify host name, or you specify the local host, initfnsw attempts to start the TM\_daemon whether or not a TM\_daemon is already running.

**-y** Do not display a confirmation message. (This option is available only with the stop, backup, restore, restart, and terminate options.)

- If you specify this option, initfnsw automatically starts processing.
- If you do **not** specify the -y option, initfnsw displays a confirmation message. Continue or terminate the process by responding to the message.

---

**Tip** If you use the -y option, you must enter before the other option on the command line (for example: initfnsw -y stop).

---

**start** Initializes FileNet software

**stop** Terminates FileNet software

**backup** Shuts down the RDBMS index database and MKF databases and initializes a minimal FileNet environment in preparation for backup of FileNet software and databases. You must run initfnsw stop **before** initfnsw backup.

**restore** Starts FileNet processes required for a restore operation

**restart** Stops then starts all FileNet programs

**status** Displays FileNet software status information

**terminate** Stops FileNet software and terminates the TM\_daemon

---

**Note** The following three options are only available on an IS system running AIX and are designed to be scripted commands.

---

**pause** Pauses magnetic disk input/output activity

**resume** Resumes magnetic disk input/output activity

**ispaused** Reports to stdout whether or not input/output activity is currently paused

## Sample Output

The following displays are sample output for frequently used `initfnsw` commands.

### `initfnsw start` Sample

After you issue the `initfnsw start` command, you see output similar to the following:

```
monterey (fnsw) /fnsw/bin> initfnsw start
Terminating processes...
Initializing FileNet software...
Starting index database...
Starting permanent database...
Starting transient database...
Starting security database...
Starting OCOR_Listen...
Starting NCH_daemon...
Starting the Security Daemon...
Starting INXbg...
Starting INXu...
Starting document services...
Starting batch_entry services...
Starting print services...
Startup of FileNet software initiated. See event log for detailed status.
```

If you attempt to use `initfnsw start` when `TM_daemon` is not running, the following error message displays:

```
Check to ensure that the IS ControlService and TM_daemon are running.
```

### **WIN**

Examine the status of the `IS ControlService` process from the Windows Control Panel's Services applet and start the process and `TM_`

daemon if necessary. For more information about starting and stopping the IS ControlService process and TM\_daemon, see the “Getting Started” chapter of the *System Administrator’s Companion for Windows Server*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

### initfnsw stop Sample

The **initfnsw stop** command terminates all FileNet software. After responding to the confirmation prompt, initfnsw stop displays the following output:

```
monterey (fnsw) /fnsw/bin> initfnsw stop
Are you sure you want to stop software on server 'monterey'? (y | n) [n] :
y
Terminating FileNet software...
Shutting down security database...
Shutting down permanent database...
Shutting down transient database...
Terminating processes...
Shutting down index database...
Termination of FileNet software completed.
```

### initfnsw backup Sample

You must stop FileNet software before you run **initfnsw backup**. If you run initfnsw backup while FileNet software is active, the following message displays:

```
monterey (fnsw) /fnsw/bin> initfnsw backup
Software is not stopped. Request denied!
```

The `initfnsw backup` command starts `NCH_daemon`, `OCOR_Listen`, `MKF_clean`, and some network programs. These components of the FileNet system must be running to perform a backup.

After stopping FileNet software, a successful `initfnsw backup` command displays output similar to the following:

```
monterey (fnsw) /fnsw/bin> initfnsw backup  
Initializing FileNet software for backup mode...  
Starting OCOR_Listen...  
Starting NCH_daemon...  
Startup of FileNet software for backup mode has been initiated.
```

### **initfnsw restore Sample**

To prepare for a restore operation, use `initfnsw restore`. The `initfnsw restore` command terminates all FileNet processes then establishes the proper environment for running a restore operation. The command prompts you for confirmation, as shown in the following output:

```
monterey (fnsw) /fnsw/bin> initfnsw restore  
Are you sure you want to enter restore mode on server 'monterey'? (y | n)  
[n] : y  
Terminating processes...  
Initializing FileNet software for restore mode...  
Starting OCOR_Listen...  
Starting NCH_daemon...  
Startup of FileNet software for restore mode has been initialized.
```

## initfnsw status Sample

To obtain the status of the FileNet software, run **initfnsw status**:

```
monterey (fnsw) /fnsw/bin> initfnsw status  
Software status for host 'monterey' (operating system = HPUX):  
    Software started since Wed Nov 20 11:04:21 1996
```

## Checklist

Before you use initfnsw, be aware of the following:

- You might prefer using the FileNet Task Manager to perform the tasks accomplished with initfnsw options.
- The initfnsw start, restart, and stop commands do not effect the TM\_daemon that runs as a service in the Image Services for Windows Server.

WIN

## Procedure

No specific procedure is required to start, stop, or restart FileNet software. However, Task Manager functions are the preferred method for performing these tasks.

For correct procedures to establish backup mode, refer to your FileNet backup documentation (see [“Related Topics” on page 592](#)).

For information about restoring your Image Services system, call your service representative. If you use Enterprise Backup/Restore, refer to the *Enterprise Backup/Restore User's Guide* for restore information. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

## Related Topics

To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

*System Administrator’s Handbook* or online help for information on using Task Manager.

“Backup” chapter in your *System Administrator’s Companion for UNIX* or *System Administrator’s Companion for Windows Server* for backup information.

“Getting Started” chapter of *System Administrator’s Companion for Windows Server* for information on starting and stopping TM\_daemon.

If you use Enterprise Backup/Restore to back up and restore your FileNet system, refer to the *Enterprise Backup/Restore User’s Guide*.



# INXdebug

## Description

INXdebug provides performance analysis information about Index Services. The INXdebug tool links to INX and sets a new debug value. Your service representative provides you with a hexadecimal debug value that represents the INX function to be analyzed.

## Use

Use INXdebug, only at the direction of your service representative, to investigate performance problems in Index Services.

### **CAUTION**

---

Do not run INXdebug unless instructed to do so by your service representative. **Disruption of in-progress functions can occur.**

---

## Syntax

```
INXdebug <hex debug value>
```

### **<hex debug value>**

A hexadecimal value representing the function to be analyzed.

Your service representative provides the value.

## Checklist

Before using INXdebug, be aware of the following:

- Do not run this program unless instructed to do so by your service representative.
- Your service representative provides a hexadecimal debug value.

## Procedure

Call your service representative for the procedure and the hexadecimal debug value.

# INX\_export

## Description

The INX\_export tool can be used to export existing document index information associated with an IS document class to make it available for import by a Content Engine (CE) system.

If you have a Remote Admin Console (RAC) attached to your system, there is a GUI version of the tool called the **CFS Connector - IS Catalog Export Tool**. It is accessed from the Application pulldown on the main Applications Executive screen of your RAC. Please see IS Catalog Export Tool related chapter of the *Remote Admin Console User's Guide* for more information on this tool. There is online help available when running the tool as well.

## Use

Logon to the Image Services Root/Index or Combined Server as fnswh and run **INX\_export** from the command line.

## Syntax

```
INX_export -n <nameDocClass> [-l <lowDocID>] [-h <highDocID>]
[-r] [-a | -d] [-g {<objStoreGuid>}] [-v]
```

- n <nameDocClass>** The name of the document class being exported.
- l <lowDocID>** Optional: Low Doc ID value - default is 100000.
- h <highDocID>** Optional: High Doc ID value - default is 3999999999.

- 
- r Optional: Forces a re-export of the catalog entries associated with the IS document class - default is do not re-export.
  - a Optional: Exports annotations only without exporting catalog entry - default is to export annotations and catalog entry together.

---

**Note** The -a and -d options are mutually exclusive.

---

- d Optional: Deletes the Image Services catalog entries associated with the document IDs falling in the range of documents to be exported from the Image services system as indicated by the -l and -h options after exporting - default is do not delete.

---

**CAUTION** Using the -d option removes the catalog entries from the Image Services Index Database making these document IDs inaccessible in the future.

---

**-g<{objStore  
Guid}>** Optional: Object Store GUID - if default object store guid is configured, guid must be enclosed with {}.

- v Verbose mode.

## Examples

In the example below, You are exporting Document IDs 100000 through 3999999999 of the Federation document class with no re-exports, and without deleting the document IDs from the IS system.

**INX\_export -n Federation**

In the example below, You are exporting Document IDs 105000 through 200000 of the Federation document class with no re-exports, deleting the document IDs from the IS system, and in verbose mode.

```
INX_export -n Federation -l 105000 -h 200000 -d -v
```

In the example below, You are exporting the Federation document class to the CE object store guid D3C9E071-62A6-4C3E-BA23-19383E939869.

```
INX_export -n Federation -g {D3C9E071-62A6-4C3E-BA23-19383E939869}
```

## Procedure

The following procedure is an example used to run the INX\_export command.

- 1 On the Root/Index server log on as **fns**.
- 2 Enter a command similar to the following depending on what you want to do:

```
INX_export -n <docClass> -g {objStoreGuide}
```

## Related Topics

The IS Catalog Export Tool related chapter of the *Remote Admin Console User's Guide*.

# INX\_tool

## Description

INX\_tool provides the ability to view and release index services (INX) capability locks and also provides command interaction with Index Services via the new CE\_INFO and CE\_CONF commands. INX uses capability locks to prevent two clients from updating the same object at one time. The following objects can obtain capability locks:

- documents
- folders
- tabs
- indexes
- document classes
- clusters
- menus
- validation tables

Under normal circumstances, capability locks are automatically released following an update. On occasion, the release does not occur. For example, a client can lock a document but, due to a bug in the client application code or an application program failure, fail to release the lock when the update completes. In another example, a lock can be held when a PC running in native mode while updating a document in a Query Match Report is accidentally rebooted or must be rebooted because of an operating system error.

## Use

Use INX\_tool to manually unlock objects when software fails to do so. Such occurrences are rare.

Do not use INX\_tool to unlock an object being used by other clients. If a program locks an object for updating and you remove the lock on that object with INX\_tool, more than one client at a time could update the object, compromising the integrity of the database and the state of internal shared memory.

To determine if an object is locked, examine the time stamp of the object. An object is usually locked for only a short time. If the time stamp indicates an object has been locked for a long time (hours or days), it is reasonable to consider unlocking the object with INX\_tool. If the application program is structured to allow locking objects for long periods of time, an application-dependent method of judging failures to unlock must be invoked.

## Syntax

INX\_tool

When you enter INX\_tool at the command line, the prompt changes to <INX\_tool>.

Enter INX\_tool commands at this prompt.

## Commands

The following table identifies all available INX\_tool commands with a brief description of each. The list is followed by detailed explanations of each command.

Command	Description
CAPLOCKS [<types>]	Lists capability locks. If types are not specified, lists all capability locks.
UNLOCK <index>	Releases capability lock on an object
CE_INFO	Displays Content Engine configuration and FileNet P8 Content Federation Services (CFS) Server for Image Services information.
CE_CONF	Configures IS document class mapping to a CE object store in a CFS for IS environment.
OUTPUTFILE <filename>	Specifies the name of an output file into which INX_tool output is directed
OUTPUT [OFF   ON]	Turns off or on the output to the output file specified in the OUTPUTFILE command
EXIT	Exits INX_tool
QUIT	Exits INX_tool
HELP [<command>]	Displays help information for INX_tool commands
!<shell_cmd>	Executes a specified shell command
?	Lists a brief description of the available commands



**caplocks [<types>]**

<types> can be one or more of the following:

A - available locks

C - document class locks

D - document locks

F - folder locks

I - index locks

L - cluster locks

M - menu locks

T - tab locks

V - validation table locks

\* - all locks

If you do not specify an object type, caplocks displays locks on all object types.

The following example shows two locked documents with doc numbers 1901 and 1900:

```
<INX_tool> caplocks
```

idx	type	#	object id	timestamp	next idx
---	-----	---	-----	-----	-----
23	doc	1	1901	97/01/27 17:06:17.453	24
24	doc	1	1900	97/01/27 17:06:12.547	-1

The first column, idx, contains the number that uniquely identifies the capability lock. The second column, type, identifies the object type that is locked. The type column contains both a type description, such as doc, and the internal type number (#). The third column is the object identifier. If the object type is doc, the object ID column holds the document's doc number. If the object type is menu, the column holds the menu's identifier, and so on. The fourth column contains the time that the object was locked. The last column contains internal indexing information and can be ignored.

### **unlock <index>**

To remove a lock and release the object, use the unlock command followed by the capability lock identifier. (Obtain the identifier from the first column of the caplocks output.) INX\_tool displays the capability lock and prompts you to confirm the action before the lock on the object is removed. Before you respond to the prompt, review the displayed information to verify that this is the correct lock to remove.

The following example shows the removal of the lock for object 23:

```
<INX_tool> unlock 23
```

idx	type	#	object id	timestamp	next idx
23	doc	1	1901	97/01/27 17:06:17.453	24

```
Unlock index 23 (y/n) ? y
<INX_tool>
```

### **ce\_info**

To display Content Engine information and CFS Server for Image Services information in a variety of ways, use the many menu-driven view options available with the `ce_info` command.

The following example shows the main display of the `cd_info` command:

```
<INX_tool> ce_info
```

```
CE INFO views
  1) Display All configured Object Stores
  2) View Object Store to DCL relationship(s)
  3) View DCL to Object Store relationship(s)
  4) View DCL default Object Store configuration
  5) View CE Shared Memory Management Variables
Select one(1-5):
```

If you enter 1, you will receive a screen similar to the following which will display object store information after answering some system prompts:

```
< Select one(1-5):1
Enter the NCH domain name [ibmmsar]:
Enter the NCH organization name [FileNet]:
Enter the logon user name [SysAdmin]:
Enter the logon password [*****]:
Known Domain/OBJECT_STORE(s):
  DOMAIN: 'ISNTTITAN'='{C8A20902-43EB-49BF-9B65-266B07140865}'
  Object Store: 'TESTCASE_B_OS'='{D5A78CDF-1BE6-423E-9CB3-67EE04AEA70C}'
  CE_OS_ID=1000
  Configured DCL(s):
    DVT_docclass1
    DVT_docclass2

  DOMAIN: 'INT3P8'='{3DF5A8EE-083C-4069-9B3E-90007E0D47C8}'
  Object Store: 'OS_BESTCASE'='{BF8AA25E-9559-43E2-A021-4EC18FA6F6EA}'
  CE_OS_ID=1001
  Configured DCL(s):
    DVT_docclass1
    doc_from_ce
    ce_class
    ce_class2
```

You enter the appropriate responses to the prompts. All prompts are self-explanatory and will provide an accurate display of your desired view.

## ce\_conf

The `ce_conf` command configures Images Services document class mapping to a Content Engine object store.

The following example shows the main display of the `cd_info` command:

```
<INX_tool>CE_CONF

Enter the NCH domain name [ibmmsar]:
Enter the NCH organization name [FileNet]:
Enter the logon user name [SysAdmin]:
Enter the logon password [*****]:
Known CE_DOMAIN/OBJECT_STORE(s):
  DOMAIN: 'CE'='{12345678-1234-1234-1234-123456789012}'
  Object store: 'OS'='{12345678-1234-1234-1234-123456789013}'
  DOMAIN: 'CE2'='{12345678-1234-1234-1234-123456789013}'
  Object store: 'OS2'='{12345678-1234-1234-1234-123456789044}'
  DOMAIN: 'CE2'='{12345678-1234-1234-1234-123456789013}'
  Object store: 'OS3'='{12345678-1234-1234-1234-123456789555}'
  DOMAIN: 'ISNTTITAN'='{0C56A5DC-1F6F-437D-A5B0-9C15A0C9BCE6}'
  Object store: 'CE_B_OS'='{5590A000-43AD-4DD2-A1D3-38B51D995E16}'
  DOMAIN: 'ISNTTITAN'='{0C56A5DC-1F6F-437D-A5B0-9C15A0C9BCE6}'
  Object store: 'CE_C_OS'='{84A7DF45-B1F7-40B2-86A2-46EB496CB7A7}'

CE_DOMAIN_NAME:
```

Normally this relationship should be managed through Enterprise Manager and the CFS Connector - IS Catalog Export Tool on RAC. For more information on this tool, see Chapter 4 of the *Remote Admin Console User's Guide*.

You are prompted for specific information for the CE domain you are interested in. In our example below, we chose CE:

```
CE_DOMAIN_NAME: ce
CE_DOMAIN_GUID: {12345678-1234-1234-1234-123456789012}
OBJECT_STORE_NAME: OS
OBJECT_STORE_GUID: {12345678-1234-1234-1234-123456789013}
Add DCL=IS_C_DCL (id=90) to CE_OS_DCL_MAP (y/n):n
Add DCL=ce_im_only (id=1) to CE_OS_DCL_MAP (y/n):y
Make this the default Object Store for DCL=ce_im_only(y/n) [n]:y
Add DCL=ce_no_cat_class (id=2) to CE_OS_DCL_MAP (y/n):n
Add DCL=ce_class (id=3) to CE_OS_DCL_MAP (y/n):n
Add DCL=ce_class2 (id=4) to CE_OS_DCL_MAP (y/n):n
Add DCL=not_ce_class (id=5) to CE_OS_DCL_MAP (y/n):n
Add DCL=modeldefault (id=6) to CE_OS_DCL_MAP (y/n):n
Add DCL=COLD_committals (id=7) to CE_OS_DCL_MAP (y/n):n
Add DCL=PDE_DM1_CATY_nums (id=8) to CE_OS_DCL_MAP (y/n):n
Add DCL=CE_IS_Scratch (id=9) to CE_OS_DCL_MAP (y/n):n
Add DCL=PDE_CATN_6 (id=10) to CE_OS_DCL_MAP (y/n):n
INX_ce_config_object_store_dcl_map: successfully called
successfully set DCL='ce_class' (id=3)
Object_store_guid={12345678-1234-1234-1234-123456789013} DEFAULT
```

Note that when answering the prompt for the CE domain and object store name GUIDs, you must enter in the brackets (**{}**) in addition to the entire string of numbers.

In the example above, notice that when you answer Y to add a document class, you are also asked if you want to make the object store the default object store. Once you have gone through all of the document classes you are given a summary of all of the successful document class mappings as well as which object store is the default.

**outputfile <filename>**

INX\_tool typically displays its output to the standard output device. You can optionally specify a file to which INX\_tool directs its output. For example, the following command causes INX\_tool to write its output to the inx010397.out file in the /tmp directory:

```
<INX_tool> outputfile /tmp/inx010397.out
```

After using the outputfile command, use the **output** command to write INX\_tool output to the specified file.

**output [off | on]**

Use the **output on** command to cause INX\_tool to write its output to the file you specified in the outputfile command. If you want the output to be displayed only to the standard output device, use the **output off** command.

**exit / quit**

Use either the exit or quit command to terminate INX\_tool.

**help [<command>]**

Use the help command to display additional information on one or all INX\_tool commands. If you do not specify a command, the help command displays brief information for all INX\_tool commands.

The following example displays help for the outputfile command:

```
<INX_tool> help outputfile

OUTPUTFILE <file name> - establishes <file name> as the file to which output
                        will be directed. The OUTPUT ON command must be used
                        to cause output to be written to <file name>.

<INX_tool>
```

**!**<shell\_cmd>****

Executes the specified shell command.

**?**

Lists brief help information for each INX\_tool command.



## Checklist

Before you use INX\_tool, be aware of the following:

- To use INX\_tool, you must be logged on as a member of the fnusr group.
- You must run INX\_tool from the Index server.

## Procedure

No specific procedure is required.

## Related Topics

Information on the CFS Connector - IS Catalog Export Tool on RAC, can be found in Chapter 4 of the *Remote Admin Console User's Guide*.

## ipc\_tool

### Description

The ipc\_tool enables you to examine the resources that allow Image Services processes to interact. These resources include:

- **Shared memory:** stores common data, accessed by multiple processes.
- **Semaphores:** control guarding abstract initialization and access to system resources.
- **Interlocks:** control access to critical sections of code.

You can run ipc\_tool as a menu-driven utility or you can enter the ipc\_tool command with one of the options at the system prompt.

## Running ipc\_tool as a Menu-Driven Utility

When you enter `ipc_tool` at the command line without flags, the following menu and prompt appear:

```
Welcome to ipc_tool

1. Abstract Shared Memory (detailed)
2. Abstract Shared Memory (summary)
3. Address to Abstract Mapping
4. Examine Shared Memory
5. System Management Semaphores
6. Claimed Application Interlocks
7. Interlock usage of a process
8. Current Processes
9. Dump Shared Memory Segment(s) to File(s)
A. Shared Memory Segment Information
->
```

As shown in this example, each menu option identifies the type of information displayed after you type the corresponding character. For example, typing an uppercase letter **A** displays the Shared Memory Segment Information report.

See [“Commands” on page 615](#) for a description of these menu options.

To quit displaying `ipc_tool` output, type **q**. To exit the menu, type **q** (quit) at the prompt (->).

## Running ipc\_tool from the Command Line

If you prefer to run `ipc_tool` commands from the command line, simply enter the appropriate flag after the **ipc\_tool** command.

---

The following example shows output displayed on an AIX platform when running the **ipc\_tool -A** option from the command line.

---

```
moorea(fnsw)/home/fnsw> ipc_tool -A
```

```
Maximum shared memory segments available in software: 11  
Segment Size: 0x01000000 bytes (16 MB)
```

```
Successfully test allocated and attached to 10 shared memory segments.  
Segment #0 (called the address manager) is small.  
The other 9 segments are 16 MB each for total shared memory = 144 MB.  
The actual amount of shared memory available during operation may be less.
```

```
Shared Memory Address Manager Information
```

Address	Shm id	Creator
0 0xc0000000	102401	Shared address manager
1 0xb0000000	618498	FileNet server software

```
System's fixed shared memory address table (derived)...
```

```
0: 0xc0000000  
1: 0xb0000000  
2: 0xa0000000  
3: 0x90000000  
4: 0x80000000  
5: 0x70000000  
6: 0x60000000  
7: 0x50000000  
8: 0x40000000  
9: 0x30000000
```

---

To quit displaying ipc\_tool output, type **q**.

## Use

Use `ipc_tool` to examine the resources which allow Image Services processes to interact. For example:

- If a process hangs, `ipc_tool` enables you to determine if any claimed semaphores or interlocks are delaying processes.
- When shared memory corruption causes problems with one or more processes, `ipc_tool` enables you to examine shared memory, which might help you find the source of the corruption.
- When you're using `fn_edit` to tune system performance by adjusting the shared memory and buffer size values, `ipc_tool` enables you to determine the number of shared memory segments in use.

---

**Note** The `ipc_tool` only provides information and does not modify the system in any way.

---

---

## Syntax

```
ipc_tool [ -a | -t | -m | -d | -s | -i | -I | -p | -D | -A ]
```

You can run the `ipc_tool` command with one of its optional flags at the system prompt. Entering the `ipc_tool` command without any flags displays the `ipc_tool` menu.

See “Flags” below for a description of each flag.

## Flags

Flags for `ipc_tool` correspond to selections from the `ipc_tool` menu. See [“Commands” on page 615](#) for a detailed description of the information associated with each flag:

- a** Abstract Shared Memory (detailed)
- t** Abstract Shared Memory (summary)
- m** Address to Abstract Mapping
- d** Examine Shared Memory
- s** System Management Semaphores
- i** Claimed Application Interlocks
- I** Interlock usage of a process
- p** Current Processes
- D** Dump Shared Memory Segment(s) to File(s)
- A** Shared Memory Segment Information

## Commands

The following items correspond to the menu options displayed when you start the menu-driven form of ipc\_tool. Each option also corresponds to a flag that you can directly enter to select a specific option at the time you start ipc\_tool from the command line.

### 1. Abstract Shared Memory (detailed)

(corresponds to the -a flag)

Produces a list of shared memory blocks grouped by shared library (abstract) name with the starting address and size of each block displayed. If the program detects data corruption at a certain address, ipc\_tool halts and produces a report of the problem.

At the end of the output, ipc\_tool reports the average block size of the shared library and the total number of bytes allocated for the shared library. The -a option can help when you are trying to determine which shared libraries have been linked since the FileNet software started.

The following example shows a partial listing of the report displayed when selecting this option on an AIX platform. The report displayed for your Image Services system might vary from this example. This example omits the complete data where indicated with an ellipsis (...).

---

```
SHM:
1: Address: 0xb0000000 Size:      6244
2: Address: 0xb068b09c Size:      528
...
SHM:  Total bytes .....    9902768

ARM:
1: Address: 0xb066aef4 Size:      6636
```

---

2: Address: 0xb066d078 Size: 980  
3: Address: 0xb066d45c Size: 6416  
ARM: Avg: 4677.33 Bytes: 14032

## ASH:

1: Address: 0xb00cef9c Size: 28  
ASH: Avg: 28.00 Bytes: 28

## BES:

1: Address: 0xb0464314 Size: 68  
2: Address: 0xb04656c8 Size: 1288  
3: Address: 0xb0689cbc Size: 1288  
BES: Avg: 881.33 Bytes: 2644

## BESl:

1: Address: 0xb0465388 Size: 132  
BESl: Avg: 132.00 Bytes: 132

...

## CNF:

1: Address: 0xb0419444 Size: 46812  
2: Address: 0xb0424b30 Size: 112  
3: Address: 0xb0424bb0 Size: 480  
CNF: Avg: 15801.33 Bytes: 47404

## CNT:

1: Address: 0xb03dfa74 Size: 7984  
CNT: Avg: 7984.00 Bytes: 7984

## COR:

1: Address: 0xb000d22c Size: 136  
2: Address: 0xb004e8e4 Size: 1980  
COR: Avg: 1058.00 Bytes: 2116

## CSM:

1: Address: 0xb0462340 Size: 48  
2: Address: 0xb0463004 Size: 908  
CSM: Avg: 478.00 Bytes: 956

## CSMl:

1: Address: 0xb041832c Size: 4360  
2: Address: 0xb0444ae4 Size: 112



```
...
CSM1:  Avg: 399.38    Bytes:    5192
...

DIS:
1: Address: 0xb04484f4 Size:    8468
DIS:  Avg: 8468.00   Bytes:    8468
...

DOC:
1: Address: 0xb04633cc Size:    144
2: Address: 0xb046346c Size:    676
DOC:  Avg: 410.00    Bytes:    820

...

GTI:
1: Address: 0xb000d304 Size:    224
2: Address: 0xb000d3f4 Size:    6160
...
GTI:  Avg: 11284.25  Bytes:   180548
...

INX:
1: Address: 0xb04172a4 Size:    40
2: Address: 0xb041741c Size:   2348
...
INX:  Avg: 536.11    Bytes:   101860
...

MKFt:
1: Address: 0xb000be1c Size:    5004
2: Address: 0xb004f0e0 Size:    1424
...
MKFt:  Avg: 19292.68 Bytes:  3762072

NCH:
1: Address: 0xb0007374 Size:    312
2: Address: 0xb00074bc Size:    272
...
```

---

NCH: Avg: 3806.40 Bytes: 19032

NCH1: Linked, but no shared memory.

...

NLT:

1: Address: 0xb00cee40 Size: 60

2: Address: 0xb00e62c8 Size: 1576

...

NLT: Avg: 632.67 Bytes: 3796

NMI:

1: Address: 0xb066972c Size: 832

2: Address: 0xb0670e30 Size: 864

...

NMI: Avg: 591.20 Bytes: 2956

OCOR:

1: Address: 0xb0017cd0 Size: 223352

2: Address: 0xb004e558 Size: 60

...

OCOR: Avg: 56053.00 Bytes: 224212

...

ODX:

1: Address: 0xb066c8f0 Size: 892

2: Address: 0xb068da14 Size: 1720

ODX: Avg: 1306.00 Bytes: 2612

OPM:

1: Address: 0xb044b8f0 Size: 88

2: Address: 0xb044b958 Size: 48

3: Address: 0xb044b998 Size: 48

OPM: Avg: 61.33 Bytes: 184

OSA:

1: Address: 0xb044a618 Size: 160

2: Address: 0xb044b9d8 Size: 1736

OSA: Avg: 948.00 Bytes: 1896

```
OSI:
1: Address: 0xb066ed7c Size:      48
2: Address: 0xb0670738 Size:      68
OSI:   Avg: 58.00      Bytes:    116
```

```
OSI1:
1: Address: 0xb066f678 Size:    4272
2: Address: 0xb06707b4 Size:     140
OSI1:  Avg: 2206.00   Bytes:   4412
```

...

```
PMM:
1: Address: 0xb0465610 Size:     32
2: Address: 0xb046937c Size:    256
PMM:   Avg: 144.00   Bytes:    288
```

...

```
PR11:
1: Address: 0xb0460094 Size:   8860
2: Address: 0xb046948c Size:    596
```

...

```
PR11:  Avg: 192006.19 Bytes: 2112068
```

...

```
QLG:
1: Address: 0xb044583c Size:     204
2: Address: 0xb0445918 Size:   10256
QLG:   Avg: 5230.00   Bytes:  10460
```

```
QMA:
1: Address: 0xb0405e54 Size:     192
2: Address: 0xb0415ec4 Size:     696
```

...

```
QMA:   Avg: 779.00   Bytes:   9348
```

...

```
SAS:
1: Address: 0xb0463950 Size:     336
2: Address: 0xb0463ab0 Size:   2064
SAS:   Avg: 1200.00   Bytes:   2400
```

---

```
SC:      Linked, but no shared memory.

SEC:
1: Address: 0xb03a1320 Size:      264
2: Address: 0xb0464f54 Size:      20
...
SEC:   Avg: 48.00      Bytes:      528
...

SKF:
1: Address: 0xb03e19b4 Size:     11972
SKF:   Avg: 11972.00  Bytes:     11972

SKFt:
1: Address: 0xb03b52b0 Size:      596
2: Address: 0xb03e4888 Size:     5964
SKFt:  Avg: 3280.00   Bytes:     6560

SLM1:
1: Address: 0xb00018bc Size:      68
2: Address: 0xb0405f94 Size:      84
...
SLM1:  Avg: 114.36    Bytes:     2516
...

WQSl:
1: Address: 0xb00cef00 Size:      28
2: Address: 0xb03ff3e8 Size:     976
WQSl:  Avg: 502.00    Bytes:     1004

WRT:
1: Address: 0xb046392c Size:      20
WRT:   Avg: 20.00     Bytes:      20

Total bytes of shared memory:....6862992
```

---

The following example shows a message ipc\_tool displayed after detecting data corruption in the MKF shared library (also called an “abstract”):

```
Corruption detected at address <0x300006c0> for abstract
<MKF>!
Leading size of <32> doesn't match trailing size of <2171424>.

MKF:
1: Address: 0x300006c4  Size:   32  <- block corrupt!
SKF:   Avg: 32.00      Bytes:  32
```

This tool can detect data corruption when the shared memory block allocated for a shared library does not match the block size values stored in adjacent shared memory addresses.

## 2. Abstract Shared Memory (summary)

(corresponds to the -t flag)

Produces a summary of shared memory use by each shared library (abstract). The output is an abbreviated version of the information provided by the detailed abstract shared memory report (menu option 1 or flag -a).

The following example shows a partial listing of the report displayed when selecting this option on an AIX platform. The report displayed for your Image Services system might vary from this example.

Abstract	Total	Avg Block	Total Bytes
-----	-----	-----	-----
ARM	3	4677.33	14032
ASH	1	28.00	28
BES	3	881.33	2644
BES1	1	132.00	132
BKG	1	52.00	52
BRLC	1	80.00	80
CDB	1	20.00	20
CDCD	0	0.00	0
CKS	1	20.00	20
CLT	1	24.00	24
CMT	1	24.00	24
CNF	3	15801.33	47404
CNT	1	7984.00	7984
COR	2	1058.00	2116
CSM	2	478.00	956
CSM1	13	399.38	5192
DBL	1	336.00	336
DBP	1	480.00	480
DBU	1	20.00	20
DCH	1	304.00	304
...			

**Abstract**

The name of the shared library (abstract).

**Total**

Total number of blocks allocated for use by the shared library.

**Avg Block**

Average size (in bytes) of each allocated memory block.

**Total Bytes**

Amount of memory (in bytes) used by the shared library.  
The calculation is  $\text{Total} * \text{Avg Block} = \text{Total Bytes}$ .

### 3. Address to Abstract Mapping

(corresponds to the -m flag)

Identifies the shared library (abstract) that is using a memory address that you specify. Use this option to identify a shared library when you know the address where data corruption has occurred but you do not know which shared library is using that memory address. After selecting this option, the program displays output similar to the following:

```
Segment 1: 0xb0000000 - 0xb0ffffff
Enter Address:
```

If you enter the hexadecimal address 0xb0112230, the program displays output similar to the following:

```
Enter Address: 0xb0112230

MKFt:
1: Address: 0xb00e6b18 Size: 200096
```

This information tells you that the MKF shared library is using memory located at address 0xb0112230, which is part of a shared memory segment starting at address 0xb00e6b18.

#### Note

If memory is corrupted at the address you specify, the information reported might not be reliable. If you suspect this condition, menu option 4 (Examine Shared Memory) could help in troubleshooting.



## 4. Examine Shared Memory

(corresponds to the -d flag)

Provides a hexadecimal and ASCII view of shared memory addresses. To display an address and the addresses immediately following it, enter the address at the prompt.

The following example shows a partial listing of the report displayed when selecting this option on an AIX platform. The report displayed for your Image Services system might vary from this example.

---

```
-> 4
Segment 1: 0xb0000000 - 0xb0ffffff
Enter Address: 0xb0112230
  b0112220: 00000000 B0350E00 00000000 00000000 .... .5.. ....
> b0112230: 00000000 00000000 00000000 00000000 .... .... ....
  b0112240: 00000000 00000000 00000000 00000000 .... .... ....
  b0112250: 00000000 00000000 00000000 00000000 .... .... ....
  b0112260: FFFFFFFF 00000000 000008B0 00000000 .... .... ....
  b0112270: 00000000 B0351200 00000000 00000000 .... .5.. ....
  b0112280: 00000000 00000000 00000000 00000000 .... .... ....
  b0112290: 00000000 00000000 00000000 00000000 .... .... ....
  b01122a0: 00000000 00000000 00000000 00000000 .... .... ....
  b01122b0: FFFFFFFF 00000000 000008B1 00000000 .... .... ....
  b01122c0: 00000000 B0351600 00000000 00000000 .... .5.. ....
  b01122d0: 00000000 00000000 00000000 00000000 .... .... ....
  b01122e0: 00000000 00000000 00000000 00000000 .... .... ....
  b01122f0: 00000000 00000000 00000000 00000000 .... .... ....
  b0112300: FFFFFFFF 00000000 000008B2 00000000 .... .... ....
  b0112310: 00000000 B0351A00 00000000 00000000 .... .5.. ....
  b0112320: 00000000 00000000 00000000 00000000 .... .... ....
  b0112330: 00000000 00000000 00000000 00000000 .... .... ....
  b0112340: 00000000 00000000 00000000 00000000 .... .... ....
```

---

```
b0112350: FFFFFFFF 00000000 000008B3 00000000 .....  
b0112360: 00000000 B0351E00 00000000 00000000 .....5..  
b0112370: 00000000 00000000 00000000 00000000 .....  
b0112380: 00000000 00000000 00000000 00000000 .....  
b0112390: 00000000 00000000 00000000 00000000 .....  
b01123a0: FFFFFFFF 00000000 000008B4 00000000 .....  
b01123b0: 00000000 B0352200 00000000 00000000 .....5".  
b01123c0: 00000000 00000000 00000000 00000000 .....  
b01123d0: 00000000 00000000 00000000 00000000 .....  
b01123e0: 00000000 00000000 00000000 00000000 .....  
b01123f0: FFFFFFFF 00000000 000008B5 00000000 .....  
b0112400: 00000000 B0352600 00000000 00000000 .....5&.  
b0112410: 00000000 00000000 00000000 00000000 .....  
b0112420: 00000000 00000000 00000000 00000000 .....  
b0112430: 00000000 00000000 00000000 00000000 .....  
b0112440: FFFFFFFF 00000000 000008B6 00000000 .....  
b0112450: 00000000 B0352A00 00000000 00000000 .....5*..  
.....
```

---

Use this option when you know the address at which memory is corrupted but other `ipc_tool` options (such as menu option 3) are unable to identify the shared library using that corrupted address.

For example, if `ipc_tool` found memory corruption at address `0x30000000`, view the ASCII text at that address.

Enter **`ipc_tool -d`** then, at the prompt, enter the address of the corrupted memory. If the ASCII text for this address appears to be associated with a WorkFlo queue, you could conclude that a WorkFlo shared library was responsible for the problem and take appropriate action.

## 5. System Management Semaphores

(corresponds to the -s flag)

Displays the status of semaphores used to manage various critical sections within the shared memory and interlock managers. Output consists of two sections. The first section lists information on system semaphores. The second section lists the status of current shared libraries (abstracts).

The following example shows a partial listing of the report displayed when selecting this option on a UNIX platform. The report displayed for your Image Services system might vary from this example. This example omits the complete data where indicated with an ellipsis (...).

Descriptions of each field follow the example.

System Sem	Current value	Waiting procs	Last op pid	Current state
Shm_Init	1	0	17551	(running)
Shm_Mod	1	0	10953	(running)
Sem_Abst	1	0	17540	(defunct)
Sem_Log	1	0	10953	(running)
Sem_Mod	1	0	17540	(defunct)
Sem_TAS	1	0	17540	(defunct)
Abstract Id	Current value	Waiting procs	Last op pid	Current state
SHM	1	0	8049	(defunct)
ARC	1	0	8049	(defunct)
ARI	1	0	8049	(defunct)
ARl	1	0	8049	(defunct)
ARlr	1	0	8049	(defunct)
ARM	1	0	8643	(running)
BES	1	0	13253	(running)
BESl	1	0	13253	(running)
BESr	1	0	8049	(defunct)
BKG	0	0	8643	(running) <- CLAIMED
BRB	1	0	8049	(defunct)
BRBt	1	0	8049	(defunct)
...				

### Current value

Indicates whether the semaphore is set. A value of 1 indicates the semaphore is not set; 0 indicates the semaphore is set.

### **Waiting Procs**

Number of processes waiting to access the shared library. If any processes are waiting, the current value field is 0.

### **Last op pid**

Identifier of the last process to modify the semaphore.

### **Current state**

Indicates whether the shared library is running or not running (de-funct). If the interlock is in the process of being claimed (a process which lasts for only a fraction of a second), then a “CLAIMED” message appears in the right hand column. In the System Management Semaphores example, the BKG shared library (abstract) is being claimed by process 8643.

## **6. Claimed Application Interlocks**

(corresponds to flag -i)

Reports on the interlocks currently claimed by FileNet software. Use this option to identify processes that could be creating a bottleneck in the system.

The output report prints the process name with the process ID. In addition, the display encloses the named interlock string in double quotes (" ").

The following example shows a partial listing of the report displayed when selecting this option on a UNIX platform. The report displayed for your Image Services system might vary from this example. This example omits the complete data where indicated with an ellipsis (...).

---

```
Interlock [8] created by COR ""
    Claim counts (initial: 0 current: 0 shared: 0)
    Last claimer: (32204) [dead]
Interlock [18] created by MKFt "MKF 0 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0)
    Claim queue (oldest first)
        [1] MKF_writer (19574) [alive]
Interlock [44] created by MKFt "MKF 1 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0)
    Last claimer: MKF_writer (34732) [alive]
    Claim queue (oldest first)
        [1] MKF_writer (34732) [alive]
Interlock [70] created by MKFt "MKF 2 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0)
    Last claimer: MKF_writer (22968) [alive]
    Claim queue (oldest first)
        [1] MKF_writer (22968) [alive]
Interlock [88] created by MKFt "MKF 2 accttr 11"
    Claim counts (initial: 0 current: 0 shared: 0)
    Last claimer: bes_commit (16600) [alive]
Interlock [89] created by MKFt "MKF 2 accttr 12"
    Claim counts (initial: 0 current: 0 shared: 0)
    Last claimer: dtp (28432) [alive]
Interlock [96] created by MKFt "MKF 3 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0)
    Claim queue (oldest first)
        [1] MKF_writer (27588) [alive]
...
Summary information...
383 total interlocks; 322 in use; 61 available.
```

---

For shared libraries that do not yet include the named interlock string, ipc\_tool prints an empty pair of double quotes. If processes are waiting on an interlock, the process name and number appear in the claim or share queue.

In the example below, interlocks 70, 76, and 227 are claimed. Processes 24474 and 19554 are waiting in the claim queue for interlock 76. Process 18746 is currently claiming interlock 76.

```
Interlock [70] created by MKFt "MKF 2 writeahead ilk"
    Claim counts (initial: 0 current: -1 shared: 0)
    Last claimer: MKF_writer (15122) [alive]
    Claim queue (oldest first)
        [1] MKF_writer (15122) [alive]
Interlock [76] created by MKFt "MKF 2 transres"
    Claim counts (initial: 0 current: -2 shared: 0)
    Last claimer: bes_commit (18746) [alive]
    Claim queue (oldest first)
        [1] bes_commit (24474) [alive]
        [2] PRI_daemon (19554) [alive]
Binary interlock [227] created by CSML ""
    Claim counts (initial: 1 current: 0 shared: 2)
    2 processes holding claimshare status:
        [1] bes_commit (24474) [alive]
        [2] dvt_bs_2 (40892) [alive]
```

Some interlocks have names in order to better identify them. The display encloses the named interlock string in double quotes (" "). In the example above, Interlock 70 (MKF 2 writeahead ilk) and 76 (MKF 2 transres) are named interlocks but 227 is not. Interlock 227 is shared by processes 24474 and 40892 (although neither has currently claimed the interlock).

If a number of processes build up in a queue waiting for an interlock, a bottleneck has probably occurred. The process that last claimed the interlock is probably responsible for the bottleneck.

---

## 7. Interlock usage of a process

(corresponds to flag -I)

**Note** The flag for menu option 7, Interlock Usage of a Process, is an uppercase i. Do not confuse this with either the lowercase i flag for menu option 6, Claimed Application Interlocks, or a lowercase L.

Prompts you for a FileNet process number then displays current interlock use for that process. Use this option when you suspect a particular process is hung on an interlock.

For example, when we request information on interlock use for process 19310, ipc\_tool reports that process 19310 is waiting on interlock 146 for write access (wait\_type: writer):

```
Process 19310 (alive); waiting on interlock [146]; wait_type: writer
Interlock [146] created by program;
Claim queue (oldest first)
*** [1] 19310 (alive)
```



## 8. Current Processes

(corresponds to flag -p)

Calls the `whatsup` program to display a list of current FileNet processes as well as any shared memory or semaphore sets allocated for FileNet programs. Use Current Processes to determine the last process that modified a given interlock. (See the description of [“`whatsup`” on page 1309](#) for additional information.)

### UNIX

The following is a partial listing of output from the `-p` option. The output is slightly different when `-p` is run on a Windows Server platforms. This sample is from a UNIX platform:

---

IS Processes on server 'moorea':

User	PID	PPID	Start Time	Processes
fns	30692	26282	13:43:33	bes_commit -f 2 Bes1:moorea:FileNet
fns	21802	30692	13:43:42	bes_commit -f 1 Bes1:moorea:FileNet
fns	42686	44220	13:56:40	OCOR_Listen CWH 8003414
fns	41366	33684	13:56:23	OCOR_Listen CWH 2
fns	26078	27612	13:56:41	OCOR_Listen CWH 8003402
fns	23526	23780	15:13:11	OCOR_Listen CWH 8003401
fns	30188	43752	13:43:33	OCOR_Listen master
fns	43490	26282	13:43:32	CSM_daemon
fns	42006	20444	13:43:39	del_commit
fns	22760	23780	15:13:11	DOCs
fns	38428	20444	13:43:41	dsched a
fns	20444	26282	13:43:32	ds_init
fns	37898	20444	13:43:38	ds_notify
fns	30220	20444	13:43:38	ds_notify
fns	22286	20444	13:43:39	ds_notify

---

fnsw	39688	20444	13:43:38	ds_notify
fnsw	27664	20444	13:43:39	ds_notify
fnsw	27936	20444	13:43:41	dtp a 2
fnsw	40222	20444	13:43:41	dtp a 1
fnsw	34066	20444	13:43:40	dtp_tran
fnsw	35608	20444	13:43:40	fbcommit
fnsw	33532	31992	13:43:34	fn_snmpd
fnsw	21498	26282	13:43:34	fn_trapd
fnsw	43760	29648	13:43:33	gti
fnsw	36012	26282	13:43:09	/fnsw/bin/ilk_daemon
fnsw	40666	34518	13:43:31	INXbg
fnsw	25312	27612	13:56:42	INXs
fnsw	24542	32472	13:43:31	INXu
fnsw	40726	32514	17:11:34	ipc_tool -p
fnsw	28846	26282	13:43:09	MKF_clean
fnsw	37060	29632	13:43:30	MKF_writer 3
fnsw	37808	29612	13:43:28	MKF_writer 2
fnsw	32924	29592	13:43:26	MKF_writer 1
fnsw	29360	26282	13:43:09	MKF_writer 0
fnsw	43928	33684	13:56:23	NCHs
fnsw	36764	32666	13:56:23	NCHs
fnsw	28626	26282	13:43:30	NCH_daemon -pt
fnsw	39150	30188	13:43:33	OCOR_Listen -pt -s32769 -t3600 -d20
fnsw	21524	20444	13:43:40	osi_migrate
fnsw	39172	26112	13:43:37	PRI_check
fnsw	26112	43238	13:43:36	PRI_daemon
fnsw	25346	26112	13:43:37	PRI_notify
fnsw	28166	26112	13:43:37	PRI_worker
fnsw	39670	41202	13:43:33	QLG_start
fnsw	20506	20444	13:43:41	rmt_commit
fnsw	41920	44220	13:56:40	SECS
fnsw	25798	26052	13:56:40	SECS
fnsw	37588	26282	13:43:31	SEC_daemon
fnsw	31256	1	11/29/99	TM_daemon -s

(END)

---

**WIN**

The following sample is from a Windows Server platforms. Note the difference in format:

```

ipc_tool
-> 8
FileNet Processes on server 'jupiter':
User      PID      TID      Start Time  Processes
Administrator0xfe 0xfc 03:27:47 PM ipc_tool

SYSTEM 0xc4 0xc3 12/04/1996 bes_commit 2 Bes1:jupiter:FileNet
SYSTEM 0xdf 0xde 12/04/1996 bes_commit 1 Bes1:jupiter:FileNet
SYSTEM 0xb4 0xbe 12/04/1996 COR_Listen -pt -d20 -t3600
SYSTEM 0xd2 0xd1 12/04/1996 CSM_daemon
SYSTEM 0xd6 0xd5 12/04/1996 del_commit
SYSTEM 0xda 0xd9 12/04/1996 dsched a
SYSTEM 0xc2 0xc1 12/04/1996 ds_init
SYSTEM 0xca 0xc9 12/04/1996 ds_notify
SYSTEM 0xcc 0xcb 12/04/1996 ds_notify
SYSTEM 0xdc 0xdb 12/04/1996 dtp a 1
SYSTEM 0xd0 0xcf 12/04/1996 dtp_tran
SYSTEM 0xce 0xcd 12/04/1996 fn_snmpd.exe
SYSTEM 0xa7 0x66 12/04/1996 fn_trapd.exe
SYSTEM 0xaf 0x7b 12/04/1996 INKbg -s IndexServer
SYSTEM 0xc0 0xbf 12/04/1996 INKu
SYSTEM 0xa4 0xad 12/04/1996 ipc_daemon
SYSTEM 0xab 0x3a 12/04/1996 MKF_clean
SYSTEM 0xb3 0xb2 12/04/1996 MKF_writer 1
SYSTEM 0xbd 0xbc 12/04/1996 MKF_writer 2
SYSTEM 0xaa 0x77 12/04/1996 MKF_writer 0
SYSTEM 0xbb 0xba 12/04/1996 MKF_writer 3
SYSTEM 0x4a 0x3f 12/04/1996 NCH_daemon -pt
SYSTEM 0xd4 0xd3 12/04/1996 osi_migrate
SYSTEM 0x6e 0xa9 12/04/1996 perf_mon -f/fnsw/local/sd/perf_mon.scri
pt -f/fnsw/lib/perf/perf_mon.script
SYSTEM 0xe9 0xe8 12/04/1996 PRI_check
SYSTEM 0xc6 0xc5 12/04/1996 PRI_daemon
SYSTEM 0xe7 0xe6 12/04/1996 PRI_notify
SYSTEM 0xeb 0xea 12/04/1996 PRI_worker
SYSTEM 0xd8 0xd7 12/04/1996 rmt_commit
SYSTEM 0xb1 0xb0 12/04/1996 SEC_daemon
SYSTEM 0xf4 0xa8 12/11/1996 SQLs
SYSTEM 0x4c 0x4b 12/02/1996 D:\FN\SW\bin\tn_daemon.exe

```

## **9. Dump Shared Memory Segment(s) to File(s)**

(corresponds to flag -D)

Dumps the existing shared memory segments to a magnetic disk file or to magnetic tape. Use this option to collect information before reporting a problem or when you want to save information on shared memory corruption before rebooting. The ipc\_tool prompts you for the destination of the information to be dumped.

## A. Shared Memory Segment Information

(corresponds to flag -A)

Displays the number of shared memory segments in use. Its output includes descriptions of the information provided. The output of option A varies, based on platform.

The AIX and SUN platforms include shared memory segment addresses. Output from option A on either an AIX or SUN platform shows the addresses at which the system will allocate segments. This table is not fixed — the system determines addresses through testing the system shared memory limits.

The following example shows output from option A run on an AIX server:

---

```
IS software shared memory segment limit: 11 segments
Current configured segment size: 0x01000000 bytes (16 MB)
```

```
Before allocating shared memory for IS, the SysV library performs a
test to determine the system shared memory limit. This test can be
used as a reference for performance tuning. The test results vary
depending on the amount of memory in use by other processes. The actual
amount of shared memory available during operation may be less. The
test results are:
```

```
    Successfully attached to 10 segments
    Successfully obtained 160 MB of shared memory
```

```
The following table displays the number of shared memory segments currently
in use by IS. Segment #0 (called the address manager) is small. The
The other segment(s) contain the actual IS data. Note that running
ipc_tool will force the creation of segments #0 and #1 even when no other
IS process is up.
```

## Shared Memory Address Manager Information

Address	Shm id	Creator
0 0xc0000000	4098	Shared address manager
1 0xb0000000	16385	FileNet server software

Total IS shared memory allocated: 16 MB (not including segment #0)

The following table lists the fixed addresses used for the shared memory segments.

System's fixed shared memory address table (derived)...

0:	0xc0000000
1:	0xb0000000
2:	0xa0000000
3:	0x90000000
4:	0x80000000
5:	0x70000000
6:	0x60000000
7:	0x50000000
8:	0x40000000
9:	0x30000000

---

Output from option A on either an HPUX or SUN platform does not show addresses at which segments will be allocated. The HPUX platform does not include shared memory segment addresses. Windows uses fixed addresses which are the same on every Windows-based Image Services server; therefore, they are not displayed with this option.

The following example shows output from option A run on an Windows server.

---

```
IS software shared memory segment limit: 129 segments
Current configured segment size: 0x01400000 bytes (20 MB)
```

Before allocating shared memory for IS, the SysV library performs a test to determine the system shared memory limit. This test can be used as a reference for performance tuning. The test results vary depending on the amount of memory in use by other processes. The actual amount of shared memory available during operation may be less. The test results are:

```
    Successfully attached to 23 segments
    Successfully obtained 460 MB of shared memory
```

The following table displays the number of shared memory segments currently in use by IS. Segment #0 (called the address manager) is small. The other segment(s) contain the actual IS data. Note that running ipc\_tool will force the creation of segments #0 and #1 even when no other IS process is up.

#### Shared Memory Address Manager Information

	Address	Shm id	Creator
0	0x45000000	FNSHM_464d0000	Shared address manager
1	0x46400000	FNSHM_464a0000	FileNet server software

```
Total IS shared memory allocated: 20 MB (not including segment #0)
```

---

## Checklist

Before you use ipc\_tool, be aware of the following:

- FileNet software must be running to detect shared memory use.
- In rare cases, the semaphore that initializes shared memory could be locked. In this case, ipc\_tool cannot operate since no shared memory is available to view. When ipc\_tool cannot claim the semaphores it requires, after about 10 seconds, the system prompts:

Can't claim semaphores. Continue anyway? (y/n)[n]:

This indicates a problem with system operation: most likely an Image Services software hang. In response to this prompt, enter a **y** to enable ipc\_tool to access Image Services information without claiming any semaphores.

The system uses semaphores to prevent access to the Image Services information while it is being changed. When ipc\_tool avoids claiming the semaphores, the information it obtains could be in a state of change, or it could be corrupted, making it invalid.



## Procedure

The following procedure is an example used to diagnose Image Services system software hangs.

- 1** Enter **ipc\_tool** at the command line.
- 2** Select menu option 1, Abstract Shared Memory (detailed).
  - If a data corruption message displays, make a note of the shared library (abstract) and location of the corruption (the shared memory address). You need this information to resolve data corruption on a shared library.
  - If a data corruption message does not appear, the problem could be related to system management semaphores. Select menu option 5, System Management Semaphores.
- 3** If you know the location of the corruption (the shared memory address), select menu options 3 and 4, entering the shared memory address at the prompts.
- 4** Before rebooting, select menu option 9 to dump shared memory information to a file. This information could be useful later in identifying the cause of the problem.

## Related Topics

### [“whatsup” on page 1309](#)

See the UNIX operating system’s documentation or the online manual (“man”) pages on a UNIX server for information about the ipcs and ipcrm tools.

- The ipcs tool displays information on message queues, shared memory segments, and semaphores. You can use this tool after shutting down all Image Services software to determine if any fnsw resources still remain on the system.
- The ipcrm tool removes a message queue, semaphore set, or shared memory ID. You can use this tool to forcefully remove any interprocess communication facilities that are still in use after shutting down the Image Services software.

### **CAUTION**

---

Do **not** use ipcrm while the Image Services software is running! Fatal and unpredictable errors could occur if interprocess communication facilities are removed while the Image Services software is active.

---

## ixdb\_stat

### Description

The ixdb\_stat tool gathers statistical information about documents and folders from the index database. It also gathers statistics on WorkFlo queues. Types of information gathered include, but are not limited to:

- Total number of documents and folders
- Smallest and largest document number
- Number of closed and filed documents
- Document class indexes

Output from the tool is written to a file you specify when you start the tool or to a default output file if you do not specify one.

### Use

Use ixdb\_stat to gather statistical information about the index database. For example, you can determine the number of documents or folders in your database, or the number of closed documents. To examine how the space in the index database is being used, see [“spacerpt” on page 1203](#).

Specify an output file of your choice into which ixdb\_stat writes its collected statistics, or optionally use a default output file. To view an output file, use a text editor or text viewing commands such as the UNIX **more** or **less** commands or the Windows Server **type** command. In Windows Server, you can also use Wordpad.

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3)

The output of the ixdb\_stat tool could report some confusing information when being used with the FileNet P8 Content Federation Services functionality. Since the function of ixdb\_stat is to gather statistical information about documents and folders from the index database, some documents stored on Image Services through FileNet P8 Content Federation Services might not have entries in the Index Database, so the output is going to indicate fewer documents than their actually are.

Refer to the table below to determine when documents on the IS server will not have the index information associated with them (the **Ns**).

Document Activity	Index on CE	Index on IS
New documents entered via Content Engine	Y	N
New documents entered via Image Services		
• Indexed on both IS and C	Y	Y
• Indexed only on CE	Y	N
Existing document images transferred to CE		
• Indexed on both IS and CE	Y	Y
• Index deleted from IS	Y	N

## Syntax

If you run `ixdb_stat` with no options, online help displays. The `ixdb_stat` syntax is the following:

```
ixdb_stat [-d] [-f] [-a] [-v] [fid]
```

- d** Gather only document statistics
- f** Gather only folder statistics
- a** Shorthand notation for the combination of `-d` and `-f`. If you do not specify either `-d` or `-f`, the program uses `-a` by default.
- v** Verbose option continuously prints the number of database rows examined. It prints the number for every 10000 rows retrieved.
- fid** User-specified name of the output file. Default files are:

`/fnsw/local/logs/inx_logs/ixdbs_<MonDD>`for UNIX platforms

`D:\fnsw_loc\logs\inx_logs\ixdbs_<MonDD>`for Windows Server platforms

where `MonDD` is the current month and day.

## Sample Output

### UNIX

This UNIX sample output is from ixdb\_stat with the -d option. You can use the less command to display document statistics written to the default output file.

The output below is a partial listing of the file contents:

---

```
corona(root)> less /fnsw/local/logs/inx_logs/ixdbs_Oct28

                INDEX DATABASE STATISTICS

Date of Report : Mon Oct 28 14:38:55 1996

DOCUMENT STATISTICS :
Total Document : 9460   lowest doc_id : 100100   highest doc_id : 528572
Documents with Archive Dates : 0   Archivable Today * : 0
Documents with Delete Dates : 0   Deletable Today * : 0
Closed Documents : 0

    * unless document is filed in a folder
DATA DICTIONARY STATISTICS :
Total Document classes : 39
Total User Indexes : 122   inverted : 13

Document Class      Total Documents   Index
-----
LM6000_class        0
LM6000_CLASS        210
HP_4X_CLASS         10
HP_2X_CLASS         90   numeric3
                                numeric2
                                name2
HP_650_CLASS        3093
IBM_4X_CLASS         82   NAME
DVT_docclass1       1   DVT_num_idx1
                                DVT_ascii_idx1
```

---

		DVT_date_idx1
		DVT_menu_idx1
DVT_docclass2	3214	DVT_menu_idx2
		DVT_date_idx2
		DVT_ascii_idx2
		DVT_num_idx2
DVT_docclass3	0	DVT_num_cluster
		DVT_ascii_idx1
		DVT_num_idx1
UpDownSQA	0	
DVT_tst_dcl1	0	DVT_menu_idx1
		DVT_date_idx1
		DVT_ascii_idx1
		DVT_num_idx1
DVT_docclass4	0	DVT_ascii_idx1
		DVT_ascii_idx2
		DVT_date_idx1
		DVT_date_idx2
		DVT_indexmenu
		DVT_indexvaltab
		DVT_menu_idx1
		DVT_menu_idx2
		DVT_num_idx1
		DVT_num_idx2
...		

Index	Non-Null Values	Number of Document Classes
-----	-----	-----
NAME	82	1
numeric1	0	0
numeric2	78	1
name2	0	1
numeric3	78	1
index1	0	0
hlee1	0	0
hlee2	0	0
hlee3	0	0
hlee4	0	0
h	0	0
DVT_num_idx1	1	4
DVT_ascii_idx1	1	4

---

DVT_date_idx1	1	3
DVT_menu_idx1	1	3
DVT_num_idx2	3214	2
DVT_ascii_idx2	3214	2
DVT_date_idx2	3214	2
DVT_menu_idx2	3214	2
DVT_num_cluster	0	1

...

## WORKFLO QUEUES STATISTICS:

-----

Workflo Queue Name	Number of Rows
--------------------	----------------

-----

WQM001Q000002	0
WQM001Q000004	0
WQM001Q000005	0
WQM001Q000006	0
WQM001Q000007	0
WQM001Q000008	0
WQM001Q000009	0
WQM001Q000010	0

continues, ...

WQM001Q000073 2000

Total # of Workflo Queues: 52

NOTE: Removing a document class that has a non-zero number of documents associated with it will cause those documents to be unretrievable. Removing a user index during the conversion process will cause any documents with non-null values for that index to lose those index values forever.

Done Mon Oct 28 14:39:11 1996



## Checklist

Before you use `ixdb_stat`, be aware of the following:

- You must run `ixdb_stat` on the Index server.
- The verbose (`-v`) option could aid in monitoring the progress made on a large database. However, `ixdb_stat` runs slower in verbose mode.

## Procedure

- 1 Enter **`ixdb_stat`** with appropriate options to start the program.
- 2 Examine the output file.

You could need the assistance of your service representative to analyze the output.

## Related Topics

See [“`spacerpt`” on page 1203](#) for information on obtaining index database space statistics.

# kermit

## Description

The kermit tool transfers files between a personal computer and a host system. In addition to operating in interactive mode, the FileNet C-based kermit program is capable of operating in **server** mode, in which kermit transmits and receives files under the direction of a remote (usually personal) computer.

The kermit tool does not change the names of files it sends, with the following exceptions:

- Changes lowercase letters to uppercase
- Removes path names
- Detects each tilde (~) character and changes it to an X character
- If the file name begins with a period, inserts an X before it

The kermit tool stores received (incoming) files under their own names with the following exceptions:

- Changes uppercase letters to lowercase
- If you specified the write protect (-w) option and the file name is the same as that of an existing file, appends a generation number to the name

## Use

Use kermit to transfer files in either interactive mode or server mode.

At the command line, enter **kermit -x** to start kermit in server mode; terminate server mode by entering **kermit -f**.

## Syntax

```
kermit [ <files> | - ] [ options ]
```

- <files>** Specifies one or more names of files you want to send or receive. Mutually exclusive with the - option.
- Causes kermit to send files from standard input, which must come from a file or a pipe. Standard input cannot come from the terminal. Mutually exclusive with the <files> option.

## Options

- s** Sends the specified files
- r** Receives files  
  
This option causes kermit to wait passively for files to arrive.
- k** Passively receives files and sends the files to standard output
- f** Sends a **finish** command to the remote server to terminate server mode
- a** Specifies an alternate name for a single transferred file  
  
Use with the -s, -r, or -g options to tell the receiving system to use the alternate name specified when storing the file. If you send or receive

---

multiple files, this option affects only the first file. The following example sends the file called “main” to be stored as “**main2**” on the receiving system:

**kermit -s main -a main2**

**-x** Starts kermit in server mode

The result is the same as entering the **server** command when kermit is running interactively.

**-l <line>** Specifies the terminal line to use for terminal connection and file transfer. The following example uses line cu10 for the transmission:

**kermit -l /dev/cu10**

**-j<host>** Specifies the network host name

**-b <bps>** Specifies the baud rate <bps> for the line <line> named in the -l option

Always include this option when using the -l option. The following example uses line cu10 and a line speed of 9600 baud for the transmission:

**kermit -l /dev/cu10 -b 9600**

**-p <x>** Specifies the type of parity, where <x> is one of the following:

e	even
o	odd
m	mark
s	space
n	none (the default)

- 
- t** Specifies half-duplex, line turnaround with XON as the handshake character
  
  - g <remote\_file>** Actively requests a remote server to send the named files  
  
<remote\_file> is a file specification in the remote host's own syntax. Ensure that you enclose in quotes those characters in <remote\_file> that would otherwise be replaced by a UNIX shell.
  
  - c** Establishes a terminal connection over the specified or default communication line **before** any file transfer takes place  
  
Use with the **-l** and **-b** options.
- 
- Tip** To return to the local system, type the escape character—normally Control+backslash or **^\**—followed by the letter **c**.
- 
- n** Similar to **-c** option but this option is performed **after** a file transfer takes place. You can use the **-c** and **-n** options in the same command. Use with the **-l** and **-b** options.
  
  - i** Specifies transferring files exactly **as is** with no conversions (a binary transfer)  
  
This option eliminates the <LF> to <CR><LF> conversion which normally occurs when transferring files between UNIX and non-UNIX systems.
  
  - w** Specifies write-protection to avoid file name conflicts with sent files that have the same name as existing files on the receiving system  
  
Use this option to prevent existing files from being inadvertently overwritten.

- q** Suppresses screen update (sets **quiet** mode) during file transfer  
File transfer proceeds in the background.
- d** Sets debug mode and records information to the debug.log file in the current directory  
  
Use this option to collect information when you encounter problems when running the kermit program.
- e <n>** Sets the receive packet length to <n>
- m<name>** Sets the modem type <name>
- v<n>** Sets the window size to <n>
- z** Forces operation of kermit to the foreground
- h** Displays kermit command line options
- S** Enters interactive dialog mode  
Default if no action command is given on the command line.

## Checklist

Before you use kermit, be aware of the following:

- You should use the write protect (**-w**) option to prevent overwriting files on the receiving system that could have the same name as a received file from another system.
- When you specify the **send** – command, kermit sends from standard input. Standard input must be from a file or a pipe; it cannot be directly from a terminal.

- Depending on how your shell interprets input parameters, you might need to enclose certain parameters, such as remote file names, in quotes.

## Procedure

No specific procedure is required.

## Related Topics

**[“dialout” on page 295](#)**

# killfnsw

## Description

The killfnsw tool is issued at the system command line to forcefully shut down components of the FileNet system such as processes, daemons, databases, and users.

---

**CAUTION**

Do not use killfnsw unless you are absolutely sure normal shutdown methods have failed. We recommend that killfnsw be used only by a trained FileNet system administrator or your service representative to avoid damage to your FileNet system.

---

The killfnsw program removes all existing interprocess communication structures including shared memory segments, message queues, and semaphore sets used by the FileNet software. It abruptly shuts down databases.

If you run killfnsw with no arguments, the Task Manager daemon (TM\_daemon) remains active. You must use a killfnsw argument to terminate TM\_daemon.



## Use

Use killfnsw **only** after normal attempts to shut down the software fail and you have evaluated the reason your system cannot shut down normally. For example, verify that the normal shutdown process or command is not just running slowly. (Allow at least five minutes for a shutdown process or command to complete before resorting to killfnsw.) Use ipc\_tool to analyze your system's state. However, if ipc\_tool does not run or cannot complete, you could be forced to use killfnsw to shut down your system or components of your system.

You could be forced to use killfnsw to terminate the TM\_daemon. Normally, the TM\_daemon continues to run across shutdown cycles of FileNet software. On occasion, you might have to terminate TM\_daemon as well. For example, when you need to install fix packs that include shared libraries on the FileNet system, you must terminate all processes that use the shared library before installing the new shared library. Since the TM\_daemon uses some of the FileNet shared libraries, you must terminate it to upgrade the shared library or install the new shared libraries.

---

**Note** For more information about starting and stopping a service process in a Windows Server environment, see the *System Administrator's Companion for Windows Server*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

---

## Syntax

killfnsw [-h] [-D] [-v] [-S] [-A] [-p <pid>] [-y] [-c] [-d] [-n] [-r]

- h** Help: displays usage
- D** Kills TM\_daemon
- v** Runs in verbose mode. This option displays names of processes as they are terminated.
- S** Kills processes when the FileNet system is in single user mode. killfnsw will not kill some IS tasks. These are tasks which need to run while killfnsw is terminating the software. This option alters the list of tasks to ignore while the system is running in single user mode. The ignore list for single user mode is smaller, so more tasks are killed with this option.
- A** Removes address manager shared memory segments. killfnsw always removes the data shared memory segments. This option also removes the address manager segment and should be used with **-D** to completely shut down the IS and remove all its resources. This option only applies to UNIX platforms.
- p<pid>** Kills the process specified by process ID <pid>
- y** Suppresses confirmation messages like skipping the killfnsw startup prompt.
- c** Kills only clean shared libraries with zero use counts (AIX/6000 platforms only)
- d** Debug mode. This option helps debug killfnsw and should only be used at the request of **your service representative** as it can cause problems in the operation of Image Services.

- n Do not claim semaphores. This option normally claims system semaphores before accessing shared memory resources. If those semaphores are already claimed by another process, killfnsw could hang. This option allows killfnsw to ignore the claimed semaphores and continue. This option should only be used when killfnsw hangs and **only** at the request of **your service representative** as it can cause problems in the operation of Image Services.
- r Remove killfnsw guard file. killfnsw creates a **guard** file (named /fnsw/etc/killfnsw) to prevent other IS tasks from starting while killfnsw is terminating Image Services. This option should only be used if killfnsw failed to remove the guard file and **only** at the request of **your service representative** as it can cause problems in the operation of Image Services.

---

**Note** Use **initfnsw stop** or the IS task manager (Xtaskman) to properly shutdown Image Services software. If problems occur during the shutdown, the **killfnsw** command could be used to clean up the system.

After Image Services has stopped, enter **killfnsw -AD** to completely remove Image Services processes and resources.

---

## Checklist

Before you use killfnsw, be aware of the following:

- You should use killfnsw **only** after normal attempts to shut down the software fail and you have evaluated the reason your system cannot shut down normally.
- In a UNIX environment, running the whatsup command restarts a TM\_daemon that you terminated with killfnsw.

## Procedure

1 Use Task Manager to list all running FileNet processes.

2 Terminate FileNet processes.

Use Task Manager or the **initfnsw stop** command.

3 Run killfnsw with appropriate options to terminate active processes.

When killfnsw completes, run whatsup to determine if the processes have been stopped.

---

**Note** If you run killfnsw with no arguments, the Task Manager daemon remains active.

---

4 Terminate the Task Manager daemon, if necessary, enter the following command:

**killfnsw -D**

---

**Tip** Do not run the whatsup command after you terminate TM\_daemon in a UNIX environment. Doing so automatically restarts TM\_daemon.

---

## Related Topics

[\*\*“INX tool” on page 598\*\*](#)

[\*\*“whatsup” on page 1309\*\*](#)

*System Administrator’s Handbook* for Task Manager information.

*System Administrator’s Companion for Windows Server* for information on starting and stopping service processes in a Windows Server environment.

To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21\*\*](#).

## ldap\_exp

### Description

LDAP (Lightweight Directory Access Protocol) is a directory service protocol that runs over TCP/IP and its service model is based on entries. In the area of system security, this would include entries like users, groups, and group members. LDAP provides a way for a client to prove his identity to a directory server, allowing for seamless access to remotely linked resources such as an IS or CS system. The FileNet software has the ability to import users and groups from an LDAP server into the FileNet proprietary security database. That means that LDAP users and groups need to only be administered once, via the LDAP server. LDAP users and groups do not need to be administered a second time using FileNet security administration tools.

There are various LDAP-based directory service products (such as Sun Java System, Novell NDS, and Microsoft Active Directory) that can be used. LDAP is a solution many companies are turning to in order to provide simplified system security centralization and authentication. For companies currently using LDAP who would like to leverage their users and groups as FileNet principals and groups (Unified Logon), this can be done through an export/import process. FileNet Image Services provides a set of tools for this LDAP solution that first exports (using ldap\_exp) the users names, group names, and group memberships from the LDAP-based directory service and then imports them into the Image Services (IS) security database using ldap\_import. Image Services can import from any supported LDAP directory via a standardized XML interchange format. FileNet Image Services currently provides LDAP support for Sun Java System, Novell NDS, and Microsoft Active Directory LDAP directory services.

ldap\_exp exports the LDAP-based security information to an XML formatted file. For more information on this file, go to [“Export File Information” on page 681](#). The tool used to perform the import function is LDAP\_import. To review the LDAP\_import tool go to [“LDAP\\_import” on page 671](#).

---

**Note** Sun Java System was formally known as Sun ONE Directory Server.

---

## Use

It is the use of the ldap\_exp tool and Web Services 3.2 in conjunction with the LDAP\_import tool that makes the single logon from a trusted Windows Server domain feature possible.

---

**Note** It is important to note that in order to use LDAP, you must be connected to a client running Web Services 3.2.

Also, The Image Services security database maintains separate entries for each supported directory service type: FileNet proprietary service database, NT LAN Manager (NTLM), and LDAP. All three security entries can co-exist, but you need to choose one directory service for administering users and groups for all thin clients of a particular Web Server.

---

Run ldap\_exp to do the following:

- Bind to an LDAP directory set up by a Web-based, networked directory server you specify with one of the ldap\_exp options.
- Create an XML output file for requested group names.

- Write an XML file of user/group information that will be used by the LDAP\_import tool. See [“Export File Information” on page 681](#) for an example of this file.
- Close the LDAP directory service.

## Requirements

There are two main requirements that must be met in order for IS support for LDAP to work:

- Communication with a client running Web Services 3.2
- Login and password to an LDAP Directory Service

## Syntax

The command for exporting the XML file:

```
ldap_exp -s <server> -d <bindDN> -w <bindpw> -t <ldap server type>
-b <baseDN> [-p <port>] [-g <group-list-filename>]
[-o <XML-output-filename>] [-l <logfile>] [-i] [-e] [-k] [-Z] [-f]
[-x <group-list-filename>] [-v] [-c1 <user object class>]
[-c2 <group object class>] [-a1 <user attribute>]
[-a2 <group attribute>] [-a3 <member attribute>]
```

---

**Note** Options are **not** case sensitive. The command switches must be prefixed by a ‘-’ on UNIX platforms and “/” on Windows Server platforms.

---

**? or h** Help. This will bring up the help screen.

**s <server>** (Required) The name of the server where the LDAP directory is installed.  
(for example, **-s sun03**)



- 
- d <bindDN>** (Required) Distinguished login name for the binding to the LDAP directory.  
(for example, **-d "cn=Directory manager"**)
- w "<bindPW>"** (Required) Encrypted password for the distinguished login name.  
(for example, **-w "7kml/6\*:/"**)  
(all passwords should be 8 characters or greater)
- t <ldap server type>** (Required) Server type for the LDAP server.  
(for example, **sun** for Sun Java System, **nov** for Novell NDS, **msft** for Microsoft Active Directory)
- b "<baseDN>"** (Required) Base distinguished name for the part of the LDAP directory held on the local server.  
(for example, **-b "dc=idmds, dc=com"**)
- p <port>** (Optional) Port for the server where the LDAP directory is installed.  
(for example, **-p 389**)
- g <group-list-file-name>** (Optional) Pathname of input file containing line-feed terminated group names.  
(for example, **-g /fnsw/ldap/groupin.txt**)
- o <XML-output-filename>** (Optional) Pathname of the output XML file. The file must **not** be on a mapped drive.  
(for example, **-o /fnsw/ldap/outputfile.xml**)
- l <logfile name>** (Optional) Pathname of the log file.  
(for example, **-l /fnsw/ldap/exportlog.log**)
- i** (Optional) Check for invalid special characters on <id> tags only. All other tags are ignored.

- e** (Optional) Remove all escape characters: dash (-) and backslash (**\**). Microsoft Active Directory prefixes a backslash character (**\**) with specific characters such as a comma (,) or a backslash (**\**). Consequently, if you use a comma (,) or backslash (**\**) character in your attribute data (any data between tags), you should use this option to remove them.
- k** (Optional) Use paged results for searching if a large amount of data needs to be returned. Option available only when using Windows Active Directory. Using Microsoft Active Directory, a single search call cannot return a large amount of data. If bulk data needs to be returned to the caller, use this option to return the data a page at a time.
- Z** (Optional) Use secure socket layer (SSL) for connecting. The LDAP server must have the secured LDAP socket enabled. For instructions on enabling the secured LDAP socket, consult your LDAP directory service's documentation. Note that this option must be run with the **f** option.
- f** (Optional) This option points to the cert7.db (client certificate) file's directory path. Option available only on UNIX platforms. For instructions on generation the cert7.db file, consult your LDAP directory service's documentation or the Netscape Communicator Guide. The FileNet software does not support cert7.db generation. Note that this option must be run with the **Z** option.
- x <group-list-file-name>** (Optional) Pathname of output file containing group names.  
(for example, **-x /fnsw/ldap/groupout.txt**)
- v** (Optional) Generate verbose log output.
- c1 <user object class>** (Optional) Class type for user. Default is **person** for all platforms.

- c2 <group object class>** (Optional) Class type for group. Default is **groupofuniqueness** for Sun Java System. Default is **groupofnames** for Novell eDirectory (NDS). Default is **group** for Microsoft Active Directory.
- a1 <user attribute>** (Optional) Attribute to use for user. Default is **uid** for Sun Java System. Default is **cn** for Novell eDirectory (NDS). Default is **userprincipalname** for Microsoft Active Directory.
- a2 <group attribute>** (Optional) Attribute to use for group. Default is **cn** for all platforms.
- a3 <member attribute>** (Optional) Attribute to use for member. Default is **uniquemember** for Sun Java System. Default is **member** for Novell eDirectory (NDS) and Microsoft Active Directory.

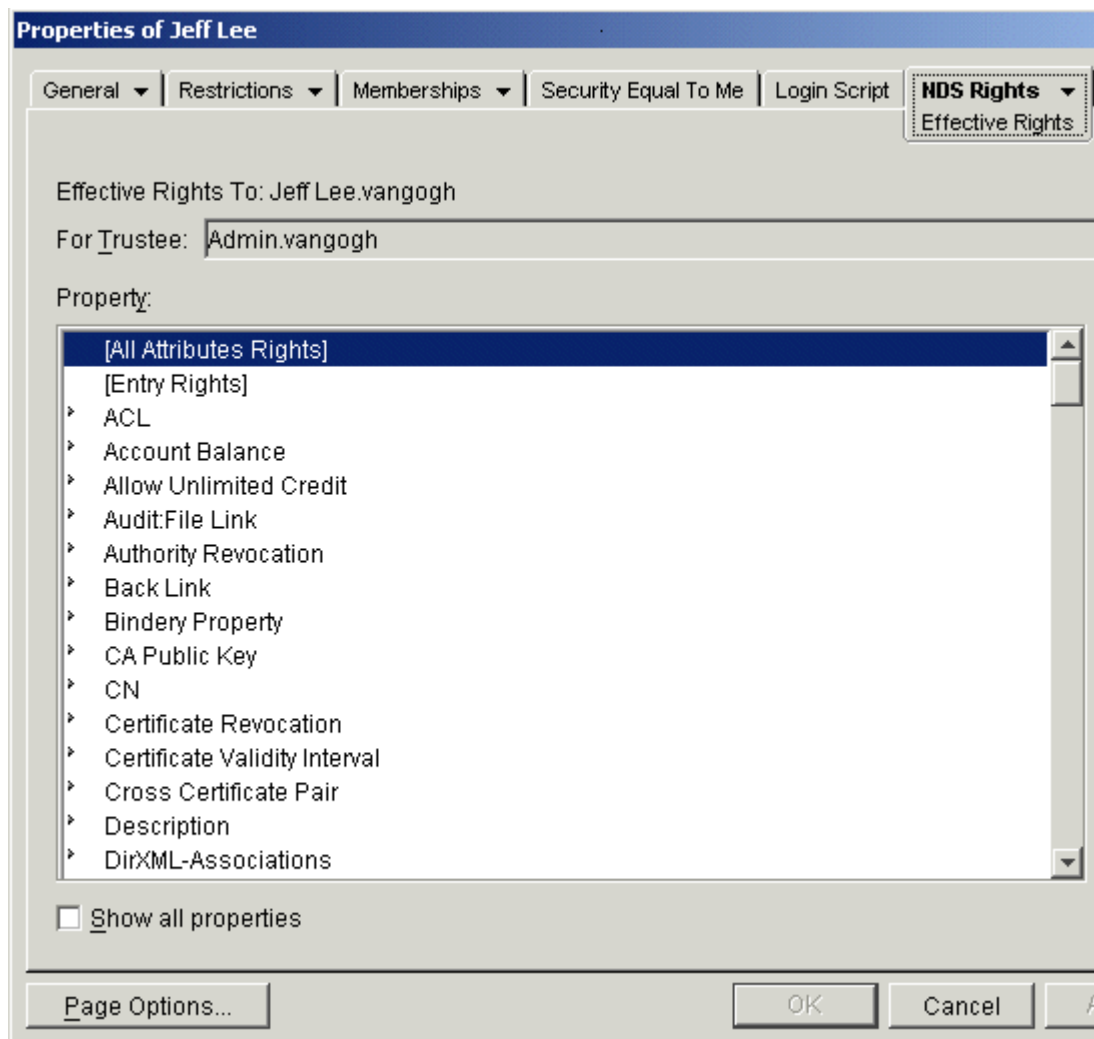
## Procedure

- 1 Obtain the following information in order to successfully run the ldap\_exp command:
  - Determine the server name and port number of the LDAP directory service you will be using along with a login name and password to access the service.
  - Decide which directory services you will be using (**-t option**).
  - Create an ASCII text file with the names of the groups you want to export (**-g option**).
  - Decide where you want the XML output file to go on the IS server (for example, /fnsw/local/tmp on UNIX) (**-o option**). Default = current directory.
  - Decide where you want the ldap\_exp log file to go on the IS server (for example, /fnsw/local/logs/ldap on UNIX) (**-l option**). Default = current directory.
- 2 Log on to the IS server or be authenticated through remote access.
- 3 The LDAP\_import and export utilities are located in the /fnsw/bin directory.

---

**Note** It is important to note that when using the NDS directory service, the binding person (Jeff Lee.vangogh in the example on the next page) needs to have the Common Name (**CN**) property listed in their NDS Rights property list. The user must have rights to the CN field.

---



**4** Review the ldap\_exp options **before** running the export command.

- 5 Export the LDAP-based security information to the XML formatted file.

When ready, use the following command (UNIX example) to export the groups and users to the XML formatted file:

```
ldap_exp -s <server> -b <bindDN> -w <bindpw>  
-t <ldap server type> -p <port> -g <group-list-filename>  
-o <XML-output-filename> -l <logfile>  
-x <group-list-filename>
```

Example:

```
ldap_exp -s sun03 -b smith -w :015K+]% -t sun -p 389  
-g /fnsw/local/tmp/groupin.txt -o /fnsw/local/tmp  
-l /fnsw/local/logs/ldap -x groupout.txt
```

## Related Topics

[“LDAP\\_import” on page 671](#) and [“LDAP\\_password” on page 690](#).

# LDAP\_import

## Description

LDAP (Lightweight Directory Access Protocol) is a directory service protocol that runs over TCP/IP and its service model is based on entries. In the area of system security, this would include entries like users, groups, and group members. LDAP provides a way for a client to prove his identity to a directory server, allowing for seamless access to remotely linked resources such as an IS or CS system. The FileNet software has the ability to import users and groups from an LDAP server into the FileNet proprietary security database. That means that LDAP users and groups need to only be administered once, via the LDAP server. LDAP users and groups do not need to be administered a second time using FileNet security administration tools.

There are various LDAP-based directory service products (such as Sun Java System, Novell NDS, and Microsoft Active Directory) that can be used. LDAP is a solution many companies are turning to in order to provide simplified system security centralization and authentication. For companies currently using LDAP who would like to leverage their users and groups as FileNet principals and groups (Unified Logon), this can be done through an export/import process. FileNet Image Services provides a set of tools for this LDAP solution that first exports (using `ldap_exp`) the users names, group names, and group memberships from the LDAP-based directory service and then imports them into the Image Services (IS) security database using `LDAP_import`. Image Services can import from any supported LDAP directory via a standardized XML interchange format. The FileNet software currently provides LDAP support for Sun Java System, Novell NDS, and Microsoft Active Directory LDAP directory services.

The step that needs to be completed before using the LDAP\_import tool is exporting the LDAP-based security information to an XML formatted file. For more information on this file, go to [“Export File Information” on page 681](#). The tool used to perform this function is ldap\_exp. To review the ldap\_exp tool go to [“ldap\\_exp” on page 662](#).

---

**Note** Sun Java System was formally known as Sun ONE Directory Server.

---

## Use

The use of the LDAP\_import tool in conjunction with the ldap\_exp tool and Web Services makes the security centralization and authentication solution possible.

The LDAP\_import tool and export utilities are located in the /fnsw/bin directory.

---

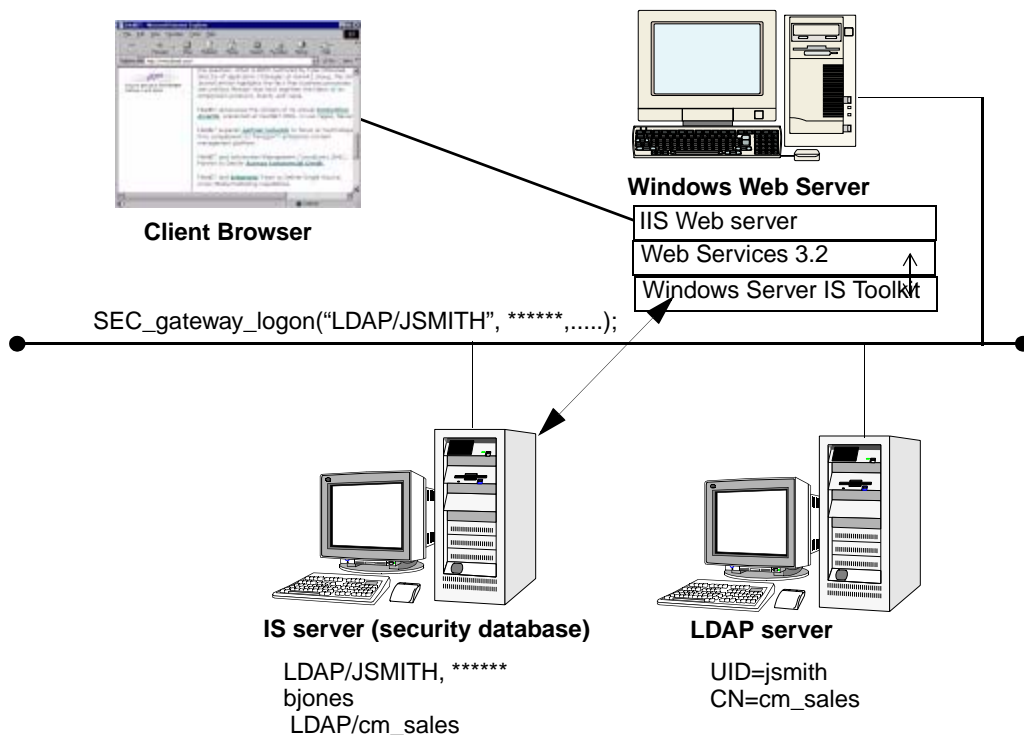
**Note** To use LDAP and the Unified Logon functionality, you must be connected to a client running Web Services 3.2.

Also, The Image Services security database maintains separate entries for each supported directory service type: FileNet proprietary service database, NT LAN Manager (NTLM), and LDAP. All three security entries can co-exist, but you need to choose one directory service for administering users and groups for all thin clients of a particular Web Server.

---



The graphic below depicts what is needed to use LDAP\_import.



Run LDAP\_import to import the LDAP-based security information into the IS Security Database. This requires the use of an XML formatted file generated by the export tool mentioned above and must be run on a Server running Image Services software. This tool can import the XML file information to any other IS server (Windows Server or UNIX) on the network. All import users/groups will be imported with upper-case names and an "LDAP/" prefix will be added denoting that the user's security administration will be authenticated via the LDAP server.

The main operation of LDAP\_import is the sync operation and it happens automatically (by default). The import will make the subset of the users and groups in the FileNet security database that are LDAP users and groups exactly match the LDAP users and groups in the export data file. However, only the users and groups that are from the LDAP source will be affected by the import program. In other words, all other users and groups in the FileNet security database will be unaffected by LDAP\_import. Also, any LDAP-related memberships NOT found in the export data file will be ignored. For example, if LDAP/mjones belongs to native group ACCT this membership will be kept. However, if native user jsmith belongs to LDAP group LDAP/eng, this membership will be ignored.

### **Document Access Use Example**

If an LDAP system has GroupA with UserA and UserB as its members and that group with its users is sent through the ldap\_exp/LDAP\_import process, it will now reside on that IS server as LDAP/GROUPA with its users LDAP/USERA and LDAP/USERB.

An IS server will typically have existing users and groups before importing and they are called Native users and groups. A Native group or user is created by the IS Security Administration application. After the LDAP\_import, by default, users only have access to permissions assigned to (ANYONE). If you want that group and its users to be able to have the same security as an object already residing on the IS server (a native group or user), you will need to go into the security administration application in Xapex and edit each user's permissions to give them the desired document access.

For more information on editing user security, go to the Security Administration chapter of the *System Administrator's Handbook*. To download IBM FileNet documentation from the IBM support page, see

**“Accessing IBM FileNet Documentation” on page 21.** Be aware that the next time an ldap\_exp/LDAP\_import is done, the LDAP\_import tool, by default, will **ignore** any existing membership where one LDAP object (can be a user or a group) is a member of a native group.

## Requirements

There are three main requirements that must be met in order for IS support for LDAP to work:

- Communication with a client running Web Services 3.2.
- To run the LDAP\_import program, the operating system user must be a member of the fnadmin group. To login to Image Services with the user (**u**) and password (**p**) options, the user must be either SysAdmin or a user that is a member of the SysAdminG group with all administrative privileges.
- The sort utility provided by the native operating system must be present. On Windows it is named **sort.exe** and on UNIX it is named **sort**.

---

### Important

Do not run LDAP\_import in the **/fnsw** directory. In some cases, an aborted LDAP\_import run (due to lack of disk space) can leave large temporary files behind without cleaning them up. In extreme cases, this could cause Image Services to hang.

---

### The Sort Utility

The sort utility used by LDAP\_import might require large temporary files during operation. LDAP\_import must be run from a directory with adequate available disk space. For example, you might need up to 800 MB for very large security databases in excess of 100,000 users.

The sort utility also uses space in the **/var/tmp** directory and might need up to 100 MB of free space for security databases in excess of 100,000 users.

The sort utility tries to locate sufficient temporary space in the following directories:

- /var/tmp** The default temporary space used during the sort command processing.
- /usr/tmp** If files cannot be created in /var/tmp, the sort utility tries to use space in /usr/tmp.
- /tmp** If files cannot be created in /var/tmp or /usr/tmp, the sort utility uses space in /tmp.

## Syntax

The minimum command for importing the XML Input file:

**WIN**

**LDAP\_import /h<host> /i<file>**

**UNIX**

**LDAP\_import -h<host> -i<file>**

### Note

Options are **not** case sensitive. Also, a / is used before an option on Windows systems and a - is used before an option on UNIX systems.

**?** Help. This will bring up the help screen.

**h<host>[:organization]**

Image Services (IS) domain, and optionally, organization. The default organization is **FileNet**. This is the IS domain you generally want to import your LDAP source to.

- 
- i<file>** LDAP XML input file (XML formatted file created in the export step). This is the name of the XML formatted input file. This file is generated by the export tool.
- u<user>** IS user name. This is an optional parameter and must be used with /p<pwd> to provide batch mode processing. LDAP\_import interactively prompts for user name for IS security authentication. By specifying user name through the command line, prompting will be skipped.
- p”<enpwd>”** Encrypted password. This is an optional parameter and must always be used with /u<user> to provide batch mode processing. LDAP\_import interactively prompts for user password for IS security authentication. By specifying user password through the command line, prompting will be skipped. The encrypted password is created using the LDAP\_password tool. For more information on the encryption tool, go to [“LDAP\\_password” on page 690](#).
- f** Force password update. If your users were created before this option became available, it is strongly recommended that all users should be re-exported and re-imported with this option set. This option forces an update on the password which is recalculated with a stronger algorithm. The new password is calculated on DN (distinguished name, the LDAP server name, and the IP address of the LDAP server. This option should only be used once for all users. If you are importing new users or if all of your users are created after this option became available, you can ignore this option. For more information on using this option, refer to the document *Implementing Enhanced LDAP Security*.
- b** Batch mode processing. This mode is used for incremental inserts. Use this mode if you do not want to be in the sync mode (default mode). Batch mode does not expire users and groups, even if they do not exist in the XML input file.

For example, if you have already successfully imported 100,000 users and would like to import an additional 5 users, you can generate your XML input file with the 5 new users and run `ldap_import` with the `b` option so all existing 100,000 users are not touched. If you import the additional 5 users without using the `b` option, the existing 100,000 users are expired.

## Procedure

- 1 Transfer the output from Step 6 (`/o` or `-o` parameter) of the export procedure ([page 670](#)) to the IS server, for example using FTP. Place the output file in a directory for example `/fnsw/local/tmp` (`\fnsw_loc\tmp` on a Windows Server).
- 2 Go to the directory on the IS server where the XML formatted file was transferred in Step 1 (for example, `/fnsw/local/tmp` on UNIX).
- 3 Review the `LDAP_import` options **before** running the import command.
- 4 Import the LDAP-based security information into the IS Security Service. This step takes the XML formatted input file and updates the Image Services security database.

Use the following command to import the groups and users to the IS system using the XML formatted file created using `ldap_exp`:

```
LDAP_import /h<host>[:org] /l<file> [/u<user>] [/p"<enpwd>"]
```

Examples:

**WIN**

```
LDAP_import /hidm1:filenet /lxml.dat /ujsmith /p":015K+]%%"
```

**UNIX**

```
LDAP_import -hidm1:filenet -lxml.dat -ujsmith -p":015K+]%"
```

---

**Note** On Windows systems, if the encrypted password contains a % or a \, you need to add a second % or \. For example, `/p":015K+]%"` needs to be entered as `/p":015K+]%%"`. A % becomes %% and a \ becomes \\.

---

When you enter the command, if you do not include the user and password arguments, you are prompted for user name and password (not encrypted) with the necessary privileges to complete the import (such as SysAdmin). You then see messages indicating you are importing an Windows Server domain input file and security information.

---

**Note** **Automated (Batch Mode) import:** If you want to set up your system so that an import is automatically run at pre-determined intervals using crontab, be aware of the following:

Set up crontab entry as shown

```
min hour monthday month weekday command
0 14 27 12 4 /home/fnsw/ldapcmd >
                               /home/fnsw/ldap.log
```

Run LDAP\_import as the FileNet software user, such as **fnsw**.

Enter following commands:

```
/home/fnsw/ldapcmd
/fnsw/bin/LDAP_import -hserver -i /home/fnsw/ldapcmd
-uSysAdmin -p":015K+]%"
```

Please note that you must enter the full paths.

---

---

**Note** If you are importing a lot of users (for example, 100,000), the import could take somewhere between 30 - 60 minutes depending upon the speed of the IS server.

---

- 5 To check if the import was successful, run Xapex, log in as Administrator, and check to see if the new groups and users are present. For example, if you imported a group called **support** from the LDAP-based directory service and you know user **johnf** belongs to the group, then after a successful import a new group (**LDAP/SUPPORT**) and a new user (**LDAP/JOHNF**) will have been created. Also note that the maximum user/group name is 35 characters with the LDAP/ prefix. If this maximum is exceeded, the names are rejected. It is important to know that Image Services does not allow duplicate names between users and users, groups and groups, and users and groups. You will see an error in this log file if you have duplicate names. For example, you cannot have a user named scan and a group named scan. If you do, you will receive the following message:

```
<92,0,108> A duplicate object already exists in the database.
```

## LDAP\_import generated Files

A log file and three ascii files are created or appended per each LDAP\_import run:

- A separate log file is created daily and then appended for each run of the tool during the day. Depending upon platform, this file is located in the following location:

**WIN**

- \fnsw\_loc\logs\ldap\ldap\_importyyyymmddlog.txt

**UNIX**

- /fnsw/local/logs/ldap/ldap\_importyyyymmddlog



- In the directory where the tool is run, three ascii files are created each time the tool is run. These files will precisely list the current FileNet IS LDAP-based users and groups and their memberships:

ASCII File Name	Contents
LDAP_import_yyyymmddhhmmss.usr	All unique users

**Example:**

```
id: LDAP/BJONES      dn: CN=Bob Jones,DC=costa_mesa,0=FileNet  desc: Good guy
id: LDAP/CSMITH     dn: UID=csmith,DC=costa_mesa,0=FileNet    desc:
```

LDAP_import_yyyymmddhhmmss.grp	All unique groups
--------------------------------	-------------------

**Example:**

```
id: LDAP/CM_SALES  dn: CN=cm_mktg,DC=costa_mesa,0=FileNet  desc: Top team
```

LDAP_import_yyyymmddhhmmss.mem	All unique group memberships
--------------------------------	------------------------------

**Example:**

```
id1: LDAP/CM_SALES  id2: LDAP/BJONES  dn1: CN=cm_mktg,DC=costa_mesa,0=FileNet  dn2: CN=Bob Jones,DC=costa_mesa,0=FileNet  type: user
```

## Export File Information

This section describes the XML specification for the export data files from LDAP databases. This ASCII export file format for FileNet security database synchronization simplifies importing logon information from LDAP databases into FileNet security databases. An example of the file is shown next.

### Sample Export File

The following is a sample XML export file with information for two users (Bob Jones and Charles Smith) and one group (cm\_sales).

```
<?xml Version="1.0" encoding="UTF-8"?>
<principals>
```

```
<server_name>ldapserver</server_name>
<ip_address>10.14.60.60</ip_address>

<users>
  <user>
    <dn>CN=Bob Jones,DC=costa_mesa,O=FileNet</dn>
    <id>LDAP/bjones</id>
    <cn>Bob Jones</cn>
    <description>Good guy</description>
  </user>
  <user>
    <dn>UID=csmith,DC=costa_mesa,O=FileNet</dn>
    <id>LDAP/csmith</id>
    <cn>Charles Smith</cn>
  </user>
</users>
<groups>
  <group>
    <dn>CN=cm_mktg,DC=costa_mesa,O=FileNet</dn>
    <id>LDAP/cm_sales</id>
    <cn>cm_sales</cn>
    <description>We make it happen</description>
  </group>
</groups>
<members>
  <member>
    <groupName>CN=cm_mktg,DC=costa_mesa,O=FileNet</groupName>
    <memberName>CN=Bob Jones,DC=costa_mesa,O=FileNet</memberName>
    <memberType>user</memberType>
  </member>
</members>
</principals>
```

## File Format Concepts

The file format is able to capture all relevant security information for Content Services and Image Services, including users, groups, and group memberships, including nested groups.

The main entries being modeled in the logon information are the security principals (users and groups). In the LDAP export file, group nodes are delimited by a group tag (<group></group>). User nodes are delimited by a user tag (<user></user>). Group memberships are delimited by member note tags (<member></member>).

A membership element must uniquely designate the group node and the member node. This is done by calling out all the unique names of the group and its members. In LDAP, the full distinguished name (DN) is unique.

A security principal is a user or a group. The syntax of the principal names in a particular FileNet document security database must be converted to and from the principal names in the LDAP export file.

## LDAP Names

For LDAP, distinguished names (DN) could contain the following elements, which can occur several times, and in a least significant to most significant order:

**CN** - Common Name

**L** - Locality Name

**ST** - State or Province Name

**O** - Organization Name

**OU** - Organization Unit Name

**C** - Country Name

**STREET** -Street Address

**DC** - Domain Component

**UID** - User ID

---

**Note** All characters, including blanks and other white space within the values of the XML tags in the support data file are significant and all characters between the tags are also part of the value.

---

- In the export data file, the value of the <dn> XML tag is the full LDAP distinguished name of the user or group.
- In the export data file, the value of <id> is the value to be stored in the target FileNet security database as the name of the user or group. For data from LDAP sources, the first 5 characters of the <id> value are **LDAP/**. The **ldap\_exp** program sets the value it develops for the <id> tag with these 5 characters. The **ldap\_exp** program can optionally develop the value of <id> from primary and secondary LDAP attributes that the user specifies, or it can develop the value of the <id> tag using its default convention. The <id> attribute is the only LDAP attribute value that the **ldap\_exp** program modifies before storing it in the corresponding XML tag in the export data file.
- In the export data file, the value of <description> is the value of the LDAP description attribute.

The **ldap\_exp** program limits the length of the names of the XML tags to 35 characters. In addition, the **LDAP\_import** program could also

reject or truncate some users or groups due to name size limits being exceeded.

### Expired Names

All LDAP objects on the IS system not found in the export data file will be **expired** on the IS system by the sync operation after the import. For example:

**In LDAP\_import 1:** The XML file contains LDAP/GROUPA and its users LDAP/USERW and LDAP/USERX. After LDAP\_import is run, the IS server will contain LDAP/GROUPA, LDAP/USERW, LDAP/USERX.

**In LDAP\_import 2:** The XML file contains LDAP/GROUPB and its users LDAP/USERY and LDAP/USERZ, and for some reason LDAP/GROUPA and its users LDAP/USERW and LDAP/USERY have been removed from the XML file because the user thought this group and its users were no longer needed in the file since they already were on the IS Server. After LDAP\_import is run this time the IS server will contain LDAP/GROUPB, LDAP/USERY, LDAP/USERZ. LDAP/USERA and its users LDAP/USERW and LDAP/USERX will be marked expired.

**In LDAP\_import 3:** The XML file contains LDAP/GROUPB and its users LDAP/USERY and LDAP/USERZ along with LDAP/GROUPA and its users LDAP/USERW and LDAP/USERY. After LDAP\_import is run this time the IS server will contain LDAP/GROUPB, LDAP/USERY, LDAP/USERZ. LDAP/USERA and its users LDAP/USERW and LDAP/USERX will be marked unexpired.

---

### Note

When users or groups are missing from current XML file and present in the previous XML file, the users are expired. If the users or groups are added back into the XML file and imported, the users and groups will be unexpired.

---

## Image Services Names

For Image Service systems, the **LDAP\_import** program gets the name of the user or group from the value of <id> in the export data file. For IS systems, user and group names stored in the FileNet security database are limited to 40 non-null characters (less the **LDAP/** prefix). **LDAP\_import** for IS systems will store the value of the <id> tag as the principal name, including the **LDAP/** prefix.

**LDAP\_import** gets the value of the description field in the security database from the value of the XML <description> tag. It will truncate the value to 79 (not 80) characters due to size limitations (one character is reserved for a termination character).

## XML File Format Description

- An LDAP export file consists of one <principals> element, which contains one <server\_name> element, one <ip\_address> element, one <users> element, one <groups> element, and one <members> element.
- The <server\_name> element and the <ip\_address> element are sources used for strong password creation.
- Within the <users> element is zero or more <user> elements. Each <user> element must contain a <dn> and an <id> element. The <dn> element must have a tag. A <user> element may optionally also contain a <description> element. The export program prefixes the value of the <id> element with the characters **LDAP/**. So, the first 5 characters of the value of the <id> element are **LDAP/**.
- Within the <groups> element is zero or more <group> elements. Each <group> element must contain a <dn> and an <id> element. The <dn> element must have a tag. A <group> element may optionally also contain a <description> element. The export program prefixes the value of the <id> element with the characters **LDAP/** for group names.

- Within the <members> element is zero or more <member> elements. Each <member> element must contain a <groupName> element, a <memberName> element, and a <memberType> element. The <groupName> contains the exact <dn> of the group. The <memberName> contains the exact <dn> of the member. The value of the <memberType> element must be **user** or **group**.

The following is the screen output generated by running the `command` with the sample XML file.

- Two users are added: LDAP/BJONES and LDAP/CSMITH
- One group is added: LDAP/CM\_SALES

- One membership: LDAP/BJONES is a member of LDAP/CM\_SALES.

```
C:\jsmith\ldap>ldap_import /hmyserver /isample.xml
Preprocessing input file 'sample.xml'...
Locating IS server...please wait

Authorized IS user name: SysAdmin
Authorized IS user password:
Generating IS data files...please wait...
Generating LDAP data files...please wait...
Importing LDAP XML inout file 'sample.xml'...
2002/01/08 12:11:57.392 92,0,458 <fnsw>ldap_import(340.2284 0x154.8ec)...
Importing LDAP security to IS host [myserver], version 4.0.0

Synchronizing users started [01-08-2002, 12:11 PM]...please wait...
Synchronizing groups started [01-08-2002, 12:11 PM]...please wait...
Synchronizing members started [01-08-2002, 12:11 PM]...please wait...
Total user objects added = 2
Total group objects added = 1
Total objects expired = 0
Total objects unexpired = 0
Total objects updated = 0
Total members added = 1
Import process completed. Please check log file for possible errors.
```

**Note** Two users, one group and one membership were added. The objects consist of users and groups.



## Related Topics

[“ldap\\_exp” on page 662](#) and [“LDAP\\_password” on page 690](#).

---

**Note** Portions of this software were developed using the “expat XML parser library”. The contents of this file are subject to the **Mozilla Public License Version 1.1** (the "License"); you might not use this file except in compliance with the License. You can obtain a copy of the License at <http://www.mozilla.org/MPL/>.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either expressed or implied. See the License for the specific governing rights and limitations under the License.

---

## LDAP\_password

### Description

The LDAP\_password tool is used to encrypt a password that will be used by LDAP\_import during the import process, so you can provide a user name with an encrypted password through the command line without revealing the real password.

To review the LDAP\_import tool go to [“LDAP\\_import” on page 671](#).

### Use

To create an encrypted password that will be used later in the LDAP\_import command, simply enter the LDAP\_password command with the real password within quotation marks as the only parameter.

For example, if the password is SysAdmin, enter:

```
LDAP_password “SysAdmin”
```

The program will display the encrypted password (for example, SysAdmin = “:015K+]%”) that you would then enter in the LDAP\_import tool command using the /p“<enpwd>” argument.

### Important

---

On Windows systems, if the encrypted password contains a % or a \, you need to add a second % or \. For example, /p”:015K+]%” needs to be entered as /p”:015K+]%%”. A % becomes %% and a \ becomes \\.

---

The encrypted password is only used through the command line. It is useful when LDAP\_import is invoked or automated by a batch file (that is, no prompting for user id and password) or for a cron job.

**Note** LDAP\_password is a standalone program, but the user that you are creating the password for should have enough rights to add, update, and delete users and groups in Image Services.

---

## Syntax

LDAP\_password "<pwd>"

## Procedure

No specific procedure is required.

## Related Topics

[“ldap\\_exp” on page 662](#) and [“LDAP\\_import” on page 671](#).

# less

## Description

The less tool is a file viewing utility that offers several benefits over other viewing utilities and text editors, such as:

- As you view a file, you can move backward as well as forward through the file contents.
- The less tool does not have to read the entire input file before starting so with large input files, it starts faster than text editors such as vi.
- A large variety of terminals support less, including limited support for hard copy terminals. On a hard copy terminal, lines that should be printed at the top of the screen are prefixed with an “up” arrow.

## Use

Use the less command to view files.

After you enter the less command with a file name, less displays the file contents. You can scroll forward (using the space bar or commands of less) or backward (by pressing the **b** key on your keyboard or entering less commands). Using less commands, you can scroll in increments of one window, a half window, or a line. When you finish viewing the file, enter **q** or press the escape key (ESC) to exit.

## Syntax

```
less [-? | -h] <filename> [ flags ]
```

**-?** Displays a summary (help screen) of the commands of less. When you specify this option, less ignores all other options and exits after displaying the help screen. (Depending on how your shell interprets the question mark, you might need to enclose the question mark in quotes, for example, "-?".)

**-h** Displays a summary (help screen) of all available options and flags of less

**<filename>** Specifies the file you want to view

## Commands

Run **less -?** to list all commands of less. The following table identifies one or more commands you can use to perform a single task.

less Command Summary

Command	Task Performed
h or H	Displays help list for less command
q, :q, :Q, or ZZ	Exits less command
e, ^E, j, ^N, or CR	Moves forward one line (or <i>N</i> lines)
y, ^Y, k, ^K, or ^P	Moves backward one line (or <i>N</i> lines)
f, ^F, ^V, or SPACE	Moves forward one window (or <i>N</i> lines)
b, ^B, or ESC-v	Moves backward one window (or <i>N</i> lines)
<ol style="list-style-type: none"> <li>Notes in parentheses indicate action given for <i>N</i> where <i>N</i> is the number entered before the corresponding command.</li> <li>^ represents the control key on your keyboard.</li> <li>By default, a window is the screen height and a half-window is half the screen height.</li> </ol>	

## less Command Summary

Command	Task Performed
z	Moves forward one window (and sets window to <i>N</i> )
w	Moves backward one window (and sets window to <i>N</i> )
d or ^D	Moves forward one half-window (and sets half-window to <i>N</i> )
u or ^U	Moves backward one half-window (and sets half-window to <i>N</i> )
r, ^R, ^L	Repaints the screen
R	Repaints the screen, discarding buffered input
/pattern	Searches forward for ( <i>N</i> th) matching line
?pattern	Searches backward for ( <i>N</i> th) matching line
ESC-/pattern	Searches from top of file for ( <i>N</i> th) matching line
!/pattern	Searches forward for ( <i>N</i> th) NON-matching line
?!pattern	Searches backward for ( <i>N</i> th) NON-matching line
ESC-!/pattern	Searches from top of file for ( <i>N</i> th) NON-matching line
n	Repeats previous search (for <i>N</i> th occurrence)
ESC-n	Repeats previous search in reverse direction
<ol style="list-style-type: none"> <li>1. Notes in parentheses indicate action given for <i>N</i> where <i>N</i> is the number entered before the corresponding command.</li> <li>2. ^ represents the control key on your keyboard.</li> <li>3. By default, a window is the screen height and a half-window is half the screen height.</li> </ol>	

**Flags**

Run **less -h** to list all flags and options

## Checklist

Before using the `-?` option of `less`, understand how your shell interprets a question mark. You might need to enclose the question mark in quotes, for example, `"-?"`.

## Procedure

No specific procedure is required.

## Related Topics

Refer to the online manual (`man`) pages for details of the `less` command and its commands, flags, and options.

# log\_create

## Description

The log\_create tool creates a circular log file to receive log records generated by performance analysis tools.

## Use

Use log\_create to create a circular log file or increase the size of an existing log file.

You must specify the name of the log file to create. As an option, you can also indicate a maximum number of 1024-byte physical records that can comprise the log file.

## Syntax

```
log_create <logfile> [-m#]
```

**<logfile>** Specifies the log file to create. If you don't specify a directory path with the file name, log\_create places the file in the current directory.

**-m#** Maximum number of physical records as specified by the value of #

The default maximum number of physical records is 512; each record is 1024 bytes in length.



## Example

The UNIX logo is a red parallelogram with the word "UNIX" in white, bold, uppercase letters.

In the UNIX platform example below, `log_create` creates a new log file named `testlog` in the `/fnsw/local/tmp/test` directory. The log file is limited to a maximum of 256 records. The `ls` command verifies the existence of the new log file in the directory.

```
corona(root)/> log_create /fnsw/local/tmp/test/testlog -m256
corona(root)/> ls /fnsw/local/tmp/test/t*
testlog testperf
```

## Checklist

Before you use `log_create`, be aware of the following:

- You must specify a log file name.
- If you do not specify a maximum number of records for the log file, `log_create` uses the default of 512 1024-byte records.

## Procedure

No specific procedure is required.

## Related Topics

[“log\\_dir” on page 699](#)

[“log\\_extract” on page 702](#)

[“perf\\_mon” on page 954](#)

[“perf\\_report” on page 968](#)

---

# log\_dir

## Description

The log\_dir tool lists and displays, in reverse chronological order, all records in a circular log file created by the perf\_mon or log\_create tool. For each record in the log file, log\_dir lists the record type, record length, and time stamp in reverse chronological order.

## Use

Use log\_dir to view entries in a circular log file. Use the short (-s) option to limit the display of entries to only the newest and the oldest entries.

A common use of log\_dir is to view entries in the default perf\_mon capture log:

/fnsw/local/sdperflog           for UNIX platforms

\fnsw\_loc\sdperflog           for Windows Server platforms

## Syntax

```
log_dir <logfile> [-s]
```

**<logfile>** Specifies the log file to display

**-s** Displays only the newest and oldest log file entries

## Sample Output

### UNIX

In the first example below from a UNIX platform, `log_dir` displays all entries in the log file in reverse chronological order. The second example, uses the `-s` option to display only the newest and oldest log file entries.

```
corona(root)/> log_dir /fnsw/local/tmp/test/log111896
LOGFILE : /fnsw/local/tmp/test/log111896 IN REVERSE CHRONOLOGICAL ORDER

number type      length          end time
11      1          6112  Mon Nov 18 14:18:57 1996
10      1          6112  Mon Nov 18 14:18:52 1996
 9      1          6112  Mon Nov 18 14:18:47 1996
 8      1          6112  Mon Nov 18 14:18:42 1996
 7      1          6112  Mon Nov 18 14:18:37 1996
 6      1          6112  Mon Nov 18 14:18:32 1996
 5      1          6112  Mon Nov 18 14:18:27 1996
 4      1          6112  Mon Nov 18 14:18:22 1996
 3      1          6112  Mon Nov 18 14:18:17 1996
 2      1          6112  Mon Nov 18 14:18:12 1996
 1      2              1  Mon Nov 18 14:18:12 1996

corona(root)/> log_dir /fnsw/local/tmp/test/log111896 -s
LOGFILE : /fnsw/local/tmp/test/log111896 IN REVERSE CHRONOLOGICAL ORDER

number type      length          end time
11 16112Mon Nov 18 14:18:57 1996
 1 21Mon Nov 18 14:18:12 1996
```

## Checklist

To use log\_dir, you must specify an existing log file name.

## Procedure

No specific procedure is required.

## Related Topics

[“log\\_create” on page 696](#)

[“log\\_extract” on page 702](#)

[“perf\\_mon” on page 954](#)

[“perf\\_report” on page 968](#)

# log\_extract

## Description

The log\_extract tool extracts a range of records from one circular log file and appends them to another circular log file. If the destination log file does not exist, the tool creates a new log file with the same attributes as the source log file.

The default circular log file is:

/fnsw/local/sdperflog           for UNIX platforms

<drive>:\fnsw\_loc\sdperflog   for Windows Server platforms

Use log\_extract to select a specified number of logical records from a circular log file. A common use of log\_extract is to obtain for later viewing some of the perflog entries before the program overwrites the entries in the circular log file.

Select a starting logical record number for extraction. The number of records to extract is determined by the value you assign to the **number** argument.

You can choose to extract the records in one of two directions—forward (records that are newer than the starting record) or backward (records that are older than the starting record). To extract the records that are newer than the starting record, simply specify a value **n** for the number argument. To extract the records that are older than the starting record, specify **-n** for the number argument.

## Syntax

log\_extract <logfile> <logical rec num> <[-]number> <destfile>

<logfile> Identifies log file from which to extract records

<logical rec num> Specifies starting number of the logical records to extract

<[-]number> Performs a forward or backward search for the specified number of records.

- If you specify a number **without** a preceding hyphen (-), the search proceeds forward from the starting record.
- If you specify a number **with** a preceding hyphen (-), the search proceeds backward from the starting record.

<destfile> Identifies destination log file name to which extracted records are appended

If the log file exists, log\_extract appends extracted records to the file. If the log file does not exist, log\_extract creates a new log file with the same attributes as the source log file and appends the extracted records to the new file.

## Sample Output

### UNIX

In the example below, log\_extract selects five records from a UNIX circular log file /fnsw/local/tmp/extlog. It writes the five extracted records to /fnsw/local/tmp/extrct5 and the log\_dir tool displays the appended records.

```
corona(root)/> log_extract /fnsw/local/tmp/extlog 1 5 /fnsw/local/tmp/extrct5
Starting at logical rec 1, and selecting 5 records
corona(root)/>
corona(root)/> log_dir /fnsw/local/tmp/extrct5
LOGFILE : /fnsw/local/tmp/extrct5 IN REVERSE CHRONOLOGICAL ORDER

number type      length          end time
-----
      5   1       7236   Mon Nov 4 16:11:46 1996
      4   1       7236   Mon Nov 4 16:06:46 1996
      3   1       7236   Mon Nov 4 16:01:46 1996
      2   1       7160   Mon Nov 4 15:56:46 1996
      1   2           1   Mon Nov 4 15:56:46 1996
corona(root)/>
```



## Checklist

Before you use log\_extract, be aware of the following:

- You must specify a log file name from which log\_extract can select the requested records.
- If the destination log file does not exist, log\_extract creates a new destination log file with the same attributes as the source log file.

## Procedure

No specific procedure is required.

## Related Topics

[\*\*“log\\_create” on page 696\*\*](#)

[\*\*“log\\_dir” on page 699\*\*](#)

[\*\*“perf\\_mon” on page 954\*\*](#)

[\*\*“perf\\_report” on page 968\*\*](#)

# manifest

## Description

The manifest tool maintains a file that specifies the contents of a server software release partition. The tool supports commands for creating, maintaining, and querying the contents of the partition. The program only maintains the file and does not maintain the file system. That is, manifest does not create, delete, or update other files.

The manifest tool stores its file in a compressed form. The manifest file includes information about each regular file in the file system of the release partition. The manifest tool attempts to filter out information about device files, configuration files, and log files.

Typical information saved in the manifest for each file include the full path name of the file, the file mode, the last date and time the file was modified, the file size, a checksum on the file, and the stamp on the file.

## Use

Each release of software includes a manifest file generated as one of the final steps in creating a release tape. As you use the manifest tool to load new versions of the file into the release partition, you can update the file using the add and delete options. You can also use manifest to identify file corruption and the installation of unauthorized versions of software.

---

**Note** The use of manifest is restricted to support personnel.

---

## Syntax

manifest <commands>

The individual manifest commands are described below.

## Commands

In the command descriptions that follow, replace items in angle brackets (< >) with an appropriate file name or station number as indicated. You can specify multiple commands in a single invocation of the manifest command.

---

**Tip** To reduce overhead associated with initiating the manifest program, specify multiple commands in a single invocation.

---

- |                                |  |
|--------------------------------|--|
| <b>add &lt;filename&gt;</b>    | Adds the specified file to the manifest  |
| <b>check &lt;filename&gt;</b>  | Checks the given file against the manifest   |
| <b>checkall</b>                | Compares all files against the manifest and prints the results to standard output (usually the display device from which you invoked the command)                              |
| <b>delete &lt;filename&gt;</b> | Deletes the specified file from the manifest   |
| <b>show &lt;filename&gt;</b>   | Shows information about the given file from both the file itself and from the manifest file<br><br>It shows data from the manifest file first, followed by data from the file. |
| <b>update &lt;filename&gt;</b> | Updates the manifest with information from the specified file  |

## Example

The following command adds the named file to the manifest.

```
manifest add relfile
```

## Sample Output

The sample below displays information from the manifest for `/fnsw/bin/TAPs`.

```
corona(fnsw)/usr/fnsw> manifest show /fnsw/bin/TAPs
From the manifest:
./fnsw/bin/TAPs 102755 96/09/06 13:34:49 67449 22779 17 sys 3.3.1.19(0)
dev 3.3.1.0.0
From the current file:
./fnsw/bin/TAPs 102755 96/09/06 13:34:49 67449 22779 17 sys 3.3.1.19(0)
dev 3.3.1.0.0
```

## Checklist

Before using manifest, be aware of the following:

- Updates apply only to the manifest file, not to the file system.
- The use of manifest is restricted to support personnel.

## Procedure

No specific procedure is required for this tool.

# migrate\_cdb\_changes

## Description

This migrate tool updates the performance-related parameters in the clone system's configuration database (.cdb) file from a specified file in the /fnsw/local/sd/conf\_db/import directory. This tool is designed to be the last step in the process of cloning the IS system configuration of multiple IS systems from a single master configuration database file and is done as a part of a migration. The other tool associated with this process is the [export\\_cdb](#) tool.

For more information on the cloning process, see [“export\\_cdb” on page 409](#).

---

**Note** All the path names documented for this tool are shown in UNIX format (e.g., /fnsw/local/sd/conf\_db/import). For Windows Server path names, substitute the /fnsw/local/ with <drive>:\fnsw\_loc\ and use a back slash (\) instead of a forward slash (/) throughout the rest of the path (e.g., <drive>:\fnsw\_loc\sd\conf\_db\import).

---

## Use

You use this tool to migrate the differences from a migrated .cdb file into an existing ims\_exp\_xx.cdb file on another system. The migrate file must be located in the in /fnsw/local/conf\_db/import directory. Only changes to the .cdb performance parameters are migrated. Modifications to devices, datasets, caches, and most relational database changes aren't migrated. This tool always selects the latest version of the .cdb file to import.

The migrate\_cdb\_changes tool should be used in conjunction with the export\_cdb tool.

## Syntax

```
migrate_cdb_changes [-s <source_prefix>]
```

### Options

**-s <source\_prefix>**

Optionally allows the user to specify a prefix for the source file other than the default “ims” prefix.

---

### Note

When you migrate using the optional -s parameter, you are still creating a new revision for the default IS configuration database file (ims\_xx.cdb). The xx in the file name will be one revision higher.

---

## Procedure

- 1 On the IS server being used as the master server, use fn\_edit to define the configuration database parameters. This has probably been done and the latest cdb file is established.
- 2 On the IS server being used as the master server, run **export\_cdb** to make a copy of the master IS server’s configuration database file by entering the following command:

### **export\_cdb**

In the above example, the export file would be created in the following directory structure: /fnsw/local/sd/conf\_db/export/ims\_exp\_xx.cdb, where xx is the revision number of the .cdb file.

If you want to use the optional **-s <source\_prefix>** parameter, <source\_prefix> is the value you use to associate the file with the clone server the configuration file is being exported to (for example, california). In this -s example, the export file would be created in the following directory structure: /fnsw/local/sd/conf\_db/export/california\_exp\_xx.cdb, where xx is the revision number of the .cdb file.

For more information on the **export\_cdb** tool, see [“export\\_cdb” on page 409](#)

- 3 Manually move or copy the newly created file from the master IS server to the clone IS server by placing it in the following location on the clone IS server:

```
/fnsw/local/sd/conf_db/import/ims_exp_xx.cdb
```

- 4 On the IS server being used as the clone server, run the following command to import the configuration database file and update the .cdb file:

### **migrate\_cdb\_changes**

Again if you want to use the optional **-s <source\_prefix>** parameter, <source\_prefix> is the value you use to associate the file with the clone server the configuration file is being exported to (for example, california).

## **Related Topics**

[“export\\_cdb” on page 409](#)

[“import\\_cdb” on page 576](#)

## MKF\_ddl

### Description

The Data Definition Language (ddl) file is the ASCII text file containing source text for an MKF database definition.

MKF\_ddl processes the ddl file and writes the database description specified in the ddl file to the MKF database; the database now contains information about itself. In addition, options are available to initialize, update, and compare the ddl file and check the syntax of the ddl file.

fn\_edit generates the ddl text files, so you need to run fn\_edit to change them. Otherwise, your changes get lost when the software is started.

The primary purpose of the initialize option is to create an empty database with a valid data description so the option is normally used only once to initially create an MKF database. You use it only rarely after that. If used on a live database, the initialize option destroys any and all data in the database. If you suspect that any valid data exists in the database you plan to initialize, back up the database before you initialize.

---

#### CAUTION

If you specify the `-initialize` option, MKF\_ddl zeroes out all the rows in the database. **All data is lost.**

---

The update option causes MKF\_ddl to write a new database description into the database. However, it does not modify or delete any files, partitions, tables, or columns in the database. It adds new data to the database (files, partitions, tables, columns, etc.) at the end of their section. For example, it adds a new table last, after all previous tables; it



adds a new column as the last column of its table; and so on. The exception is the size of the last data file or last log file.

---

**Note** When you run `MKF_ddl -update`, verify that `MKF_ddl` is the first program to use the database after you start FileNet software. If another program uses the database before `MKF_ddl -update` runs, `MKF_ddl` generates an error.

---

You can change some database parameters with `MKF_ddl -update`. The following table indicates which parameters you can change:

Database Parameter	Can be changed with MKF_ddl -update?
number_of_buffers	Yes
rl_update_frequency	Yes
read_after_write	Yes
max_concurrent_transactions	Yes
max_long_transactions	Yes
max_tables	No
max_columns_per_row	No
rl_buf_blocks	Yes

The final step in update processing calls MKF to update the available space btrees.

---

**Note** If the system fails before the update phase completes, rerun `MKF_ddl update`.

---

`MKF_ddl` can perform a syntax check on the specified ddl file without deleting or changing existing rows in the ddl file or the database.

MKF\_ddl can also compare the ddl file and the database, returning an exit status indicating the results of the comparison. When you run MKF\_ddl with the comparison option, it does not delete or change existing rows in the database. The format of the exit status is:

exit status = **n** (text)

where **n** is the exit status code followed by the text for the exit status condition.

The table below describes valid exit status codes:

Exit Status Code and Text	Description
0 (equivalent)	The ddl text and the database itself are the same.
1 (update OK)	The ddl text and the database itself differ, but MKF_ddl <code>-update</code> would succeed in making them identical.
2 (unchangeable parameter)	The ddl text differs from the database itself and attempts to change a database parameter for which change is not allowed.
3 (parameter value error)	The ddl text specifies an illegal value for a database parameter.
4 (illegal update FILES)	The ddl text specifies an illegal update to the FILES section.
5 (illegal update TABLES)	The ddl text specifies an illegal update to the TABLES section.
6 (syntax error)	A common cause for this exit status is a ddl update that increases the number of columns for a single table or the number of tables beyond the maximum allowed by the PARAMETERS section.
7 (command line error)	An error exists in the MKF_ddl command line options.
8 (other)	Some other error has occurred, for example, a file I/O error.

## Use

System configuration tools (System Configuration Editor and `fn_edit`) are normally used to create, initialize, and update the `ddl` text file. The system configuration tools generate the Data Definition Language (`ddl`) source file and using `MKF_ddl` to update it could result in out-of-synchronization problems between the `ddl` source file and the database.

However, on rare occasions, your service representative could direct you to use `MKF_ddl` to initialize or update the MKF transient, permanent, or Network Clearinghouse databases.

Using other options, you can check the `ddl` file for proper syntax and compare the contents of the `ddl` file to the database.

## Syntax

MKF\_ddl <ddlsource file> [-initialize | -update | -syntax | -compare]

- <ddlsource file>** Full path name of the ddl file for the MKF database you want to initialize or update
- For example, the full UNIX path name of the MKF transient database ddl file is /fnsw/local/sd/1/transient.ddl.
- initialize** Zeroes out the database specified in the named ddl file, losing all rows stored in the database
- update** Adds information to the database description without deleting or changing any existing rows
- syntax** Checks for correct syntax of a ddl source file without deleting or changing any existing rows
- compare** Compares the text data in the ddl file with the database itself and returns an exit status code

## Sample Output

The following sample output shows the messages issued for an MKF\_ddl update of the permanent database.

```
> MKF_ddl /fnsw/local/sd/1/permanent.ddl -update
updating avsp tables for table family_disk ...
updating avsp tables for table surf_dyn_info ...
updating avsp tables for table surf_stat_info ...
updating avsp tables for table docs ...
updating avsp tables for table cluster_map ...
updating avsp tables for table scalar_numbers ...
updating avsp tables for table od_stats ...
updating avsp tables for table annotations ...
updating avsp tables for table surf_locator ...
updating avsp tables for table family_locator ...
updating avsp tables for table remote_family ...
"MKF_ddl /fnsw/local/sd/1/permanent.ddl -update" exit status = 0
(success)(This is not an error.)
```

The sample below shows the messages issued for initialization of the permanent database on a Windows system.

```
zeroing F:\FNSW_DB\PERMANENT_DB0
  blocks 0x0000000000000000 to 0x0000000000000A7FF
zeroing F:\FNSW_DB\PERMANENT_DB0
  blocks 0x00000000000003000 to 0x0000000000000A7FF
zeroing F:\FNSW_DB\PERMANENT_DB0
  blocks 0x00000000000006000 to 0x0000000000000A7FF
zeroing F:\FNSW_DB\PERMANENT_DB0
  blocks 0x00000000000009000 to 0x0000000000000A7FF
zeroing G:\FNSW_DB\PERMANENT_RL0
  blocks 0x0000000000000000 to 0x00000000000002FFF
"MKF_ddl permanent.ddl -initialize" exit status = 0 (success) (This is not
an error.)
```

## Checklist

Before you use MKF\_ddl, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_ddl.
- You should not use MKF\_ddl to initialize or update a database unless specifically directed to do so by your service representative.
- The MKF\_ddl program should only be run when the database is shut down, no processes are linked to MKF, and no shared memory for MKF exists.

---

### Note

If you attempt to use the `-initialize` option against an in-use database, the program writes an error message to the system event log and the function terminates.

---

- You must run MKF\_ddl on the server on which the target database resides.
- The `-initialize` option zeroes out all database rows for the target database. **Do not use this option** unless you are specifically directed to do so by your service representative.
- Initializing a transient database deletes any affiliated objects in the associated cache.
- Initializing a permanent database deletes any affiliated documents on storage media.
- Initializing the NCH database deletes all resource names stored in the database.

- The `-update` option does not affect any current rows in the database. Updating the database is only necessary if you have changed the contents of its corresponding ddl file. Use the Configuration Editor to change the contents of a ddl file.
- Entering the `MKF_ddl` command with no options displays help information.
- MKF database initialization can also be accomplished with the initialization options of the `fn_util` command. See online help for `fn_util`.

## Procedure

Follow these steps to initialize a database:

- 1 Shut down the database.

Use **killfns** **-A -D -y** to kill the FileNet software and delete shared memory.

- 2 At the command line, enter MKF\_ddl with the appropriate ddl file name and options.

For example, to initialize the transient database, use the following command:

```
MKF_ddl /fns/local/sd/1/transient.ddl -initialize (UNIX)  
MKF_ddl D:\FNSWDEV1\TRANSIENT.DLL -initialize (Windows)
```

If you initialize the NCH database, the Image Services System Configuration Builder restores the NCH database resource names when FileNet software is restarted.

- 3 Restart FileNet software.

Restarting the software automatically updates the ddl file.

## Related Topics

For information about system configuration tools (System Configuration Editor or fn\_edit), see your FileNet system's online help.

Refer to online help for the fn\_util command.



---

## MKF\_debug

### Description

The MKF\_debug program displays information to assist in diagnosing database hangs. MKF\_debug displays a list of all processes linked to the MKF shared library (abstract) and the status of all known MKF databases on the server. MKF\_debug takes a “snapshot” and therefore reports only events at the time you start the program.

### Use

Use MKF\_debug to determine whether a database is hung. If the database is hung, MKF\_debug can also identify the process causing the hang.

Use the quick option to display only critical information, for example, which, if any, processes died in MKF, the currently open databases, and the states of currently open databases. Use the verbose option to display more detailed output.

---

**Tip** As a first step in debugging, check the output report for a **yes** in the **database fatal err** field. See [“database fatal err” on page 728](#) for more information.

---

### Syntax

```
MKF_debug [ -q | -v ]
```

- q** Quick option displays only critical information
- v** Verbose option displays detailed information (the default)

## Sample Output

The output below shows which processes are linked locally to the MKF shared library (abstract):

```
All processes using an MKF database on this station:

* if  + in
dead  MKF   process num  program name
-----
                                9244  NCHs
                                10539  MKF_clean
                                11565  INXs
                                11825  PRIs
=====
```

- An asterisk (\*) appearing in the first column (“\* if dead”) indicates the process terminated.
- A plus sign (+) in the second column (“+ in MKF”) indicates the process is executing MKF code to perform an MKF operation (for example, find, insert, etc.).
- The process num column contains the process ID number of the program listed in the program name column.

Combinations of indicators show what is happening to your MKF databases and in the MKF subsystem. Use the following guidelines to interpret these indicator combinations.

### Asterisk (\*), no plus sign

An asterisk (\*) without a plus sign (+) in the second column indicates the process terminated but was not executing MKF code at the time it

terminated. The fact that the process is dead might indicate a problem. If it is a problem, the problem is typically not in MKF. For example, an asterisk displays in this column even if a process terminates normally, a user issues a Control+c key sequence to prematurely terminate a program or process, or a user kills a process linked to MKF.

The background cleanup daemon, MKF\_clean, eventually backs out the transactions of processes that died in the middle of an MKF transaction but were outside of MKF kernel code at the time of the death. Such conditions usually indicate a bug or problem in the application program.

### **Plus sign (+), no asterisk**

On an active system, you frequently see a plus sign in the second column, but no asterisk in the first column. This combination indicates that, at the time the snapshot is taken by MKF\_debug, the process is in the middle of an MKF operation.

### **Asterisk (\*) and plus sign (+)**

If the process dies in MKF kernel code, the background cleanup daemon, MKF\_clean, cannot release the hung database. Check for a core file and check the system event log for error messages.

---

**Tip** To release the hung database, restart the FileNet software.

---

The output below provides additional information about the MKF databases. Output fields are described following the output sample.

---

Thu Apr 12 16:36:12 2001

All processes using an MKF database on this station:

\* if + in

dead	MKF	process num	program name
-----	-----	-----	-----
		2076	MKF_writer 0
		1080	MKF_clean

=====  
MKF database number 0: at 0x4600CBE0

base data file name: F:\FNSW\_DB\PERMANENT\_DB0  
db state: 2 (normal)  
database fatal err: no  
description valid: 1  
db is recovered: 1  
abort mode: 0  
opendescpid: 0  
number of buffers: 256  
num virgin buffers: 255  
block list at: 0x46081D00, elt sz = 92  
num avail buffs: 0  
num bufs in mru ring: 1  
tot olayable buffs: 256  
tot non-olayable bufs: 0  
tot unmod data blks: 0  
&acttr: 0x4600D084, elt sz = 588, num elts = 13  
&datafileinx: 0x4600EE60, elt sz = 4, num elts = 8  
&aijfileinx: 0x4600EE80, elt sz = 4, num elts = 8  
&bijfileinx: 0x4600EEA0, elt sz = 4, num elts = 8  
blklisthead: 0x46087980 num\_ble\_hh = 331  
max long trans cc: 1

---

```
long trans claim cnt: 1
max trans claim cnt: 3
trans claim count: 3
transres_ilk          0xFACE000E
longtransres_ilk     0xFACE000F
small->large trans: 0
small->large tr fail: 0
small->large tr err: 0
getbuffer retries: 0
getbuffer errs: 0
cb_p:                0x46017000
mbi demon pid:       2076
mbi demon enabled: 0
log_updaij_time:    0
mbi heap at:        0x4628B470
num_heap:           257
heap_av at:         0x4628C4B0
num_heap_levels:    9
heap_avbv:          0x7FC00000
num writeahead:     0x0000000000000000
num writeahead bursts: 0 (0x00000000)
# calls MKF_shutdown: 0
# calls MKF_open: 1
# calls MKF_close: 1
# calls MKF_open_cursor: 11
# calls MKF_close_cursor: 11
# calls MKF_get_item_number: 0
# calls MKF_bind: 0
# calls MKF_begin_transaction: 0
# calls MKF_end_transaction: 0
# calls MKF_abort_transaction: 0
# calls MKF_set_position: 0
# calls MKF_find: 0
# calls MKF_insert: 0
# calls MKF_update: 0
# calls MKF_delete: 0
# calls MKF_open_description: 2
```

---

```
# calls MKF_close_description:    2
# calls MKF_get_parameters:       0
# calls MKF_set_parameters:       0
# calls MKF_get_file:             0
# calls MKF_set_file:             0
# calls MKF_get_record:           0
# calls MKF_set_record:           0
# calls MKF_get_item:             0
# calls MKF_set_item:             0
# calls MKF_get_aij_bsn:          0
# calls MKF_update_avsp:          0
# calls MKF_transaction_state:    0
# calls MKF_set_abortmode:        0
# short transactions started:     33
# short transactions completed:   33
# short transactions aborted:     0
# long transactions started:      0
# long transactions completed:    0
# long transactions aborted:     0
# attempts to convert short to long transaction: 0
# successful conversions of short to long trans: 0
# deadlocks:                      0
# single-operation MKF_find:       0
# single-operation MKF_insert:     0
# calls on mkf_readblock:         1
# buffer pool cache hits:         0
# buffers:                        256
# physical reads of aij:          1
# blocks read from aij:           24
# physical writes to aij:         0
# blocks written to aij:          0
# physical reads on bij:          0
# blocks read from bij:           0
# physical writes to bij:         0
# blocks written to bij:          0
# physical reads on data files:    1
# blocks read from data files:    1
```

---

```
# physical writes to data files:    1
# blocks written to data files:    1
# calls on getbuffer:              1
# total calls on mkf_divest:        1
# calls on mkf_divest(completely):  1
# total calls on mkf_search:        0
# calls on mkf_search (firsttry=TRUE): 0
# calls on mkf_search (forupdate=TRUE): 0
# calls on mkf_searchtable:         0
# calls on mkf_insertkey:           0
# calls on mkf_deletekey:           0
# calls on mkf_merge:               0
# B-tree table merges:              0
# B-tree table splits:              0
total blocks in data files:         43008
number of non-virgin blocks in data files: 2195
number of blocks in global empty block list: 0
```

No active transactions on this database.

```
***** end report *****
Thu Apr 12 16:36:12 2001
```

---

### **base data file name**

The base data file contains the description of the database. The name is defined in the database's ddl file.

### **db state**

The database control block contains the database state. The following table describes the legal states:

<b>Legal State</b>	<b>Description</b>
0	Database not completely initialized
1	Database shut down
2	Normal
3	Database is being restored
4	Database is being backed up off line
5	Fix up states

### **database fatal err**

A **yes** in this field indicates a fatal error occurred. Check for an event log entry and a core, dump, or Dr. Watson file, depending on your platform.

### **description valid**

A value of 1 means that the database description was successfully read from the base datafile. A value of 0 usually means that this has not happened yet.



**db is recovered**

A value of 1 means that crash recovery completed successfully during the initial opening of the database. A value of 0 means that crash recovery has not been run yet or has failed. A value of zero also indicates that crash recovery will be attempted again the next time the database is opened.

**abort mode**

For development use only. This mode helps exercise the deadlock handling code of FileNet server software.

The value should always be zero (0).

**xxx\_ilk**

The ilk entry displays hexadecimal value of the interlock handle. Most of the interlocks are used as locks. A few are used as resource-blocking interlocks.

**opendescpid**

Process ID of the process that executed an MKF\_opendescription call but has not yet executed MKF\_closedescription. MKF\_ddl is the primary caller of MKF\_opendescription.

**number of buffers**

The number of buffers in the database's buffer pool, as defined in the ddl file for the database. This parameter controls the main buffer pool of memory for the database. When writing to a block in the database, two buffers are consumed—one for the current copy of the block's

information and one for a copy of the buffer before the information was modified.

**num virgin buffers**

The number of buffers that have never been used.

**block list at**

Hexadecimal location of the block list (an array of blocks in the buffer pool) and size, in bytes, of each element in the array.

**num avail buffs**

The number of buffers in the available buffer list. These buffers were in use at one time but currently do not contain information.

**num bufs in mru ring**

The number of buffers in the overlayable (most recently used) buffer ring. These buffers contain valid information, but can be reclaimed; if reclaimed, these buffers can hold the contents of a database block not currently resident in the buffer pool. The overlayable ring is maintained in most-recently-used order, which facilitates choosing the least-recently-used buffer as the buffer to overlay. Overlaying the least-recently-used buffer helps minimize I/O to the database.

**tot olayable bufs**

The number of buffers in an available buffer list plus the number of buffers in the overlay ring.

**tot non-olayable bufs**

The number of buffers currently in use. These buffers cannot be overlaid at the present time (actually, the time the “snapshot” was taken by MKF\_debug).

**tot unmod data blks**

The number of unmodified data blocks in the buffer pool.

**&acttr**

Hexadecimal address of the array of active transaction structures, size, in bytes, of each element in the array, and number of elements in the array.

**&datafiledef**

Hexadecimal address of the array of data file definitions, size, in bytes, of each element in the array, and number of elements in the array.

**&datafileinx**

Hexadecimal address of the array of data file indexes into the global table of files, size, in bytes, of each element in the array, and number of elements in the array.

**&aijfiledef**

Hexadecimal address of the array of file definitions for the after-image journal (aij), size, in bytes, of each element in the array, and number of elements in the array.

**&aijfileinx**

Hexadecimal address of the array of aij file indexes into the global table of files, size, in bytes, of each element in the array, and number of elements in the array.

**&bijfiledef**

Hexadecimal address of the array of file definitions for the before-image journal (bij), size, in bytes, of each element in the array, and number of elements in the array.

**&bijfileinx**

Hexadecimal address of the array of bij file indexes into the global table of files, size, in bytes, of each element in the array, and number of elements in the array.

**blklisthead**

Hexadecimal address of the array of hash heads (the hash table is used to locate database blocks in the blocklist array) and the hexadecimal hash mask.

**max long trans cc**

The maximum value of the claim count for large transactions. This is the maximum number of transactions that can execute concurrently. A large transaction (also known as a “long” transaction) is one that requires more blocks than can fit in  $1/n$  of the buffer pool, where  $n$  is the maximum value of the transaction claim count. This value must be less than or equal to the claim count.

**long trans claim cnt**

The current value of the claim count for large transactions (also known as “long” transactions). If positive or zero (0), the value is the number of new large transactions that can start. If negative, the value is the number of processes waiting to begin a large transaction.

**max trans claim cnt**

The current value of the transaction claim count. This value represents the maximum number of transactions of either type (large or small) that can concurrently execute against the database. This value must be greater than or equal to the long transaction claim count.

**trans claim count**

The current value of the transaction claim count. If positive or zero (0), the value is the number of new transactions of either type (large or small) that can currently begin. If negative, the value is the number of processes waiting to begin transactions against the database. The maximum number of concurrent transactions is limited to maximize performance. If you allow more than an optimal number of concurrent transactions, buffer pool thrashing and/or excessive abort transactions occur.

**small→large trans**

The number of transactions that were successfully converted from small to large transactions.

**small→large tr fail**

The number of failed attempts to convert small transactions to large transactions.

**small→large tr err**

The number of small transactions that ultimately failed to convert to large transactions. Multiple attempts might be made to convert a small transaction to a large transaction. These attempts ultimately succeed or fail, with the number of failures noted in this field.

**getbuffer retries**

The number of retries to obtain the use of a buffer so that a new database block can be read into the buffer pool. The first attempt is not considered a retry.

**getbuffer errs**

The number of times a transaction failed to get a buffer to hold a database block.

**ctlblock\_p**

Hexadecimal pointer to a buffer containing the database control block.

**ctlblklock**

The interlock used as a lock for updating control block information.

**recdef**

Hexadecimal location of the row definitions (for example, table definitions) array.

**mbi demon pid**

Process ID of the writeahead (mbi) process. (**mbi** is “modified before-image.”)

**mbi demon enabled**

If the value is 0, the writeahead (mbi) demon is disabled. If 1, it is enabled.

**log\_updaij\_time**

If the value is 0, debugging information is not written to the system event log. If 1, debugging information (for example, elapsed time for updating the pointer to the beginning of the aij, number of buffers written, and so on) is written to the system event log.

**mbi\_heap at**

Hexadecimal address of the mbi heap.

**num\_heap**

Number of elements in the mbi heap.

**heap\_av at**

Hexadecimal address of the available mbi heap element array.

**num\_heap\_levels**

Number of levels in the mbi heap.

**heap\_avbv**

Bit vector for available elements in the mbi heap.

**num writeahead (ms)**

Most significant longword for the number of buffers written by the writeahead demon. The concatenation of the ms and ls longword is a 64-bit integer.

**num writeahead (ls)**

Least significant longword for the number of buffers written by the writeahead demon. The concatenation of the ms and ls longword is a 64-bit integer.

**num writeahead bursts**

Number of times (in decimal and hexadecimal notation) the writeahead daemon awoke to write a burst.

The writeahead daemon alternates between “sleep” and “awake” (active) states. A “burst” occurs when the daemon awakens, writes a number of buffers ahead, then returns to a sleep state.



## Checklist

Before using MKF\_debug, be aware of the following:

- You can be a member of the fnadmin, fnop, fndba, or fnusr groups to run MKF\_debug.
- You must run MKF\_debug on the server where the database resides.
- MKF\_debug takes a snapshot of MKF activity and therefore reports only events occurring at the time you start the program.
- You can safely run MKF\_debug any time. MKF\_debug cannot hang a database and MKF\_debug itself will not hang, even if one or more databases are hung.

## Procedure

No specific procedure is required.

## Related Topics

[“MKF\\_dump” on page 738](#)

[“MKF\\_tool” on page 801](#)

# MKF\_dump

## Description

MKF\_dump provides a combined hexadecimal and ASCII display of the contents of a file or partition. MKF\_dump can also display MKF database data files and recovery log files in a fully formatted or semi-formatted mode. MKF\_dump can also provide a display of the block headers in Enterprise Backup and Restore (EBR) backup files and MKF\_export output files. You can use MKF\_dump on any file. MKF\_dump has hexadecimal and ASCII editing capability.

MKF\_dump does not perform its functions through the MKF subsystem. Rather, MKF\_dump opens files directly by calling file open routines.

You can use MKF\_dump on any file.

## Use

An MKF **expert** can use MKF\_dump to view MKF databases and after-image files. The most common uses of MKF\_dump are to search for corruption in a database and to view the database control block.

You can view the files in formatted, semi-formatted, or hexadecimal plus ASCII format. The hexadecimal plus ASCII format consists of hexadecimal data block displayed on the left with equivalent ASCII characters displayed in a column on the right, as shown below:

---

BLOCK 230 (0x00000000000000E6): F:\fnsw\_db\tran\_db\_1k  
0x0000E6 01 (230): 0x0011: ai 17 of short trans  
trans id: 370 (0x172) aij bsn: 0x000000000000 (0)  
DATA BLOCK table: 1 numrecs: 59 gas: 0 Btree gas: 0 bv bytes: 8

row 0 (0x0):	251: F0000000 00000001 00000001 00000001		
row 1 (0x1):	247: F0000000 00000002 00000002 00000002		
row 2 (0x2):	243: F0000000 00000003 00000003 00000003		
row 3 (0x3):	239: F0000000 00000004 00000004 00000004		
row 4 (0x4):	235: F0000000 00000005 00000005 00000005		
row 5 (0x5):	231: F0000000 00000006 00000006 00000006		
row 6 (0x6):	227: F0000000 00000007 00000007 00000007		
row 7 (0x7):	223: F0000000 00000008 00000008 00000008		
row 8 (0x8):	219: F0000000 00000009 00000009 00000009		
row 9 (0x9):	215: F0000000 0000000A 0000000A 0000000A		
row 10 (0xA):	211: F0000000 0000000B 0000000B 0000000B		
row 11 (0xB):	207: F0000000 0000000C 0000000C 0000000C		
row 12 (0xC):	203: F0000000 0000000D 0000000D 0000000D		
row 13 (0xD):	199: F0000000 0000000E 0000000E 0000000E		
row 14 (0xE):	195: F0000000 0000000F 0000000F 0000000F		
row 15 (0xF):	191: F0000000 00000010 00000010 00000010		

---

```
row 16 (0x10):
  187: F0000000 00000011 00000011 00000011 |           |
row 17 (0x11):
  183: F0000000 00000012 00000012 00000012 |           |
row 18 (0x12):
  179: F0000000 00000013 00000013 00000013 |           |
row 19 (0x13):
  175: F0000000 00000014 00000014 00000014 |           |
row 20 (0x14):
  171: F0000000 00000015 00000015 00000015 |           |
row 21 (0x15):
  167: F0000000 00000016 00000016 00000016 |           |
row 22 (0x16):
  163: F0000000 00000017 00000017 00000017 |           |
row 23 (0x17):
  159: F0000000 00000018 00000018 00000018 |           |
row 24 (0x18):
  155: F0000000 00000019 00000019 00000019 |           |
row 25 (0x19):
  151: F0000000 0000001A 0000001A 0000001A |           |
row 26 (0x1A):
  147: F0000000 0000001B 0000001B 0000001B |           |
row 27 (0x1B):
  143: F0000000 0000001C 0000001C 0000001C |           |
row 28 (0x1C):
  139: F0000000 0000001D 0000001D 0000001D |           |
row 29 (0x1D):
  135: F0000000 0000001E 0000001E 0000001E |           |
row 30 (0x1E):
  131: F0000000 0000001F 0000001F 0000001F |           |
row 31 (0x1F):
  127: F0000000 00000020 00000020 00000020 |           |
row 32 (0x20):
  123: F0000000 00000021 00000021 00000021 |           !   !   !|
row 33 (0x21):
  119: F0000000 00000022 00000022 00000022 |           "   "   "|
row 34 (0x22):
```

---

115:	F0000000	00000023	00000023	00000023		#	#	#	
row 35	(0x23):								
111:	F0000000	00000024	00000024	00000024		\$	\$	\$	
row 36	(0x24):								
107:	F0000000	00000025	00000025	00000025		%	%	%	
row 37	(0x25):								
103:	F0000000	00000026	00000026	00000026		&	&	&	
row 38	(0x26):								
99:	F0000000	00000027	00000027	00000027		'	'	'	
row 39	(0x27):								
95:	F0000000	00000028	00000028	00000028		(	(	(	
row 40	(0x28):								
91:	F0000000	00000029	00000029	00000029		)	)	)	
row 41	(0x29):								
87:	F0000000	0000002A	0000002A	0000002A		*	*	*	
row 42	(0x2A):								
83:	F0000000	0000002B	0000002B	0000002B		+	+	+	
row 43	(0x2B):								
79:	F0000000	0000002C	0000002C	0000002C		,	,	,	
row 44	(0x2C):								
75:	F0000000	0000002D	0000002D	0000002D		-	-	-	
row 45	(0x2D):								
71:	F0000000	0000002E	0000002E	0000002E		.	.	.	
row 46	(0x2E):								
67:	F0000000	0000002F	0000002F	0000002F		/	/	/	
row 47	(0x2F):								
63:	F0000000	00000030	00000030	00000030		0	0	0	
row 48	(0x30):								
59:	F0000000	00000031	00000031	00000031		1	1	1	
row 49	(0x31):								
55:	F0000000	00000032	00000032	00000032		2	2	2	
row 50	(0x32):								
51:	F0000000	00000033	00000033	00000033		3	3	3	
row 51	(0x33):								
47:	F0000000	00000034	00000034	00000034		4	4	4	
row 52	(0x34):								
43:	F0000000	00000035	00000035	00000035		5	5	5	

---

```
row 53 (0x35):
    39: F0000000 00000036 00000036 00000036 |      6   6   6|
row 54 (0x36):
    35: F0000000 00000037 00000037 00000037 |      7   7   7|
row 55 (0x37):
    31: F0000000 00000038 00000038 00000038 |      8   8   8|
row 56 (0x38):
    27: F0000000 00000039 00000039 00000039 |      9   9   9|
row 57 (0x39):
    23: F0000000 0000003A 0000003A 0000003A |      :   :   :|
```

---

---

**Note** By default, only the first two rows and the last row of the block are drawn.

---

## Syntax

MKF\_dump <file> [... <file>]

**<file>** Specifies the first raw partition or file to be viewed.

## Commands

To start MKF\_dump, enter **MKF\_dump** followed by one or more file names at the system prompt.

**CAUTION** Before you can edit hexadecimal/ASCII data, you must know the password provided by your service representative. Manually editing data in hexadecimal is extremely dangerous to the integrity of the data.

When you start MKF\_dump, the MKF\_dump prompt follows a help message. Also, you must use the full path name to make the command work. In UNIX:

```
corona(root)/. 170 ) MKF_dump /fnsw/dev/1/permanent_db0  
Enter ?<cr> for help.  
MKF_dump: ?
```

In Windows:

```
corona(root)/. 170 ) D:\MKF_dump \FNSW\DEV\1\PERMANENT_DB0  
Enter ?<cr> for help.  
MKF_dump: ?
```

At the prompt, enter commands to view data and perform tasks. To display the available commands, enter a question mark (?) followed by Enter. The following list of commands and their descriptions displays:

```
q          quit this program (<control>-c is OK too)
?          display context dependent help information
?<letter> display detailed help for the specified command
!<shell cmd> escape to C-shell (/bin/csh)
.          (re)display current block
+          display next block
-          display previous block
<int>     display file-relative block <int>
          NOTE: All integers <int> may be decimal or hex.
          Hex integers are of the form 0x<hex digits>
+<int>    move forward <int> blocks and display block
-<int>    move backward <int> blocks and display block
f          control current displayable file. See ?f
e          control echo block display output to file. See ?e
#<comment> if echoing, output <comment> to echo file.
c          control the saved block search condition. See ?c
s          search for block satisfying a condition. See ?s
t          control whether block display twists longwords. See ?t
d          control assimilation of database description. See ?d
u          control whether block displays are unformatted. See ?u
w          control whether whole block is displayed. See ?w
p          control page size and file type
m          interactively modify current block
z          zero block(s) on disk
```

To display detailed help for an individual command, enter a question mark followed by the first letter of a command. Press the space bar at the prompt to page through multi-screen help messages.



---

MKF\_dump: ?s

Searches normally start with current block.

s+	search forward using current search condition. (See ?c)
s++	continue search forward. Same as s+ but omits current block
s-	search backward using current search condition. (See ?c)
s--	continue search backward. Same as s- but omits current block
s+b<int>	search forward for block with self addr <int>
s-b<int>	search backward for block with self addr <int>
s+a<int>	search forward for for block with aij bsn <int>
s-a<int>	search backward for for block with aij bsn <int>
s+c	search forward for corrupt block
s-c	search backward for corrupt block

Enter <SP> for more help, 'q' to quit:

The forms s+, s++, s-, s-- are used to start and continue searches using the current saved search condition. (See ?c) The other forms do not use the current search condition, and their parameter (i.e., block address or aij block sequence number) are not remembered for use in subsequent search commands. The search for a block address or aij bsn can also be accomplished with a saved search condition. The search for corruption, however, can not be accomplished in any other way. Note, however, that whenever a block is displayed as the result of any command (i.e., a search command or a direct display command), the program indicates whether the displayed block is corrupted or not.

---

To display the current setting of a command, enter the command followed by **Enter**.

---

The following examples show the help displayed with the question mark properly placed:

```
MKF_dump: ?t
t          show whether longwords are being twisted
t          t+ set (t- reset) twist-longwords on block display
MKF_dump: ?w
w          show if whole block is displayed
w+         w+ set (w- reset) display whole block
           NOTE: if w-, only block summary information is displayed
```

---

**Note** A ? placed after the command letter is ignored.

---

## Examples

Try MKF\_dump to become familiar with it. You can do no harm as long as you do not use the modify-a-block (m) or zero-a-block (z) commands. Use of these editing commands requires a password.

MKF\_dump uses the concept of a current block and a current file. Initially, you are positioned at block 0 of file 0 (the first parameter). To move to the next block, enter a plus (+). Enter a minus (–) to go to the previous block. To move to a specific block, either forward or backward from the current block, type a +<int> or –<int>, where <int> is the integer for the block number you want to examine.

To move to block n of the file, type n, where the value of n is either a decimal block number or a hexadecimal block number. Hexadecimal block numbers start with 0x. For example, to display block 100, enter 100 followed by a carriage return or 0x64 followed by a carriage return.

In the sample below, MKF\_dump displays Block 1 of the transient database on a UNIX system. (Blocks are the base datafile if an MKF database is the database control block.) Use the options from the menu bar at the bottom of each screen to move through the displays.

```
MKF_dump: 1

BLOCK 1 (0x000001): /fnsw/dev/1/transient_db0
  0x000001 06 (1): 0x0000: ai 0 of short trans
  trans id: 0 (0x0)   aij bsn: 0x00000000000000
DESCRIPTION BLOCK -- DATABASE CONTROL BLOCK
  formatlevel      3
  epoch            15
  epochchanging    0
  datafilenum      0
  dbstate          2: normal
  number of buffers 256
  max short trans  2
  max long trans   1
  cur num rec types 17
  max num rec types 20
  max cols. per rec 32
  cur num data files 1
  cur num aij files  1
  cur num bij files  0
  addr data file defns 0x00000200 (2)
  addr aij file defns  0x00000300 (3)
  addr bij file defns  0x00000400 (4)
  addr table defns     0x00000500 (5)
<sp>=next screen  d=down  u=up  b=goto begin  e=goto end  q=quit
```

```
addr first data blk  0x00002D00 (45)
  empty blk list head 0x00002D00 (45)
  addr root by-size   0x00002F00 (47)
  addr root by-loc    0x00003000 (48)
  cur aij file num    0
  cur-file-rel aij blk 293
  most sig aij bsn    0x00000000 (0)
  least sig aij bsn   0x0126 (294)
  next avail trans id 0x00005658 (22104)
  aij update freq     8000 blocks
  bij partition size  1000 blocks
  debug dk write lim  2147483647
  dump lock           0
  target station name DocServer
  ai exist            1
  continuitydts       0x2D0E0188
  checksumstate       2: csum_on
  read_after_write    1: set
  bi exist            0x00000000
  no bij              true
  use block 0         1
  olb_bsn_ms          0x00000000 (0)
  olb_bsn_ls          0x0000 (0)
<sp>=next screen    d=down  u=up   b=goto begin  e=goto end  q=quit
```

Using the keys indicated in the options bar at the bottom of each display, you can move up and down from your current position in the data display, or go to the beginning or end of the displayed data. CR moves down one line in the display. To end this display and return to the MKF\_dump prompt, press the q key.

**Note** To properly set the display on a Windows system you must right click on the screen and select Properties: From there do the following:

Layout --> Screen Buffer size --> Height - 25  
Options --> Command History --> number of buffers - 1

You can also select different data display formats. Use the **u** command to select an unformatted display or to reset an unformatted display back to formatted. As shown below, enter **u+** to display unformatted hexadecimal data for block 230. To reset the display to a formatted view of block 177, enter **u-**.

```
MKF_dump: u+
MKF_dump: 230
```

```
BLOCK 230 (0x0000000000000000E6): F:\fns_w_db\tran_db_1k
  0: 0000E601 00000172 00000000 00000011 |          r          |
  4: 013B0800 00000000 00000000 0000003F | ;          ?          |
 252: 00000001 00000001 00000001 8E2CB845 |          , E          |
```

```
BLOCK 230 (0x0000000000000000E6): F:\fns_w_db\tran_db_1k
  0: 0000E601 00000172 00000000 00000011 |          r          |
  4: 013B0800 00000000 00000000 0000003F | ;          ?          |
 252: 00000001 00000001 00000001 8E2CB845 |          , E          |
```

```
BLOCK 230 (0x0000000000000000E6): F:\fns_w_db\tran_db_1k
  0: 0000E601 00000172 00000000 00000011 |          r          |
  4: 013B0800 00000000 00000000 0000003F | ;          ?          |
  8: FBF7F3EF EBE7E3DF DBD7D3CF CBC7C3BF |          |
 12: BBB7B3AF ABA7A39F 9B97938F 8B87837F |          |
 16: 7B77736F 6B67635F 5B57534F 4B47433F | {wsokgc_[WSOKGC? |
 20: 3B37332F 2B27231F 1B171313 F0000000 | ;73/+'#          |
 24: 0000003A 0000003A 0000003A F0000000 | : : :          |
 28: 00000039 00000039 00000039 F0000000 | 9 9 9          |
```

32:	00000038	00000038	00000038	F0000000		8	8	8	
36:	00000037	00000037	00000037	F0000000		7	7	7	
40:	00000036	00000036	00000036	F0000000		6	6	6	
44:	00000035	00000035	00000035	F0000000		5	5	5	
48:	00000034	00000034	00000034	F0000000		4	4	4	
52:	00000033	00000033	00000033	F0000000		3	3	3	
56:	00000032	00000032	00000032	F0000000		2	2	2	
60:	00000031	00000031	00000031	F0000000		1	1	1	
64:	00000030	00000030	00000030	F0000000		0	0	0	
68:	0000002F	0000002F	0000002F	F0000000		/	/	/	
72:	0000002E	0000002E	0000002E	F0000000		.	.	.	
76:	0000002D	0000002D	0000002D	F0000000		-	-	-	
80:	0000002C	0000002C	0000002C	F0000000		,	,	,	
84:	0000002B	0000002B	0000002B	F0000000		+	+	+	
88:	0000002A	0000002A	0000002A	F0000000		*	*	*	
92:	00000029	00000029	00000029	F0000000		)	)	)	
96:	00000028	00000028	00000028	F0000000		(	(	(	
100:	00000027	00000027	00000027	F0000000		'	'	'	
104:	00000026	00000026	00000026	F0000000		&	&	&	
108:	00000025	00000025	00000025	F0000000		%	%	%	
112:	00000024	00000024	00000024	F0000000		\$	\$	\$	
116:	00000023	00000023	00000023	F0000000		#	#	#	
120:	00000022	00000022	00000022	F0000000		"	"	"	
124:	00000021	00000021	00000021	F0000000		!	!	!	
128:	00000020	00000020	00000020	F0000000					
132:	0000001F	0000001F	0000001F	F0000000					
136:	0000001E	0000001E	0000001E	F0000000					
140:	0000001D	0000001D	0000001D	F0000000					
144:	0000001C	0000001C	0000001C	F0000000					
148:	0000001B	0000001B	0000001B	F0000000					
152:	0000001A	0000001A	0000001A	F0000000					
156:	00000019	00000019	00000019	F0000000					
160:	00000018	00000018	00000018	F0000000					
164:	00000017	00000017	00000017	F0000000					
168:	00000016	00000016	00000016	F0000000					
172:	00000015	00000015	00000015	F0000000					
176:	00000014	00000014	00000014	F0000000					

---

180:	00000013	00000013	00000013	F0000000		
184:	00000012	00000012	00000012	F0000000		
188:	00000011	00000011	00000011	F0000000		
192:	00000010	00000010	00000010	F0000000		
196:	0000000F	0000000F	0000000F	F0000000		
200:	0000000E	0000000E	0000000E	F0000000		
204:	0000000D	0000000D	0000000D	F0000000		
208:	0000000C	0000000C	0000000C	F0000000		
212:	0000000B	0000000B	0000000B	F0000000		
216:	0000000A	0000000A	0000000A	F0000000		
220:	00000009	00000009	00000009	F0000000		
224:	00000008	00000008	00000008	F0000000		
228:	00000007	00000007	00000007	F0000000		
232:	00000006	00000006	00000006	F0000000		
236:	00000005	00000005	00000005	F0000000		
240:	00000004	00000004	00000004	F0000000		
244:	00000003	00000003	00000003	F0000000		
248:	00000002	00000002	00000002	F0000000		
252:	00000001	00000001	00000001	8E2CB845		, E

---



## Checklist

Before you use MKF\_dump, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_dump.
- You should not attempt to modify database information unless you are an MKF expert.
- To modify database information, you must first obtain an authorization password from your service representative.

## Procedure

The following procedure is based on a scenario in which the system is running slowly even after turning off the **read after write** option for the database. We want to determine if the database has been updated with our request to disable the read\_after\_write option.

Perform the following steps:

- 1 Run MKF\_dump on the transient database.

```
corona(root)/.> MKF_dump /fnsw/dev/1/transient_db0
Enter ?<cr> for help

MKF_dump: ?
```

```
corona(root)/.> MKF_dump D:\FNSW\DEV\1\TRANSIENT_DB0
Enter ?<cr> for help

MKF_dump: ?
```

- 2 Enter a **w+** at the command prompt to display whole blocks of data, then verify the correct mode has been set with the **w?** command.

```
MKF_dump: w+
MKF_dump: w?
full-block-mode = 1
```

- 3 Enter a plus sign (+) at the prompt to advance to and display block 1 of the database. Block 1 is the database control block.

```
MKF_dump: +
```

- 4 Examine pertinent information in the control block, such as:

- Does the block contain the correct base data file name?  
(/fnswh/dev/1/transient\_db0)
- Is the state of the database normal?  
(dbstate 2: normal)
- Is read\_after\_write disabled?  
(read\_after\_write 1: set)

Use the following sample output as a reference.

---

```
BLOCK 1 (0x000001): /fnsw/dev/1/transient_db0
  0x000001 06 (1): 0x0000: ai 0 of short trans
  trans id: 0 (0x0)   aij bsn: 0x00000000000000
DESCRIPTION BLOCK -- DATABASE CONTROL BLOCK
  formatlevel          3
  epoch                78
  epochchanging        0
  datafilenum          0
  dbstate              2: normal
  number of buffers    64
  max short trans      3
  max long trans       1
  cur num rec types    10
  max num rec types    15
  max cols. per rec    24
  cur num data files   1
  cur num aij files    1
  cur num bij files    0
  debug dk write lim   2147483647
  dump lock            0
  target station name  DocServer
  ai exist              1
  continuitydts        2a2d68be
  checksumstate        2: csum_on
  read_after_write    1: set
  bi exist              0x00000000
  no bij                true
  use block 0          1
```

---

## 5 Enter **q** to exit MKF\_dump and reply to the confirmation prompt:

```
MKF_dump: q
```

```
Enter y<cr> to confirm you wish to terminate this
program: y
```

---

# MKF\_export

## Description

MKF\_export scans the data in an MKF database and copies the data to a magnetic disk file or magnetic tape.

MKF\_export first shuts down the specified MKF database. The database remains shut down for the duration of the export.

You cannot resume a terminated MKF\_export job from the point at which it terminated. If an MKF\_export job is killed or terminates abnormally during the export, you must rerun the export from the beginning.

---

### Tip

If MKF\_export is killed or terminates abnormally, you might need to run MKF\_startup to restart the database.

---

To enhance performance, MKF\_export skips over blocks and rows that seem to have problems. It writes notification messages of skipped blocks or rows to the system event log. Occasionally some skipped data could be lost. At the completion of an MKF\_export job, check the system event logs for messages related to skipped blocks or rows.

MKF\_export achieves high performance by reading the MKF database sequentially with large multi-sector I/Os and by ignoring all blocks except data blocks. For data blocks, MKF\_export sequentially exports each valid row.

MKF\_export writes 512 KB blocks to the output file, starting with the database description, followed by all the rows of the database.

The output blocks have a block sequence number and a checksum. This enables MKF\_import to tell if any blocks are recovered, missing,

or duplicated. There are ECC blocks in the output, so that MKF\_import can tolerate a certain degree of I/O errors or corrupted data by reconstructing the blocks in error. This helps make the import/export process robust. The data in the output file is compressed. The exact size of the output file depends on the database content. As a rule of thumb, the output file will probably be one half to one third the size of the In-use portion of the database.

## Use

MKF\_export scans the data in an MKF database (NCH, security, transient, or permanent) and copies it to a magnetic disk file or magnetic tape.

Use MKF\_export (followed by MKF\_import) to transport all or some data from one MKF database to another. For example, you can use MKF\_export and MKF\_import to move MKF database data to a different platform or when making a major configuration.

## Syntax

MKF\_export <basefile> {out=<disk filename>|out=<tape parameters>}

**basefile** The full path name of the base file of the MKF database (for example, **/fnsw/dev/1/permanent\_db0** for UNIX, or **D:\FNSWDEV\1\PERMANENT\_DB0** for Windows) that is to be exported. All the data in the database is exported. MKF\_export makes no provision for specifying that less than the entire database should be exported.

**out** Specifies whether the data exported is sent to a disk file or a local tape device. The out options are:

{out=<disk filename>|out=tape <tape parameters>}

**out=<disk filename>**

---

**Note** <tape parameters> is not allowed with this option.

---

This must be the full pathname of the disk file to be created for the output data. The output file can also be a UNIX raw partition, which will increase throughput drastically over magnetic tape or ordinary disk files.

**out=tape <tape parameters> ::= <type> <device>**

This causes the output to go to magnetic tape. It is assumed that the tape drive is local to the host when MKF\_export is running. If the exported data does not fit on one tape, the tape will be rewound, and MKF\_export will ask the user to mount another tape. This will be repeated until all the row data has been exported, no matter how many tapes are necessary. Care should be taken when marking the tapes,

because MKF\_import will ask the user to mount the tapes in the same order as they were written.

**<type> ::= type= {reel | QIC | 8mm | 4mm | IBM348 | dlt}**

Type of tape device, as described in the table below:

Type	Description
<b>reel</b>	9 track reel-to-reel tape (the default)
<b>QIC</b>	1/4-inch cartridge (QIC) tape
<b>8mm</b>	8mm cartridge tape
<b>4mm</b>	DAT (4mm) cartridge tape
<b>IBM3480</b>	IBM 3480 cartridge tape
<b>dlt</b>	Digital linear tape

**<device> ::= device=<special device file>**

The full path name of the tape device that rewinds the tape when the device is closed.

UNIX example: **device=/dev/tape1.**

Windows example: **device=TAPE0.**

## Sample Output

Displays one line every four seconds with status indication how many seconds have passed and estimating how many seconds remain along with a percentage of rows:

```
MKF_export begin: Thu Apr 12 17:47:20 2001

5 secs., 21 secs. remain, 465,431 rows, 19%
10 secs., 21 secs. remain, 1,168,512 rows, 32%
15 secs., 18 secs. remain, 1,830,431 rows, 45%
20 secs., 14 secs. remain, 2,533,953 rows, 58%
25 secs., 11 secs. remain, 3,076,452 rows, 69%

Rows exported:      3,076,452
Blocks output:      73
Bytes output:       38,273,024
Total elapsed time: 29 seconds
MKF_export F:\fnsw_db\permanent_db0:
    Successful completion. (This is not an error.)
MKF_export end: Thu Apr 12 17:47:49 2001
```



## Examples

### General Examples (UNIX)

```
MKF_export /fnsw/dev/1/permanent_db0 out=/tmp/perm.edata
```

```
MKF_export /fnsw/dev/1/permanent_db0 out=tape type=4mm device=/dev/tape1
```

### General Examples (Windows)

```
MKF_export D:\FNSW\DEV\1\PERMANENT_DB0 OUT=D:\TMP\PERM.EDATA
```

```
MKF_export D:\FNSW\DEV\1\PERMANENT_DB0 OUT=TAPE  
TYPE=4MM DEVICE=TAPE0
```

## Checklist

Before using MKF\_export, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_export.
- At MKF\_export startup, the database shuts down automatically.
- If MKF\_export processing is interrupted or aborted, you cannot resume the operation at the point it terminated. If a program or system interruption disrupts MKF\_export completion, restart the export from the beginning.
- MKF\_export shuts down the database during the export. If you issue the kill command to terminate the program, if the program terminates abnormally, or if the system fails, you could have to restart the database. (MKF\_startup can be used to start the database.) If MKF\_export completes normally, restarting the database is unnecessary.
- MKF\_export always exports all data in the database. MKF\_export does not perform partial exports of data.
- When MKF\_export completes, check the system event log for skipped block or row messages.

## Procedure

- 1** Determine the base name of the MKF database you want to export to disk or tape.
- 2** Verify that the database is not being used before you start MKF\_export. (MKF\_export requests that the FileNet software be shutdown.)
- 3** Run the MKF\_export command with appropriate options.  
  
Use the basefile name of the database you want to export. Remember to specify out=tape if exporting to a tape drive.
- 4** When the procedure completes, check the system event log for errors.

## Related Topics

[\*\*“MKF\\_import” on page 770\*\*](#)

# MKF\_fixup

## Description

MKF\_fixup attempts an in-place recovery of an MKF database and is designed to maximize the probability of executing to completion. MKF\_fixup recovers any sort of corruption to the **data** portion of the MKF database, minimizing loss of user data. However, it is not intended to recover corrupted **description information** in the MKF database. (Other features, not described here, provide recovery of this information.) In fact, MKF\_fixup depends on this description information being correct.

MKF\_fixup has four distinct phases:

- Inspect/coalesce
- Sort
- Merge
- Zero/Wrapup

MKF\_fixup performs the following tasks:

- Assimilates the database description.
- Makes an initial pass through all the data files.
- Counts the number of rows and data blocks of each table.
- Checks the validity of every data block numerous times.
- Collects the data blocks for all tables, orders them by table number, and condenses them to the very beginning of the data files.

- Scans the rows of data blocks for the current table and extracts the key values for the current key. Sorts the keys into ascending order, and generates the leaf level btree blocks. Generates all the higher level btree blocks, thereby regenerating the entire btree for the current key. Repeats these steps for each table.
- Zeroes the blocks in data files beyond the new high-water mark.
- Zeroes all bij and aij files.
- At completion, leaves the database shut down so you do an off line backup immediately.

MKF\_fixup periodically writes restart information to disk. If interrupted and restarted, MKF\_fixup restarts at its last restart point. The restart points are as follows:

- The last disk write in the inspect/coalesce phase.
- The beginning of the sort phase.
- The beginning of the zero/wrapup phase.

## Use

Use MKF\_fixup in either one of the following two cases:

- As the final step of a fast initial load. An example of this case is a database format conversion (for example, changing the database block size). In this case, the original database is backed up and exported with MKF\_export. Then the database is redefined using fn\_edit and initialized by running MKF\_ddl with the **-initialize** option. Finally, MKF\_import is run with the **init** option. At the end of the import, MKF\_import will automatically run MKF\_fixup.
- When data in the database has been corrupted in such a way that it cannot be recovered using normal MKF recovery/rollforward mechanisms. An example of this case is when there are bad disk blocks and no backup was done, or all of the backup tapes are unavailable. In this case, MKF\_fixup is run to repair the database as best it can.

---

### CAUTION

Do not run MKF\_fixup without the consent and participation of your service representative. MKF\_fixup is complex and can execute for long periods of time.

---

MKF\_fixup is not intended for use in normal situations. (Crash recovery is automatic. Disk head crash recovery is done using backup dumps and rolling the database forward.) Use MKF\_fixup when:

- The database is corrupted by some means (how doesn't matter—hardware, software, firmware, operator, or other).
- The corruption has entered the backup dumps, so that a rollforward would restore corrupted data.

MKF\_fixup is a last resort (assuming rescanning and re-indexing of all documents is impractical) to be used only when you must make do with whatever data you currently have in your database. A few rows (those rows that were in corrupted data blocks) will likely be lost in the process.

We recommend that you run utilities such as MKF\_verify to check the invariant conditions of the database and to check for lost rows after MKF\_fixup completes. Recovering lost information must be done on a special case basis.

## Syntax

```
MKF_fixup <base_data_file_name>
```

**base\_data\_file\_  
name**

The full path name of the base data file of the MKF database to be fixed. (UNIX: The first character must be a */*. Windows: must start with **<Drive letter>:\.**) This is the only required parameter and is normally the only parameter specified. Examples of <basefile> specifications for the permanent and transient database partitions in a UNIX environment are **/fnsw/dev/1/permanent\_db0** and **/fnsw/dev/1/transient\_db0**. In a Windows environment they are: **D:\FNSWDEV\1\PERMANENT\_DB0** and **D:\FNSWDEV\1\TRANSIENT\_DB0**

## Examples

The following command example runs MKF\_fixup against the permanent database:

```
MKF_fixup /fnsw/dev/1/permanent_db0 (UNIX)
```

```
MKF_fixup D:\FNSWDEV\1\PERMANANET_DB0 (Windows)
```

## Checklist

Before you use MKF\_fixup, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_fixup.
- MKF\_fixup must be run on the host where the base data file resides.
- The database and shared memory must be shutdown before MKF\_fixup starts.

### **killfnsw -A -D -y**

If you have trouble shutting the database down, a qualified support person must be contacted to take special measures to shut it down.

- You should run MKF\_fixup only at the direction of your service representative and only when all other attempts to save a database have been unsuccessful.
- You should back up the MKF database before running MKF\_fixup.
- Once MKF\_fixup begins, it must finish. You cannot use the FileNet software until MKF\_fixup completes.
- If interrupted, MKF\_fixup can be restarted. To restart, run MKF\_fixup again from the current directory with the same parameters.
- You must restore a full backup of the database if for any reason MKF\_fixup does not complete.



- MKF\_fixup saves only salvageable data. Some user data could be lost.
- After MKF\_fixup completes, a full backup should be run.
- Backup the database immediately after MKF\_fixup completes.

## Procedure

Contact your service representative for a procedure that is specific to your site needs.

## Related Topics

None.

# MKF\_import

## Description

MKF\_import reads the data produced by MKF\_export and stores the data in an MKF database. It can read the data from a magnetic disk file or magnetic tape. The export data can be from any supported platform (AIX, HP-UX, Solaris, Windows 2000).

A description of MKF\_import needs to be discussed as two separate cases: **add** and **init**. The **add** option adds data using manual database inserts. The **init** option is only allowed on freshly initialized databases and it for fast initial load of the database.

### The add Option

In contrast to MKF\_export, MKF\_import does not shut down the database. If you attempt to run MKF\_import while the database is shut down, MKF\_import stops until the database is restarted.

MKF\_import alters the MKF database. If the system crashes or the program does not complete for any reason, you re-run the **add** option and it will data from the point it left off.

With the **<rs>** parameter, you assign a restart file name for the restart file (for example, **rs=import.rs**) and is only used with the Add option. When initially run, MKF\_import creates this file, and stores restart information in it (including any modifications that were specified by the **<mods>** parameter). When MKF\_import completes normally, it deletes the restart file. If the restart file is already present when MKF\_import starts up, MKF\_import uses it to resume at the point where it left off.

**Tip** Magnetic tapes written on one platform might not be readable on a different platform due to vendor incompatibility. If that is the case, you will want to export to a disk file, use ftp to get the export file to the destination system, and import from the copy on the destination system.

---

You can make dynamic, or “on-the-fly,” modifications to the exported data with the **<mods>** parameter. If the ddl descriptions for the target and source databases are identical, the program imports all rows. However, if they differ, it checks the parameters in **<mods>**. If the file specified by **<mods>** is absent, it is a fatal error.

If **mods=prompt** is present, the program prompts you interactively for certain options that you select from a set of menus. If **mods=<file-name>** is present, the MKF\_import checks the specified ASCII text file for the modification options to be performed.

The modifications allowed are to omit importing specified tables.

### The init Option

The **init** option is for fast initial load of a freshly initialized database. This option is much faster than the **add** option. The **init** option does not use normal database inserts. Instead, after the import is finished, it automatically runs **MKF\_fixup** to generate B-trees.

When **init** starts, the database could be started up or shut down. When **MKF\_fixup** finishes, the database will be in the shutdown state.

## Use

Use MKF\_import to insert the data gathered by MKF\_export into an MKF database. You can optionally omit importing specified tables.

## Syntax

```
MKF_import <basefile> {init | add <restart>}
      {in=<filename>|in=tape <tape_params>} [ <mods> ]
```

The first parameter must be <basefile>. You can specify all other parameters in any order.

---

**Note** Do not include blank spaces within MKF\_import parameters.

---

**<basefile>** The full path name of the base file of the MKF database (for example, **/fnswh/dev/1/permanent\_db0** that is to be imported on UNIX or **D:\FNSWDEV\1\PERMANENT\_DB0** on Windows).

**init|add** **init**

If **init** is specified, the target database must be a freshly initialized MKF database. The only way to create a freshly initialized MKF database is to manually run **killfnswh -A -D -y** and then run **MKF\_ddl <DDL text file> -initialize**.

---

**Note** A restart file is not allowed with the **init** option.

---

**add <restart>**

If **add** is specified, the target database could be a new or existing MKF database. Importing the data will incrementally add rows to the data-

---

base using normal database inserts. It is your responsibility to prevent duplicate key errors by pre-planning. The **<restart>** parameter is:

rs=<full path name of restart file>

This specifies the name of the restart file.

---

**Note** When you use the add option, you **MUST** use the mods option also to suppress all of the duplicate tables.

---

**input** Location from which you are importing rows, that is, the source of the input. Specify input in the format:

in=<filename>|in=tape <tape\_params>

- If the input is in a magnetic disk file, the input parameter format is:

in=<filename>

It must be the full path name of the input file.

- If the input is from magnetic tape, the input parameter format is:

in=tape <tape\_params>

**Tape Parameters** Specifies the tape parameters when you choose tape as the input source (in=tape). The tape parameters **<tape\_params>** are **<type>** **<device>**

---

**Note** It is assumed that the tape drive is local to the host when MKF\_import is running.

---

The **<type>** parameter is **dev={reel | QIC | 8mm | 4mm | IBM348 | dlt}**

Type of tape device, as described in the table below:

Type	Description
<b>reel</b>	9 track reel-to-reel tape (the default)
<b>QIC</b>	1/4-inch cartridge (QIC) tape
<b>8mm</b>	8mm cartridge tape
<b>4mm</b>	DAT (4mm) cartridge tape
<b>IBM3480</b>	IBM 3480 cartridge tape
<b>dlt</b>	Digital linear tape

The **<device>** parameter is **device=<special device file>**

The full path name of the tape device that rewinds the tape when the device is closed.

UNIX example: **device=/dev/rmt0** (file name)

Windows example: **device=TAPE0**

**<mods>** Specifies the source of modifications to the database description. The format is:

`mods={prompt|<filename>}`

**prompt** causes the program to prompt for tables to suppress and it leads you to a multi-level, menu-driven program. At first, you will see the following menu:

```
Main menu for modifications to exported data:
 1 Proceed with import
 2 Suppress import of some tables
 3 Show tables exported
 4 Show tables in target database
 5 Terminate this program.
Enter choice number:
<MKF_import>
```

Option 1 - Allows you to go ahead with the import. Be aware that, by default, all tables will be imported.

Option 2 - Gives you another prompt to enter the table name you want to modify. Once you select a table the following menu choices display:

```
 1 Suppress import of table
 2 Unsuppress import of table
 3 Show columns of exported table
 4 Show columns of target table in target database.
 5 Show tables exported
 6 Show tables in target database
 7 Select another table.
 8 Exit menu.
Enter choice number:
<MKF_import>
```

Sub-Option 1 - You select this option if you do not want to import the table.

Sub-Option 2 - This option will only be selectable once you have suppressed that table. Use this option to place the suppressed table back on the list of tables to be imported.

Sub-Option 3 - Shows the columns of the exported table.

Sub-Option 4 - Shows the columns of the table you have selected.

Sub-Option 5 - Shows the current list of tables to be imported and whether they are suppressed or not.

Sub-Option 6 - Displays a list of tables in the database.

Sub-Option 7 - Allows you to change tables.

Sub-Option 8 - Returns to the previous menu level.

Option 3 - Shows the current list of the tables to be imported in that database. If you haven't suppressed any tables, all of the tables will appear in the list.

Option 4 - Shows a list of all the tables in that database. The *System Reference Guide* has a description for each of these tables. To download IBM FileNet documentation from the IBM support page, see [\*\*"Accessing IBM FileNet Documentation" on page 21.\*\*](#)

Option 5 - Quits the mods=prompt program.

**<filename>** is the name of the ASCII text file that contains the tables to suppress. When using the filename parameter, you need to create a file containing the names of the tables you want to suppress (for example, batch\_data). This file should contain nothing more than the



table name(s) followed by the word `suppress`, and should look similar to the following:

```
batch_data suppress
print_options suppress
```

## Examples

You can run `MKF_import` interactively. The program prompts you for parameters, as shown in the short example below.

```
corona(fnsw)/home/fnsw> MKF_import
MKF_import begin: Thu Jan 4 13:25:20 2001
Enter h for help message and quit, i for interactive mode: i
The restart file is created if not present, and is deleted
upon completion of the import. It holds restart information
in the event that the import is interrupted and must be
resumed.
Enter restart file name:   /home/fnsw/saveit.rs
```

To obtain detailed help for all parameters and options of `MKF_import`, enter `h` at the prompt as shown below:

```
$ MKF_import
MKF_import begin: Thu Jan 4 13:26:53 2001
Enter h for help message and quit, i for interactive mode: h
```

### General Examples (UNIX)

```
MKF_import /fnsw/dev/1/permanent_db0 init in=/tmp/perm.exp
```

```
MKF_import /fnsw/dev/1/permanent_db0 init in=tape type=4mm  
device=/dev/rmt0
```

### General Examples (Windows)

```
MKF_import D:\FNSW\DEV\1\PERMANENT_DB0 INIT IN=D:\TMP\  
PERM.EDATA
```

```
MKF_import D:\FNSW\DEV\1\PERMANENT_DB0 INIT IN=TAPE  
TYPE=4MM DEVICE=TAPE0
```

## Checklist

Before using MKF\_import, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_import.
- The database must be started but not in use.
- If you must restart MKF\_import, use exactly the same parameters that you originally used, especially the restart file parameter. To ensure using the same parameters, consider using a shell script to run MKF\_import.
- If you run MKF\_import with no parameters, the program prompts you to enter parameters. If you run MKF\_import with only the **help** parameter or with incorrect parameters, several screens of help information display.

- You must not include blank spaces in MKF\_import parameters. The first parameter must be <basefile>, but the others can be in any order.

## Procedure

Enter the MKF\_import command using information from [“Syntax” on page 772](#). Use the base file name of the database you want to import.

---

**Tip** Remember to specify **in=tape** if you import from a tape drive.

---

## Related Topics

[“MKF\\_fixup” on page 764](#)

[“MKF\\_export” on page 756](#)

*System Reference Guide*

To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

## MKF\_order

### Description

MKF\_order ensures that the byte ordering of an MKF database data file is the native byte ordering of the CPU on which it is currently running. It updates files and partitions in place, as necessary, to accomplish this. MKF\_order only reorders the blocks that need to be reordered.

Normally, you never use MKF\_order, except on a back up using EBR, and then restoring using the reconfigure\_onto option. MKF\_order is only used in unusual circumstances.

You must run this program on the **destination** system after transferring MKF database files to the destination system from a **source** system of different byte ordering. You can use a byte-oriented transfer program such as ftp to perform such transfers.

---

**Note** It is **very important** to shut down the MKF database on the **sending** system **before** you begin the transfer to the destination system. MKF\_order issues an error message if the MKF database is not shut down.

---

If the destination files and partitions have different names, sizes, starting offsets, or partition flags than those on the sending system, you must run MKF\_rename on the destination system **after** MKF\_order completes normally.

When an MKF database is shut down, the aij and bij contain no needed information. Therefore, you do not need to transfer the aij or bij to the destination system—transfer only the data files and partitions. You must manually create and zero the recovery log files and partitions on the destination system.

To create files (but not raw partitions), you can use the UNIX **touch** command. All MKF blocks, particularly aij and bij blocks on the destination system, must be initialized to zero. If you need to run MKF\_rename after MKF\_order completes, MKF\_rename performs that initialization.

You can safely run MKF\_order on an MKF database that already has the native byte ordering. You can also safely rerun MKF\_order if it does not finish the first time. In fact, MKF\_order can be safely run any time the database is not in use.

MKF\_order checks the checksum and self-address in each block. If incorrect, the byte ordering of the block is reversed and the checksum and self-address are rechecked. If correct, MKF\_order updates the block on disk. It reads updated blocks back on disk to ensure that the write operation completed correctly.

Since the checksum and self-address of every block are checked, upon normal termination there is extremely high confidence that the transfer from the sending system was accomplished without any undetected I/O errors, and that the database was assembled properly.

MKF\_order leaves the database in the shutdown state. You usually need to run MKF\_rename after MKF\_order completes. Finally, run MKF\_startup to enable normal processing against the database, and, if necessary, update the DDL source text file to reflect the new file layout. This is done by making changes in fn\_edit.

## Use

The MKF\_order program ensures that byte ordering of a database is appropriate for a given CPU. This in-place operation completely changes a database. It is **only** necessary to run MKF\_order after transferring an MKF database from a system with a CPU that has a byte order opposite of the byte ordering of the destination CPU.

## Syntax

```
MKF_order< file> [ -numblks ] [<file> [ -numblks ] ...
```

**<file>** Name of the MKF base data file or partition to be byte-ordered. You can specify up to eight files.

**-<numblks>** Number of 1024-byte blocks to be transformed. For files, all blocks are transformed by default. The -numblks parameter is unnecessary for files, but required for partitions.

---

**Note** You must include all data files of the database. However, do **not** include the aij (after-image journal), bij (before-image journal), and recovery log files.

---

## Example

The following example reorders 1,024 blocks (for 1024 byte blocks this is one megabyte) in the transient database partition:

```
MKF_order /dev/1/transient_db0 -1024 (UNIX)  
MKF_order D:\DEV\1\TRANSIENT_DB0 -1024 (Windows)
```

## Checklist

Before using MKF\_order, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_order.
- MKF\_order does an in-place byte reordering of the specified database and completely changes the database contents (if the byte ordering needs to be corrected).
- You must shut down the databases on the sending system before starting MKF\_order.
- You do not need to transfer the aij or bij recovery logs to the destination system.
- You must manually create and zero out the recovery log files or partitions on the destination system.
- It is usually necessary to run MKF\_rename after MKF\_order completes normally.

## Procedure

- 1 Create and initialize the recovery log files and partitions on the **destination** system.
- 2 Initialize all MKF blocks, particularly aij and bij blocks, on the destination system.

Running MKF\_ddl -initialize is usually the best method to do this.

- 3 Shut down the MKF databases on the **sending** system.
- 4 Transfer the MKF data files from the sending system to the destination system.
- 5 Run MKF\_order on the destination system.
- 6 When MKF\_order completes, run MKF\_rename, if necessary, on the destination system.
- 7 Restart the databases on the sending and destination systems.

## Related Topics

[\*\*“MKF\\_rename” on page 785\*\*](#)



# MKF\_rename

## Description

MKF\_rename performs the following functions:

- Modifies the blocks in the new base data file that contain the description of all files and partitions of the database.
- Modifies certain fields in the control block.
- Could modify fields in other blocks of the base data file (for example, the empty-block-list-head block).
- Zeros the blocks in the data files from the first virgin block to the end.
- Zeros all recovery log files.

MKF\_rename accesses all files directly and does not call MKF. The database must be shut down during MKF\_rename processing.

---

**Note** You don't use MKF\_rename to reconfigure a database on the same host. The normal way to transfer a database to another system is to use Enterprise Backup and Restore/**reconfigure into**. MKF\_rename is only used in unusual circumstances.

---

---

**Note** Data partitions should be on separate spindles from recovery log partitions whenever possible. However, before-image journal partitions could be on the same spindles as the data partitions.

---

## Use

Use MKF\_rename to rename, reconfigure, and copy the data and recovery files of an MKF database.

### CAUTION

---

If done incorrectly, reconfiguring your database in place could result in data loss. Call your service representative before you attempt to use MKF\_rename.

---

## Syntax

```
MKF_rename <infile>
```

### <infile>

The name of an ASCII file that contains the new file or partition names and descriptions.

The data in the input file <infile> is in three sections. The information in each section is line oriented. The first section is the data section, indicated by “data:” appearing on a line by itself. The next two sections are for recovery logs and before-image journals. These sections begin with a separate line containing “rl” or “bij:”, respectively.

### Note

---

The recovery log was formerly known as the after-image journal.

---

Blank lines are not allowed in the file. The lines in each section have four fields separated from each other by one or more blanks characters:

- The first field is the full path name of the file or partition (including drive letter, if any) and must begin with a forward slash character(/).
- The second field is the partition flag. Flag values are 0 for files and 1 for partitions.

- The third field is the zero-relative starting block. For files or partitions in the data section, this field must be zero. The block size of the databases was established when **MKF\_dll -initialize** was run. For more information, see [“MKF\\_ddl” on page 712](#). The block size cannot be changed.
- The fourth field is the number of blocks.

The following is an example of an input file that describes a database with three partitions:

```
data:
/dev/3/permanent_db0 1 0 1024
/dev/3/permanent_db1 1 0 2048
/dev/3/permanent_rl0 1 0 100000
bij:
/dev/3/permanent_db0 1 1024 1000
```

The first line describes the base data file, `/dev/3/permanent_db0`, which is 1024 blocks long. (Each MKF database block is 1024 bytes in size.) The 1 on this line indicates that this is a partition, not a file.

The second partition is `/dev/3/permanent_db1`. This partition holds the second part of the data and is 2048 blocks long.

The third partition is `/dev/3/permanent_rl0` and holds the recovery log. The partition is 100,000 blocks long.

The before-image journal starts at the end of the base data partition. The before-image journal is 1,000 blocks long.

## Checklist

Before you use MKF\_rename, be aware of the following

- You must be a member of the fnadmin or fnop group to run MKF\_rename.
- Reconfiguring your database in place could result in data loss or corruption if done incorrectly. Call your service representative before you run MKF\_rename.
- You should back up the old database before you reconfigure or rename it with MKF\_rename.
- You must first copy the data from the original database files or partitions into the new set of files or partitions. Use operating system copy utilities.
- You must shut down the database before starting MKF\_rename.
- MKF\_rename leaves the database in a shutdown state. When MKF\_rename completes, you must run MKF\_zeroajj to initialize the recovery log files then enable the database for normal processing. (You can use MKF\_startup, an appropriate fn\_util command, or recycle FileNet software to enable normal database processing).
- The database should be offline while copying the data and renaming the files.
- MKF\_rename does not copy any data from old files to the new files.

## Procedure

- 1 Call your service representative before you use MKF\_rename.
- 2 Back up all databases you want to reconfigure.
- 3 Verify that data from the original database files or partitions is in the new set of files or partitions.

Use operating system copy utilities such as UNIX dd or cp or the Windows Server copy command.

- 4 Use fn\_util command options to shut down the database.

Refer to fn\_util online help for command option descriptions.

- 5 Run MKF\_rename.

Specify the full path name of the file containing new file or partition descriptions.

- 6 When MKF\_rename completes, run MKF\_zeroaij to initialize the recovery log files.
- 7 Enable normal processing for the database.

## Related Topics

[“MKF\\_order” on page 780](#)

[“MKF\\_shutdown” on page 790](#) and [“MKF\\_startup” on page 793](#)

[“MKF\\_zerobij” on page 823](#)

Refer to online help for fn\_util command options.

# MKF\_shutdown

## Description

MKF\_shutdown shuts down an MKF database. Before shutting down the database, MKF\_shutdown waits for all current database transactions to complete and causes any future calls to the database to be suspended until the database is restarted. The shutdown and startup processes are transparent to programs using the database except for a delay while the database is shut down.

If the system fails while the database is shut down, the database remains shut down when you restart the system after the failure.

---

**Note** Normally, you don't need to run MKF\_shutdown because the FileNet software shuts down all of the databases when the software is stopped, or because EBR shuts down the MKF database if you want to make an offline backup of it.

---

## Use

You must run MKF\_shutdown locally.

Use MKF\_shutdown to shut down the MKF permanent, transient, or security databases. Normally, you don't need to run MKF\_shutdown because the FileNet software shuts down all the databases when the software is stopped, or because EBR shuts down an MKF database if you want to make a backup of it.

## Syntax

`MKF_shutdown <base_data_file>`

**<base\_data\_file>** Full path name of the base data file of the MKF database.

You can find this name in the ddl file for the database.

## Example

The following example shows the command to shut down the MKF transient database and the completion message displayed by MKF\_shutdown on a UNIX system:

```
corona> MKF_shutdown /fnsw/dev/1/transient_db0  
Database is shut down.
```

The following example shows the command to shut down the MKF transient database and the completion message displayed by MKF\_shutdown on a Windows Server system:

```
corona> MKF_shutdown D:\FNSW\DEV\1\TRANSIENT_DB0  
Database is shut down.
```

## Checklist

Before using MKF\_shutdown, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_shutdown.
- Before you run MKF\_shutdown, verify that Storage Library Control is not running.
- MKF\_shutdown must be run at the server on which the database resides.

## Procedure

No specific procedure is required. However, verify that Storage Library Control is inactive before you start MKF\_shutdown.

## Related Topics

**[“MKF\\_startup” on page 793](#)**

Refer to online help for related fn\_util command options.

See the *System Administrator’s Handbook* for a description of Storage Library Control. To download IBM FileNet documentation from the IBM support page, see **[“Accessing IBM FileNet Documentation” on page 21](#)**.



# MKF\_startup

## Description

When you run MKF\_startup, the MKF database that was stopped by the MKF\_shutdown program is released for normal processing. Any processes that were suspended in MKF resume with a maximum delay equal to their polling interval (currently 10 seconds).

## Use

You must run MKF\_startup locally.

Use MKF\_startup to start each permanent or transient database that has been shutdown.

## Syntax

```
MKF_startup <target_station> <base_data_file>
```

**<target\_station>** Identifies the name of the server on which the MKF database resides. You must run MKF\_startup locally.

**<base\_data\_file>** Identifies the full path name of the base data file of the MKF database. You can also find this name in the ddl file for the database.

---

**Tip** If the system seems hung, run **MKF\_debug -a** to check if there are shutdown databases. If so, running MKF\_startup on them will allow the systems to proceed.

---

## Example

The following example shows the command to start the MKF transient database and the completion message displayed by MKF\_startup on a UNIX system:

```
MKF_startup /fnsw/dev/1/transient_db0
Database started up.
```

The following example shows the command to start the MKF transient database and the completion message displayed by MKF\_startup on a Windows Server system:

```
MKF_startup D:\FNSW\DEV\1\TRANSIENT_DB0
Database started up.
```

## Checklist

- You must be a member of the fnadmin or fnop group to run MKF\_startup.
- Before using MKF\_startup, be aware that the program must run at the server on which the database resides.

## Procedure

No specific procedure is required.

## Related Topics

[\*\*“MKF\\_shutdown” on page 790\*\*](#)

Refer to online help for related fn\_util command options.

## MKF\_stats

### Description

MKF\_stats gathers and displays statistical information about how disk space is being used in an MKF database.

You can run MKF\_stats with the MKF database active or shut down. However, to obtain the most accurate information, the database should be shut down. MKF\_stats displays a warning message if the database is not shut down.

### Use

Use MKF\_stats to check the space available within a specified MKF database. You can also view the allocation of index space via a map.

You should run MKF\_stats once a week to monitor the size of your MKF databases, particularly the permanent database.

### Syntax

MKF\_stats <flag> <database>

**<flag>** Specifies which type of report to run. There are three possible flags, -q, -v, or -d. See [“Flags” on page 796](#) for further information.

**<database>** Identifies the full path name of an MKF database

**Flags**

- q** Displays a quick report containing total number of blocks, blocks used, and space used in the database.
- v** Displays a report and a view of the database by block type. Does not display blocks that have never been used but does include them in the statistics.
- d** Displays detailed statistics reports of block counts by type and percentages.

## Sample Output

The following examples use the MKF\_stats options to examine the MKF permanent database.

The following samples display reports on a Windows system. However, in the **-q** example, the database is active and MKF\_stats issues a warning message that the statistics might not be completely accurate:

---

```
TSG26(1):7> MKF_stats -q F:\FNSW\DEV\1\TRANSIENT_DB0
MKF statistics for F:\FNSW\DEV\1\TRANSIENT_DB0:
```

```
WARNING: F:\FNSW\DEV\1\TRANSIENT_DB0 is NOT shutdown
Statistics may not be up to date completely.
```

```
MKF statistics for F:\FNSW\DEV\1\TRANSIENT_DB0:
```

```
MKF statistics for F:\FNSW\DEV\1\TRANSIENT_DB1:
```

```
MKF statistics for F:\FNSW\DEV\1\TRANSIENT_DB2:
```

```
Max number of blocks in data base =                30,000
Virgin blocks in data base         =                 2,735
Nonvirgin blocks in data base      =                27,265    90%
```

---

---

```
root TSG26(1):8> MKF_stats -v F:\FNSW\DEV\1\TRANSIENT_DB0
MKF statistics for F:\FNSW\DEV\1\TRANSIENT_DB0:
```

```
Block types:
```

```
'D'=data      'd'=description      '?'=unknown
'I'=index     'A'=avsp directory    'r'=MKF_import restart
'_'=empty     'L'=avsp list head    'R'=reserved
'.'=virgin    'h'=empty list head
```

```
0x0000000000000000: .ddddddd dddddddd dddddddd dddddddd dddddddd ddddhdRI
0x0000000000000030: IIIIIIII IIIIIIII IIIIIIII IIIIIIII IIIIIIII IIIIIIII
0x0000000000000060: IIIIIIII IIIIIIII IIIIIIII IIIIIIII IIIIIIII IIIIIIII
0x0000000000000090: IIIIIIII IIIIIIII IIIIIIII IIIIIIII IIIIIIII IIIIIIII
0x00000000000000C0: IIIIIIII IIIIIIII IIIIIIII IIIIIIII IIII__ _D__D__
0x00000000000000F0: __D__ D__D__ __D__D__ __D__ __D__D__ __D__
0x0000000000000120: D__D__ __D__D__ __D__D__ __D__D__ __D__D__ DDDDD DDDDDDD
0x0000000000000150: DDDDIIDD DDDDDDD DDDDDDD DDDDDID DDDDDDD DDDDDDD
0x0000000000000180: DDDDDDD DIDDDDD DDDDDDI DDDDDDD DDDDDDD DDIDDDDD
0x00000000000001B0: IIDDDDD DDDDDDD DDDDDDD DDDDDDD DIDDDDD DDDDDDI
0x00000000000001E0: DIIIDD DDDDDDD DDDDDDD DDDDDID DDDDDDD DDIDDDDD
0x0000000000000210: DDDDDDD DDDDDDD DDDDDDD DDDDDID DDDDDDI IIDIIIII
0x0000000000000240: IIDIIDD DDDDDDD DDDDIIDD DDDDDDD DDDDDDD DDIDDDDD
0x0000000000000270: DDDDDDD DDD_DDD DDDDDDD DDDDDDD D
```

```
Max number of blocks in data base = 10,000
Virgin blocks in data base = 9,343
Nonvirgin blocks in data base = 657 6%
```

---

The following sample displays a detailed report of the MKF transient database on a Windows system:

```
TSG26(1):7> MKF_stats -d F:\FNSW\DEV\1\TRANSIENT_DB0
```

```
MKF statistics for F:\FNSW\DEV\1\TRANSIENT_DB0:
```

Block type	Count	% nonvirgin	% of Total
Empty	77	11	0
Data	316	48	3
Index	217	33	2
Description	44	6	0
Empty list head	1	0	0
Reserved	1	0	0
Restart import	0	0	0
=====	=====	=====	=====
Nonempty	580	88	5

```
Max number of blocks in data base = 10,000
Virgin blocks in data base = 9,343
Nonvirgin blocks in data base = 657 6%
```

## Checklist

Before using MKF\_stats, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_stats.
- You must run the program at the server that contains the database.
- You must use the full path name of the database, for example:  
  
/fnsw/dev/1/permanent\_db0 for UNIX platforms  
  
D:\FNFW\_LOC\DEV\1\PERMANENT\_DB0 for Windows platforms
- MKF\_stats can take a while to complete and display output to your display device, depending upon the size of the database.
- To obtain the most accurate information and to avoid degrading system performance, run MKF\_stats when the database is shut down.

## Procedure

No specific procedure is required.



---

## MKF\_tool

### Description

MKF\_tool provides commands for viewing and editing tables of FileNet Multi-Keyed File (MKF) databases (the permanent database, transient database, and security database).

### Use

Use MKF\_tool to view or edit tables of an MKF database.

You must run MKF\_tool locally.

### Syntax

```
MKF_tool [<station_name> <base_data_file>]
```

**<station\_name>** Identifies the name of the server on which the MKF database resides. You must run the MKF\_tool locally. Currently, "::" is the only valid value for the server, and signifies the local server.

**<base\_data\_file>** Identifies the full path name of the base data file of the MKF database.

If you do not enter a server name and base data file name, MKF\_tool opens the permanent and transient databases by default. If you need to open a specific database for examination, you must specify the local server name and base data file name for the database.

The following command opens the permanent database only:

The UNIX logo consists of the word "UNIX" in white, bold, sans-serif capital letters, centered within a red, parallelogram-shaped background.

```
MKF_tool :: /fnsw/dev/1/permanent_db0
```

**WIN****MKF\_tool :: D:\FNSWDEV\1\PERMANENT\_DB0**

When you enter MKF\_tool at the command line, the MKF\_tool> displays. Enter commands at this prompt. See [“Commands” on page 804](#) for command descriptions.

---

**Tip** MKF\_tool always opens the security database. If MKF\_tool doesn't want to come up, it could be because one of the databases it opens is shut down.

---

## Examples

The examples in this section include several commonly-used MKF\_tool commands and the select command.

### Commonly-Used Command Examples

The examples below illustrate some commonly-used MKF\_tool commands:

- Display online help for **all** MKF\_tool commands:

**help**

or for a specific command:

**help describe**

- List all the MKF database tables:

**desc**

- Display the attributes of the DOCS table:

### **desc docs**

- Change the last command given (desc docs) to **desc csm\_used\_space**:

**^docs^csm\_used\_space**

- Count the total number of documents on the Storage Library server:

**count docs \***

- Count the total number of write\_requests:

**count write\_request \***

### **Select Command Examples**

The manner in which you use the select command affects performance. Using wildcard characters (\*) could cause excessive execution time. Your system could **appear** to “hang.” For example, the following command could take a while to execute:

**select docs \* where doc\_id=1000000000**

The wildcard (\*) causes MKF\_tool to select every document in the DOCS table for IS checking. To obtain the information efficiently, use the following format, which takes only a few milliseconds to execute:

**select docs doc\_id=1000000000**

The following are typical select command examples:

- Display the current surface assigned to family ID 2:

**select family\_disk family\_id=2**

- Display the dynamic surface table relating to surface **3008**. This is updated, like other rows in this table, each time an image is written to the surface:

**select surf\_dyn\_info \* where surface\_id=3008**

- Display static information about a surface (such as the family ID, disk status, date labeled, and so on):

**select surf\_stat\_info \* where surface\_id=3008**

- Display the maximum document ID in DOCS:

**select docs doc\_id < 0xffffffff**

- Display a range of document IDs in DOCS:

**select docs 103303 < doc\_id < 103538**

## Commands

Many MKF\_tool commands are available to you. The following table identifies all commands, in alphabetical order, with a brief description of each. The list is followed by detailed explanations of each command.

Command	Description
ALLOWUPDATES	Establishes correct permissions for running certain commands. Requires a password.
COMPUTE	Displays the value of a given expression
COUNT	Displays the number of selected rows
DESCRIBE	Describes columns in a selected table

Command	Description
HARDCOPY	Places a copy of all MKF_tool output into a file
HEADERS	Acts as a toggle switch to write headers to the output device
HELP	Displays help information for MKF_tool commands
JOIN	Joins two tables into a single table
QUIT	Exits MKF_tool
SELECT	Displays database rows
TERMOFF	Turns off terminal output for commands
TERMON	Turns on terminal output for commands
TRANOPS	Sets the number of operations per MKF transaction
! <code>&lt;shell_cmd&gt;</code>	Executes a specified shell command
<code>^&lt;old_pattern&gt; ^&lt;new_pattern&gt;</code>	Executes the command formed by replacing <code>&lt;old_pattern&gt;</code> with <code>&lt;new_pattern&gt;</code> in a prior command

In the command descriptions that follow, replace items in angle brackets (< >) with an appropriate value. Items in square brackets ( [ ] ) are optional and items that you should enter **as is** are in UPPERCASE. All commands are case insensitive; therefore, you can enter any keyword or MKF column name in either uppercase or lowercase. You can enter some keywords in abbreviated form. You only need to enter the uppercase characters. For example, the describe command is listed as DESCRibe. To enter the abbreviated form of the command, enter **desc** as the command line.

You can break up long commands into multiple lines by suffixing all lines except the last with the backslash (\) character. As an alternative, you can type long commands as a single long line, up to 2000 characters.

Some commands have additional parameters that are described in [\*\*“Command Parameter Descriptions” on page 811.\*\*](#)

### **ALLOWUPDATES**

Before you can delete any data from a database table, you must use the ALLOWUPDATES command. To issue the ALLOWUPDATES command, log on a user that is a member of the fnadmin group. After entering ALLOWUPDATES, the program prompts you for a password (supplied by your service representative).

### **COMPUTE <expression>**

Displays the value of the given expression. In addition to constants, the expression could include variable names from the previously selected row.

### **COUNT <table\_name> <key\_cond> [WHERE <expression>]**

Displays a count of the number of selected rows.

### **DESCRibe [<table\_name>]**

Describes the columns in the specified table. If you don't specify a table name, this command lists all table names (without descriptions).

### **HARDcopy [<filename>]**

Sends MKF\_tool output to a file. If you specify a <filename>, this command creates a file using that name and copies all screen output to the that file. If you do not specify <filename>, it closes the previous output file (if it exists), displaying all information on the screen only.

---

## HEADERS

Toggles whether headers are written to the output device. Useful when generating files of data with no column headers or other nondata columns.

## HELP [<command>]

Displays help information. If you specify <command>, displays help for that command; otherwise displays help for all commands.

## JOIN <table1> TO <table2> AS <table3> BY [<column1>] <column2>

Joins <table1> and <table2>, creating <table3>. Each row in <table3> is a row from <table1> combined with a row from <table2>. If <column1> is not specified, <column2> is an MKF column that must exist in both tables, and must have the same type. If <column1> is specified, <column1> is an MKF column in <table1>, and <column2> is an MKF column in <table2>. Both columns must be of the same type. Note that this command does not change any data in the database—you can only view the data of <table1> and <table2> in a new manner.

You can perform any operation on the new table, including selects, deletes, updates, and inserts. If a row exists in <table1> but not in <table2> (or vice versa), attempts to select, delete, or update this row using the new table will fail, unable to locate it. Attempts to insert the row generates a “duplicates” error.

Restrictions: When accessing <table3>, the keys in <table2> other than <column2> are unusable and appear as ordinary data columns.

<column1> and <column2> must be the name of a unique nongroup key.

Example: When you run this program in the default manner, the following command automatically executes:

```
join surf_stat_info to surf_dyn_info as surf_info by surface_id
```

This join executes a select command such as:

```
select surf_info surface_id > 0 where family_id=1 and num_act_docs=0
```

---

**Note** family\_id is in the surf\_stat\_info table and num\_act\_docs is in the surf\_dyn\_info table. Before you can use the select command above, you must issue the join command to join the surf\_stat\_info and surf\_dyn\_info tables into a single table view.

---

## Quit

Terminates MKF\_tool.

**SElect <table\_name> <key\_cond> [WHERE <expression>] [SHOWING <column\_list>;]**

Displays rows in the database.



Care must be taken when using date and time formats as shown in the example below:

```

MKF_tool> sel csm_used_space * where created >="2003/02/25 11:25:54 AM"
cache_id.....1 ssn.....5769
obj_id.....100005 page.....0
maxlen.....136 curlen.....136
sec_len.....0 refcnt.....1
seqnum.....1 numchunk.....1
startary.....0x0100012f lenary.....1
created.....1046200434 => 2003/02/25 11:13:54
read.....0 => 1969/12/31 16:00:00
sec_info [0].....
client_attr [0]...2,0,0,0,0,0,0,0

```

When you formulate a select command, you really need to see what date/time format you are using first in your operating systems settings. If you use a format different than the one set for your system, you will get a failure.

```
Invalid datetime format
```

The date can be represented as either:

- integers
- strings (enclosed in double quotes)

If you are not sure of your format, just enter:

```
sel <tablename> *
```

and look for a line in the display similar to the following:

```
created.....1046200434 => 2003/02/25 11:13:54
```

### TERMOFF

Turns off terminal output, which can be desirable if writing large amounts of data to an output file via the HARDCOPY command.

### TERMON

Turns terminal output back on after being shut off with TERMOFF.

### TRANOPS <operations\_per\_transaction>

Sets the number of operations per MKF transaction in the delete and update commands. Setting this to a low value can prevent overflowing the before-image partition. If you set this to a high value, the program can perform an entire command in one transaction, thus guaranteeing that either all or none of the changes are made. The default is 100.

---

**Note** Transactions do not span commands, and therefore an “end of transaction” is done at the end of each command regardless of the setting of this parameter.

---

### !<shell\_cmd>

Executes the specified shell command.

### ^<old\_pattern>^<new\_pattern>

Executes the command formed by replacing <old\_pattern> with <new\_pattern> in the previous command. The characters “^^” repeat the last

---

command unchanged. The character “^” displays the last command, but does not execute it.

## Command Parameter Descriptions

The following are MKF command parameter descriptions.

**<table\_name>=<name>|.**

Where <name> is the name of a table. Use a period (.) to specify the last table referenced.

**<key\_cond>=[<key\_const><op>]<key\_name><op><key\_const>**

**<key\_cond>=\***

**<key\_cond>=<key\_name> in <filename>**

A <key\_cond> (key condition) of “\*” means display all columns in a table using the first defined key where the key value is greater than or equal to zero.

When key values are a list of constants in <filename>, use the format:

**<key\_name> in <filename>**

Specify one constant per line when using this format.

**<key\_const>=<exp>[:<key\_const>]**

One <exp> (mathematical expression) must exist in the <key\_const> (key constant) for each component of the key. Key components are items in a group key, and elements in any array except an ASCII array.

An ASCII array counts as one <exp>, and does not need to be entered as <char>:<char>:<char>.

**<exp>=a mathematical expression**

Same as <expression> except uses only constants.

**<expression>=a mathematical expression**

The following table describes the operators:

Operator	Function	Operator	Function
+	add	–	subtract
*	multiply	/	integer division
%	modulo	&	bitwise logical “and”
	bitwise inclusive “or”	~	ones complement
and	logical “and”	or	logical “or”
!	logical “not”	>	greater than
<	less than	>=	greater than or equal to
=	equal	<=	less than or equal to
!=	not equal		

You can use parentheses to override the precedence of operators.

Operands are:

- MKF column names
- String constants
- Character constants

- Hexadecimal constants
- Decimal constants
- The asterisk character ( \* )

You must type MKF column names as declared in the ddl. No abbreviations are allowed, but names are case insensitive. MKF column names that are non-ASCII arrays must be subscripted with [`<expression>`]. MKF variable names that are ASCII arrays might be subscripted. ASCII arrays that are not subscripted might only be compared against string constants or other ASCII arrays. columns of type `MKF_column_date` are either integers or date strings in the format **yyyy/mm/dd**. columns of type `MKF_column_datetime` are also defined as integers or date/time strings in the format **yyyy/mm/dd hh:mm:ss**.

String constants are surrounded with double quotes, and any pair of double quotes is treated as a single quote within the string.

Character constants are any character surrounded by single quotes. Only one character is allowed between the quotes, and it can be the single quote (') character.

Hexadecimal constants must be preceded with **0x**; all numeric constants not preceded with **0x** are assumed to be decimal.

The operand `'*` denotes a null MKF column and can be used in expressions of the form `"<mkf_column>=*"` or `"<mkf_column> !="`. The expression is TRUE or FALSE, depending on whether the MKF column is null or not.

---

**Note**

If an MKF column is null and used in any expression other than the above two, the MKF column is zero (0).

---

---

**<column\_list>=<column\_name> [<column\_list>]**

The <column\_list> indicates which columns in the row to display.

**<column\_name>=<mkf\_ddl\_name>[[ [ min | max | sum | avg ] (<column\_name> ) ]]**

Where <mkf\_ddl\_name> is the MKF ddl name of a column in a table, or the functions min, max, sum, or avg, followed by the column name surrounded by parentheses.

The functions give the minimum, maximum, sum, and average of the identified column for all rows selected. For the average function, null columns in selected rows are ignored and do not change the average.

## Checklist

Before you use MKF\_tool, be aware of the following:

- To use MKF\_tool, you must be logged on as a member of the fnadmin group or have a valid fnlogon session with SysAdmin privileges.
- You must run the program from a terminal attached to the server where the databases reside. MKF\_tool does not operate remotely.
- The database must **not** be shutdown and the security database must **not** be shutdown.
- If you want to modify any of the tables, contact your service representative for the password to the ALLOWUPDATES command. The password is changed monthly.

## Procedure

- 1 Start MKF\_tool.

Enter **MKF\_tool** at the command line of the server on which the databases reside. The MKF\_tool prompt appears:

```
<MKF_tool>
```

- 2 Enter commands at the MKF\_tool prompt.

# MKF\_verify

## Description

MKF\_verify validates the contents of any database. In the first phase of its operation, the MKF\_verify checks the Btrees against the data rows, verifying that the rows pointed to exist and have the correct key value. In the second phase, MKF\_verify checks the data rows against the btrees by extracting the key values and verifying that the keys exist in the Btrees and point back to the correct rows.

---

**Note** MKF\_verify can take a long time to run.

---

If MKF\_verify is interrupted during its processing, you can restart it by simply running MKF\_verify again and passing the name of the restore file.

## Use

MKF\_verify validates the contents of the permanent, transient or security database. Use it to locate corruption in the database. You must run MKF\_verify locally.

MKF\_verify is restartable. When you initially start MKF\_verify, use **restartfile=<file>** to create a new restart file. MKF\_verify reuses the restart file you specify to resume from the point at which it was interrupted. If you do not specify a restart file name, MKF\_verify uses a default restart file (**MKF\_verify.res**).



---

## Syntax

```
MKF_verify <targetstation> <basefile> [table=<tablename>]  
[phase=<n>] [restartfile=<file>] > <reportfile>
```

**<targetstation>** Name of the server on which the MKF database to be verified resides. MKF\_verify must be run locally.

**<basefile>** Full path name of the base data file of the MKF database partition. This name is in the ddl file for the database.

**table=<tablename>** Name of a single table to be verified. If you do not specify a table, the program verifies all tables in the database.

**phase=<n>** The **n** in the phase option could be set to one of the following:

- 1 Checks the validity of the btrees against the data rows.
- 2 Checks the validity of the data rows against the btrees.
- 12 Performs phase 1, followed by phase 2. This is the default.

**restartfile=<file>** Name of the restart file to create or reuse.

If you do not specify a name, MKF\_verify uses the default restart file, **MKF\_verify.res**, if it exists. Otherwise, it creates the **MKF\_verify.res** file.

**> <reportfile>** Redirects status information generated by MKF\_verify to a report file (for later viewing) and to standard output.

## Example

The following example verifies the transient database on a UNIX system, redirecting the output of MKF\_verify to the /tmp/rpt file. A partial listing is shown.

```
corona(root)/> MKF_verify /fnsw/dev/1/transient_db0 > /tmp/rpt
Begin MKF_verify Thu Nov 14 13:43:44 1996
  DocServer:/fnsw/dev/1/transient_db0
restart point 1.  e.t.: 1 secs  phase: 1
  table num: 0  index: 0
restart point 2.  e.t.: 1 secs  phase: 1
  table num: 0  index: 1
restart point 3.  e.t.: 1 secs  phase: 1
  table num: 0  index: 2
restart point 4.  e.t.: 1 secs  phase: 1
  table num: 1  index: 0
restart point 5.  e.t.: 1 secs  phase: 1
  table num: 1  index: 1
restart point 6.  e.t.: 1 secs  phase: 1
  table num: 1  index: 2
...
table num: 17  index: 0
restart point 44. e.t.: 4 secs  phase: 2
  block: 0
restart point 45. e.t.: 5 secs  phase: 3

End MKF_verify.  e.t. 5 secs  Thu Nov 14 13:43:49 1996
```

**Note** In the above example, phase 3 is the program's wrap-up phase which is very short and exists so if the program is killed or the system dies during phase 3, it will restart in phase 3 on a restart, as it should.

The command you would enter on a Windows Server system would be: **MKF\_verify D:\FNSWDEV\1\TRANSIENT\_DB0 > D:\TMP\RPT**

## Checklist

Before you use MKF\_verify, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_verify.
- You must run MKF\_verify at the server on which the database resides.
- The database must be started. However, the FileNet software can be up or down when you run MKF\_verify.
- If the database is **in use** when you run MKF\_verify, it could degrade system performance.
- You should run this program during off-peak hours. Each verification phase takes 8 to 9 hours per one million rows.
- MKF\_verify is restartable if interrupted by using a restart file.

## Procedure

- 1 Run MKF\_verify with the correct parameters.
- 2 After MKF\_verify completes, check the contents of your report file and the event log.

## Related Topics

**[“MKF\\_fixup” on page 764](#)**

## MKF\_zeroaij

### Description

MKF\_zeroaij exists for one reason: to zero the after-image journal (aij) files after a disk spindle on which the aij resides has failed. The aij is automatically zeroed by FileNet processes for any other situations that require the aij to be zeroed.

---

**Note** “Recovery log” and “after-image journal” are synonymous.

---

When you start MKF\_zeroaij, the MKF\_zeroaij program zeroes the after-image recovery log. The program requests the base data file name rather than the recovery log file name because the base data file has the names of **all** recovery log files and data files. The base data file name is in the ddl file for the database (for example, **/fnsw/local/sd/1/permanent.ddl** or **/fnsw/local/sd/1/transient.ddl** on UNIX, or **D:\FNWSWSD1\PERMANENT.DDL** or **D:\FNWSWSD1\TRANSIENT.DDL** on Windows Server).

### Use

The MKF\_zeroaij program clears any data that exists in either the transient or permanent recovery logs. If the magnetic media containing the recovery log fails, a new drive must be initialized prior to running normal operations. Use MKF\_zeroaij to initialize the new disk drive.

### Syntax

```
MKF_zeroaij <base_data_file>
```

**<base\_data\_file>** Identifies the full path name of the base data file of the MKF database. Be sure to specify the base data file name and not the after-image journal file name.

## Example

**MKF\_zeroaij /fnsw/dev/1/permanent\_db0** (on UNIX)

**MKF\_zeroaij D:\FNSW\DEV\1\PERMANENT\_DB0** (on Windows)

## Sample Output

```
>MKF_zeroaij /fnsw/dev/1/permanent_db0
zeroing /fnsw/dev/1/permanent_rl0 (block 0 through 40959)
MKF_zeroaij: done
```

```
>MKF_zeroaij D:\FNSW\DEV\1\PERMANENT_DB0
zeroing D:\FNSW\DEV\1\PERMANENT_RL0 (block 0 through 40959)
MKF_zeroaij: done
```

## Checklist

Before you use MKF\_zeroaij, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_zeroaij.
- You must notify your service representative before you attempt to run MKF\_zeroaij.
- You must use the base data file name, **not** the recovery log file name.

- You must run MKF\_zeroaij at the server on which the database resides.
- You must shut down the database before running this command.
- No processes should be linked to MKF when MKF\_zeroaij is started. Normally, you should shut down FileNet software before you start MKF\_zeroaij.
- MKF\_xeroaij will refuse to zero the aij if it contains information needed for crash recovery. If this error occurs, contact your service representative. A restore will normally be required.

## Procedure

- 1 Call your service representative.
- 2 Shutdown the FileNet software.
- 3 Run MKF\_zeroaij.
- 4 Restart the FileNet software.

## Related Topics

**[“MKF\\_zerobij” on page 823](#)**

## MKF\_zerobij

### Description

MKF\_zerobij exists for one reason: to zero the before-image journal (bij) files after a disk spindle on which the bij resides has failed. The before-image journal is automatically zeroed by FileNet processes for any other situations that require the bij to be zeroed.

MKF\_zerobij requests the base data file name rather than the recovery log file name because the base data file has the names of **all** recovery log files and data files. The base data file name is in the ddl for the database (for example, **/fnsw/local/sd/1/permanent.ddl** or **/fnsw/local/sd/1/transient.ddl** on UNIX, or **D:\FNSW\SD\1\PERMANENT.DDL** or **D:\FNSW\SD\1\TRANSIENT.DDL** on Windows Server)

### Use

Use MKF\_zerobij to clear data that exists in either the transient or permanent before-image journal. If the magnetic media containing the before-image journal fails, you must initialize a new drive prior to running normal operations. Use MKF\_zerobij to initialize the new disk drive.

---

**Note** The FileNet software currently does not use MKF before-image journals for any MKF databases.

---

### Syntax

```
MKF_zerobij <base_data_file>
```

**<base\_data\_file>** Identifies the full path name of the base data file of the MKF database.

---

**Note** Be sure to specify the base data file name and **not** the before-image journal file name.

---

## Example

**MKF\_zerobij /fnsw/dev/1/permanent\_db0** (on UNIX)  
**MKF\_zerobij D:\FNSWDEV\1\PERMANENT\_DB0** (on Windows)

## Sample Output

```
>MKF_zerobij /fnsw/dev/1/permanent_db0
MKF_zerobij: done
```

```
>MKF_zerobij D:\FNSW\DEV\1\PERMANENT_DB0
MKF_zerobij: done
```

## Checklist

Before using MKF\_zerobij, be aware of the following:

- You must be a member of the fnadmin or fnop group to run MKF\_zerobij.
- You must call your service representative before you attempt to run this program.
- You must use the base data file name. Do not use the recovery log file name.
- You must run the program at the server on which the database resides.



- You must shut down the database before running this command.
- No processes should be linked to MKF when MKF\_zerobij is started. Normally, the FileNet software should be shut down before running MKF\_zerobij.

## Procedure

- 1 Call your service representative.
- 2 Shutdown the FileNet software.
- 3 Run MKF\_zerobij.
- 4 Restart the FileNet software.

## Related Topics

**[“MKF\\_zeroaij” on page 820](#)**

# move\_disk

## Description

The move\_disk tool logically moves storage media from one Storage Library server to another. The move\_disk tool reads storage media information from the MKF permanent database of the Storage Library server on which the media currently reside. It then writes the information into the destination Storage Library server database and updates surface locator tables. Finally, move\_disk deletes the storage media information from the permanent database of the original Storage Library server.

## Use

Use move\_disk to balance the load on the system or to reassign optical media to servers after running add\_osvr or del\_osvr.

In the case of multiple Storage Library servers, always move the primary and tranlog surfaces together to the same server using move\_disk.

Complete the entire move disk procedure (ejecting the surfaces, running move\_disk, and inserting the surfaces) before changing the mode of any of the libraries.

For MSAR surfaces on a network drive, be careful not to eject a surface from a library in Backup mode and move and insert the surface into a library that is in Normal mode. Doing so could easily alter the surface file.

---

If any MSAR library on the local Storage Library server is in Backup mode, the move\_disk program will not allow the move disk operation to be done on any MSAR surface.

## Syntax

```
move_disk <surfid1> <surfid2> ... <surfidn> <dest_server_num>
```

**<surfid1>**  
**<surfid2> ... <surfidn>**

One or more surface IDs.

**<dest\_server\_num>**

Storage Library server ID of the server to receive the media.

## Checklist

Before using move\_disk, be aware of the following:

- Before running move\_disk, physically eject from the storage library all media you want to move.
- The move\_disk tool is typically used as part of a procedure that includes the use of add\_osvr and del\_osvr.
- You can also use Database Maintenance tasks to balance the system load.

## Procedure

The move\_disk tool is typically used as part of a procedure that includes use of the add\_osvr and del\_osvr tools. For an example of a typical scenario in which these tools are used together, see **[“Typical Scenario” on page 290](#)**, under the description of the del\_osvr tool.

## Related Topics

[“add\\_osvr” on page 126](#)

[“del\\_osvr” on page 288](#)

See the “Database Maintenance” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

---

## msar\_io\_test

### Description

The msar\_io\_test program is used to determine relative speed of I/O for a specific directory. You can use this program on the MSAR creation directory.

### Use

This program creates a temporary file in the specified directory. The program will simulate writes of documents and short descriptors. It will also simulate IS document reads from this file. Test results will be printed to standard output and logged in the system log file. Time in minutes will also be displayed for the write and read phases.

### Syntax

```
msar_io_test [-s <surface size in MB>] [-d <document size in KB>]
[-r <read documents (X times)>] [-o] <directory path>
```

---

**Note** All parameters shown in the table below are optional with the exception of the directory parameter.

---

Parameter	Parameter Definition
-s	Surface size in MB (10-32000). The default is 1024 MB (1 GB).
-d	Document size in KB (1-1000). The default is 80 KB.

Parameter	Parameter Definition
-r	Read documents (0-100). The number of reads is specified in multiples of writes. For example, if '2' is used then 2 times the number of write documents will be read. If there were 10,000 document writes, there would be 20,000 reads. Reads will be done in random locations. The default is 1.
-o	Open and Close temporary file on every read. The default is that the temporary file is opened once.
directory	MSAR creation directory path (required parameter)

---

**Note** All reads and writes are done in multiples of 1K sector sizes.

---

For example, if you accept the defaults the command would look like:

```
msar_io_test d:\msar
```

Or if you choose some optional parameters, the command might look like:

```
msar_io_test -s 2048 -r 2 d:\msar
```

You can see the results of running the msar\_io\_test program by viewing the system log.

## Related Topics

[\*\*“sync\\_write\\_test” on page 1267\*\*](#)

Also see the *MSAR Procedures and Guidelines* document. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## msar\_sync\_test

### Description

The msar\_sync\_test program has been enhanced and the name has been changed to sync\_write\_test. See [“sync\\_write\\_test” on page 1267](#)

# nch\_check

## Description

The nch\_check tool verifies that the NCH\_daemon program is running and that the NCH database is providing service for the server's default domain. If a server supports multiple network interfaces or multiple protocol stacks, nch\_check reports all network addresses at which the NCH\_daemon is running.

The system displays error messages on the standard output device. If nch\_check cannot verify that NCH is available, it returns a non-zero exit status.

FileNet initialization routines normally invoke nch\_check to delay the startup of certain software components until NCH is running. Because initialization routines invoke nch\_check with a wait option, nch\_check does not return information until the domain's NCH service is available.

You can also use the nch\_check tool to diagnose network problems. See "Use" below and [“Troubleshooting” on page 834](#).

## Use

After initializing your system, you can use nch\_check as a diagnostic tool to verify that the NCH\_daemon program is running and that the NCH database is providing service for the server's default domain.

When using nch\_check as a diagnostic tool, be aware that nch\_check does not flush the server's domain-address cache. Before you run nch\_check, run the nch\_flush tool to flush the cache. Flushing the cache data ensures nch\_check verifies that the NCH\_daemon program is running on the correct NCH server.



## Syntax

```
nch_check [-w] [-q] [-d <domain>]
```

- w** Waits for NCH services to respond. This option is only used by FileNet software initialization routines.
- q** Runs in quiet mode  
If specified, nch\_check logs error messages to the system event log. If not specified, error messages display on the standard output device.
- d <domain>** Performs verification on the domain specified by <domain>. If you do not specify the <domain>, the program performs verification on the server's default domain.  
<domain> is of the form <domain>[:<organization>].

## Example

In this example, nch\_check verifies that the NCH\_daemon serving corona:FileNet is running and displays the network address that NCH used to locate the domain's NCH\_daemon process.

```
corona(root)/fns/bin> nch_check  
NCH_daemon serving domain "corona:FileNet" running  
at network address [135.0.75.50,32769]
```

Refer to [“nch\\_tool” on page 841](#) for more information on network addresses.

## Troubleshooting

Analyzing nch\_check messages can make you aware of a variety of communication problems that could be occurring in your system. This section describes some of the typical situations you could encounter along with associated error messages and possible corrective actions.

### Configuration Errors

Since nch\_check performs verification for the server's default domain, use the `-d <domain>` option to specify a domain other than the default when you are diagnosing communication problems in cross-system or multiple-system environments.

The following message could indicate that your server configuration is in error:

```
Unable to get default domain name
```

Check for the existence of `/fnsw/local/sd/nch_domain` and that it contains the correct default domain name. If the `nch_domain` file is missing, run the Configuration Builder to create it. If it exists but the default domain name is incorrect, correct the name with the Configuration Editor and run Configuration Builder to rebuild the `nch_domain` file.

### Message 156,0,56

You could get the following message and explanation text:

```
Unable to locate NCH_daemon serving domain "corona:FileNet", error = <156,0,26>
fn_msg 156,0,56
Unable to locate an NCH server for the specified domain.
```

Output from the `fn_msg` tool indicates that `nch_check` is unable to locate an NCH server for the specified domain. Several causes could be associated with this one error message:

- The domain name is incorrect.

Check for the correct domain name in `/fns/local/sd/nch_domain` on the server. It must match that in the configuration database. A typographical error could have occurred when entering the domain name during system configuration.

- The Root server is booting.

You can also see message 156,0,56 if you run `nch_check` while the Root server is booting. When booting is complete, the situation resolves. However, if the message occurs while the Root server is running, verify that the `NCH_daemon` process is running.

Use Task Manager or the **ps -aux** command (shown below) to display the processes. If the NCH\_daemon is not running, use Task Manager to restart the system.

```
corona(root)> ps -aux
  UID    PID    TTY    TIME  CMD
    0   6624  console/0  0:01  gcpsvr
    0   6887  console/0  0:01  gloginsvr
    0   7405  console/0  3:00  scrsvfnsw
    0   8162  console/0  0:00  gloginsvr
.
.
.
   80 54183  hft/0    0:01  ds_init
   80 54935  hft/0    0:24  NCH_daemon
   80 55205  hft/0    0:01  INXbg
```

- The NCH server address is incorrect.

The server's name resolution capability (that is, its ability to map server names to network address through the /etc/hosts file) is unable to provide the name if the address is incorrect. Correct the NCH server address. The following display shows address-to-name mapping contained in a sample /etc/hosts file.

```
# example configuration
#IP address      name

127.0.0.1        localhost
135.0.75.50      corona
```

## Waiting

When you run `nch_check -w`, you could receive messages from `nch_check` indicating that it is waiting, such as:

Waiting for Root Server to start Clearinghouse (NCH)

You can get this message if the Root server is in the process of booting. When booting is complete, the situation resolves. However, if the Root server is running, verify that the `NCH_daemon` process is running. Use Task Manager or the `ps -aux` command to display the processes. If the `NCH_daemon` is not running, use Task Manager to restart the system.

## NCH Inactive

Certain conditions could arise in other areas of the system that cause `nch_check` to issue error messages because the `NCH_daemon` is running but NCH is not. These situations are often the result of configuration errors, and could result in the following messages and error tuples:

NCH server does not support domain, error = <n1,n2,n3>

where the three-part tuple <n1,n2,n3> can be any of the following:

<156,0,25> Probably an MKF-related error

<156,0,24> Network-related error

<156,2,15> Domain name or address configuration error

Some of the possible causes, and actions to take, are:

- NCH is inoperable. Check for possible MKF hangs.

- The default domain name was changed and the Root server's NCH was rebuilt with the Configuration Builder. However, nch\_check is running on a server whose default domain name was not rebuilt and now points to another NCH instance. Delete /fnsw/local/sd/nch\_domain and run Configuration Builder on the server in question to build a correct nch\_domain file for the server's default domain.
- The NCH IP address in /etc/hosts is incorrect and is referencing another NCH instance. Correct the address.

## Checklist

Before you use nch\_check, be aware of the following:

- When using nch\_check as a diagnostic tool, run nch\_flush before you run nch\_check.
- nch\_check performs verification on the server's default domain unless you change the domain with the `-d <domain>` option.

## Procedure

No specific procedure is required.

## Related Topics

[\*\*“nch\\_flush” on page 839\*\*](#)

[\*\*“nch\\_tool” on page 841\*\*](#)

# nch\_flush

## Description

The `nch_flush` tool dynamically resizes cache used by the network clearinghouse (NCH). While resizing cache, `nch_flush` removes in-memory copies of the NCH objects and property values and clears all current NCH server-address cache contents.

---

**Tip** The `nch_tool resizecache` command performs an identical function.

---

## Use

A typical use of `nch_flush` is to force a client NCH to obtain data from the NCH server if the server data has changed and the client has a stale copy of that data.

You can safely run `nch_flush` at any time.

## Syntax

```
nch_flush
```

## Example

In the example below, `nch_flush` clears the cache. No output displays at the terminal if the tool completes successfully.

```
corona(root)/fns/bin> nch_flush  
corona(root)/fns/bin>
```

## Checklist

Before using nch\_flush, be aware of the following:

- No output displays at the terminal if the nch\_flush completes successfully.
- You can safely run nch\_flush at any time.

## Procedure

No specific procedure is required.

## Related Topics

[“nch\\_check” on page 832](#)

[“nch\\_tool” on page 841](#)



## nch\_tool

### Description

The nch\_tool program provides information on the system resources defined in the Network Clearinghouse (NCH) database. The NCH database (NCH\_db0) contains information about the hardware and software resources on your FileNet system. Resources include servers, databases, tape drives, printers, and WorkFlo queues.

#### Entries in the NCH Database

The following is a sample of some entries in an NCH database:

Bes:corona:FileNet

Cache1a:corona:FileNet

DocServer:corona:FileNet

IndexServer:corona:FileNet

Each entry has three parts. Each part tells you something about the system resource. Use the last entry, **IndexServer:corona:FileNet**, to examine the three parts as they are described in the next section.

## NCH Resource Name Description

The first part of this resource description, which is referred to as the **object**, gives you the name of the system resource. In this case, it is IndexServer. This entry relates to the software and hardware for index database management.

The second part of the entry, **domain**, tells you the name of the system where the resource resides. In this case, that system is called "corona." The corona system includes its own server hardware, FileNet applications, and cache; corona could also have attached scanners, printers, and workstations. Many system resources (objects) make up the corona system. IndexServer is just one of them.

The third part of the name, **organization**, is the company or group to which the system belongs. In most cases, the organization is **FileNet** even though the resources belong to a customer. Many resources (objects) make up a system (domain), and, similarly, many systems can make up an organization. In the example above, IndexServer is part of the corona system. And the corona system is part of the FileNet organization.

Each system resource of the NCH database contains three parts, in the format "object:domain:organization." This three-part terminology is used throughout this discussion of nch\_tool.

## Use

To use nch\_tool, you must be logged on as a member of the fnop or fnadmin group.

You normally use nch\_tool when you have detected or suspect an error in the NCH database. In most cases, errors involve an incorrectly-named system resource.

---

### CAUTION

You should use nch\_tool only to diagnose problems. To change the NCH objects and properties, you must use the FileNet System Configuration tools.

---

The FileNet System Configuration Builder automatically rebuilds the NCH database upon a restart of the FileNet software. Resource name changes are automatically updated in this way. However, if a resource name changes but is not updated in the NCH database, the resulting condition could exist undetected until a particular procedure, such as one of the following, is performed and fails with error messages:

- Accessing a WorkFlo queue that is incorrectly named
- Backing up a dataset that has a name change
- Trying to print to a printer that is not properly named

Error message tuples that involve the NCH database start with 156, as shown in the example below:

```
<156,2,16> The name's object does not exist.
```

If you see such an error, you could have an incorrect resource name. Use nch\_tool to locate the problem.

This discussion of nch\_tool includes only options you would use for viewing information, such as listings of NCH objects and properties. If you need, for example, to alter individual entries in the NCH database or resize the NCH cache, call your service representative.

## Syntax

```
nch_tool [-l] [-u] [<file>]
```

- l** Establishes local mode of operation. Makes available some nch\_tool commands that are otherwise not available. Use these extra commands to make changes to the NCH database. In addition, some commands (for example, listproperties) provide more output when you use the -l flag.

---

**Note** You can use the -l flag only on the Root server (where the NCH database resides). Contact your service representative before you use this flag.

---

- u** Establishes update operation mode. Only the FileNet configuration software uses this mode.

Normally, nch\_tool returns an error if it cannot perform a command operation. Update mode instructs nch\_tool to assume that the intent of nch\_tool commands is to update an existing database and to work around the otherwise normal command operation errors. For example, the CreateObject command normally returns a "no change" error if the object to be created already exists in the database. In update mode, the error is not issued.

- <file>** Identifies the name of a input file that contains a list of nch\_tool commands. The FileNet system configuration software uses this file.

---

When nch\_tool is started, the **nch\_tool>** prompt displays. Enter nch\_tool command at this prompt. The syntax of the commands is described below. Individual commands are described in [“\*\*Command Overview\*\*” on page 848](#).

## Syntax Conventions

The nch\_tool commands include references to an NCH resource name that you specify as a three-part, two-part, or one-part name. In addition, the command could include a property specification. This section describes the syntax conventions you must observe when entering an NCH resource name and properties.

An item surrounded by angle brackets (< >) is required input. An item inside square brackets ( [ ] ) is optional.

Commands deal with system resources that contain three parts separated by colons, <object>:<domain>:<organization>, or an abbreviation. You can also use a **pattern**.

Patterns differ from names in that they can use the wildcard character (\*) in the left-most part of the name component. However, the asterisks can only appear in the first (left-most) part of the name. The following example is **not** a legitimate object name because asterisks occur in the first and second parts of the name:

```
Def*.*:FileNet
```

Specify the components of the NCH three-part name as described below.

**<object>** Identifies a Network Clearinghouse three-part resource name, or a suitable abbreviation using a pattern

---

You can use a wildcard character (\*) in the first part of the object name. For example, the following are both acceptable object names:

```
DefaultIMS:rojo:FileNet
Def*:rojo:FileNet
```

You can also specify multiple wildcard characters, as shown below:

```
*ef*:rojo:FileNet
```

If the Domain and Organization variables have a default of "rojo" and "FileNet," respectively, you could also specify:

```
DefaultIMS
```

**<domain>** Identifies a Clearinghouse two-part domain name, or a suitable abbreviation. You can also use a pattern. The following is a two-part name example:

```
rojo:FileNet
```

If the organization variable has a default of "FileNet," you can abbreviate the domain name to **rojo**.

You can use the wildcard character (\*) in the first (left-most) part of a domain name. For example, domain name formats below are acceptable:

```
rojo:FileNet
ro*:FileNet
```

---

You can also specify multiple wildcard characters, such as:

\*oj\*:FileNet

However, the asterisks can only appear in the first (left-most) part of the name. So the following example is **not** a legitimate domain name because asterisks occur in the first and second part of the name:

ro\*:File\*

### <organization>

A Clearinghouse one-part organization name. You can use the wildcard character (\*) in the organization name. All of these organization name formats are acceptable:

FileNet

File\*

\*NET

### <property>

A Clearinghouse property name. Each NCH object can have one or more associated properties. Properties contain data values that describe the object. To view a list of recognized property names, enter **help properties** at the nch\_tool prompt.

## Command Overview

Many nch\_tool commands are available to you. The following list identifies all available commands, in alphabetical order, with a brief description of each. For detailed descriptions and use, see [“Commands” on page 850](#).

Command	Description
adddomain	Adds a domain record
additemproperty	Adds an item property
address	Finds the address of a specified server
cachestats	Displays usage and performance statistics for the NCH cache
changeobject	Changes an object name
changeitem	Changes an item property value
createobject	Creates an object
defaultdomain	Displays the two-part NCH name of the default domain
deleteobject	Deletes an object
deleteproperty	Deletes a specific property
domainserved	Lists domains served by the server from which you enter the command
echo	Echoes input from the command line
exit	Terminates nch_tool (You can also use <b>quit</b> .)
findserver	Displays the network address of the server for a specified domain
help or ?	Displays help for a command
initialize	Creates an instance record
listdomains	Lists domains known by the local or default NCH server



---

Command	Description
listobjects	Lists all objects matching a specific object name
listorgs	Lists organizations known by the local or default NCH server
listproperties	Lists all properties and property values associated with the specified objects and properties
lookupdomain	Displays information on a specified domain
lookupobject	Displays the three-part NCH name of an object
netaddresses	Lists network addresses associated with the local or default NCH server
primarydomain	Changes the name of an existing primary domain
quit	Terminates nch_tool; you can also use <b>exit</b>
read	Redirects input from a specified file
removedomain	Removes a domain
removeremotes	Removes all remote domains
resizecache	Changes cache size
retrieveitem	Displays the property value of a specified object or property
set	Sets or displays a variable value
updatedomain	Updates a domain record in the NCH database
updatenetwork	Updates network numbers in the NCH database

## Commands

Each nch\_tool command is described in detail below. When you list commands using ? at the nch\_tool prompt, the system lists commands by functional categories. However, this section lists commands in alphabetical order.

This section uses uppercase characters to show the minimum number of characters you must enter to run the command. You can always enter more or all characters of a command to run it but you cannot enter fewer characters than those shown in uppercase. For example, to run the changeobject command (shown as CHANGEObject), you must enter at least the following characters:

**changeo**

At the command line, you do not have to enter any character in uppercase letters. Uppercase simply shows the required syntax.

**ADDDomain <name> <service> [<network address>]**

Adds a domain record to the NCH database.

<name> is a domain name.

<service> is specified as one of the following selections: primary, secondary, or tertiary. Specify **primary** for all local domains. Specify **tertiary** for domains supported by other servers but known by this server, that is, other internetworked systems.

<net address> is optional and should only be specified if the service is tertiary. If specified, <net address> is the full network address of the NCH server for that domain.

**ADDItemproperty <name> <property> <value>**

Creates a new object property for the object specified by <name>

<value> is a property value.

If the property already exists, nch\_tool normally returns a "no change" error. However, in update mode, the program does not return this error or report the property value as updated.

## ADDRess <name>

Locates a network address for a symbolic network server name that can be of several forms, including:

- Three part NCH name
- TCP address of the form `###.###.###[,#]` where `#` is decimal
- TCP/IP Internet host name

This function also indicates whether the address is local to the server upon which it is making the call.

## CAchestats

Displays use and performance statistics about the NCH cache, as in the following example:

```
nch_tool> cachestats

    1 cache resizes
    4 server address cache slots
  100 object hash table slots
    50 object cache entries
  200 extent entries, 114 unused
      Timeout = 7200 seconds
    2 find server calls, 0 hits
    1 add server calls, 0 invalidate server calls
  247 find prop calls, 180 hits, 32 timeouts
    78 add prop calls
    0 delete prop calls, 0 delete object calls
    39 free prop calls, 0 free object calls
```

The example above shows that one (1) attempt to dynamically resize cache was made, and displays statistics on table and extent settings for the NCH cache.

"Calls" refer to NCH administrative functions. The example above shows that two (2) find server calls were made, and neither call was satisfied by data in the cache (0 hits). In addition, cachestats reports that: one (1) add server call was made; 0 invalidate server calls (that is, remove a server from the server address cache) were made; 247 find property calls were made, with 180 successfully satisfied by data in cache, and 32 property values were found to have aged beyond the time-out period. Other types of calls are reported as shown.

#### **CHANGEObject <name> <newname>**

Changes an existing object name specified by <name> to a new object name specified by <newname>. The <newname> must be in a domain supported by the same server that supports the domain specified in <name>.

#### **CHANGEItem <name> <property> <value>**

Changes the value of an existing property <property> for the item specified by <name> to the new property value specified in <value>

#### **CRreateobject <name>**

Creates an object specified by <name> in the NCH database

If the object already exists, nch\_tool normally returns a "no change" error. However, in update mode, this error is not returned if the object exists and the object is reported as created.

## DEFaultdomain

Displays the name of the default domain. The domain name has two parts, the domain itself and the organization.

```
nch_tool> defaultdomain
The default domain is corona:FileNet
```

## DELETEObject <pattern> [property]

Deletes all objects that match the object name specified in <pattern> and property criteria. See [“LISTObjects <object> \[<property>\]” on page 858](#) for more information about selection criteria.

In nonlocal mode, deleteobject deletes only a single object, which you must specify exactly. The property argument is not relevant in this mode. In nonlocal mode, nch\_tool normally returns a "no change" error if the object does not exist. However, in update mode, this error is not returned if the object does not exist and the object is reported as deleted.

## DELETEProperty <name> <property>

Deletes the specified property for the object specified by <name> from the NCH database.

If the object does not exist, nch\_tool normally returns a "no change" error. However, in update mode, this error is not returned if the object does not exist and the property is reported as deleted.

## D**O**mainsserved [<service>]

Lists the domains served by this server. The domain name has two parts: the domain itself and the organization. If you do not specify the optional <service> parameter, the primary domain is included, but tertiary domains are not. See below for instructions on how to specify the service. An example is:

```
nch_tool> domainsserved
corona:FileNet
```

The <service> parameter describes a domain service level. Any particular Clearinghouse server can provide one of two levels of service for a particular domain—primary or tertiary.

Primary domain service level indicates that the server is acting as the master for the particular domain. Primary servers manage object and property records for the domain.

Tertiary domain service level indicates that the server knows about the existence of a particular domain but does not actively manage any object or property data for that domain. Remote systems are currently configured by creating tertiary domains with the appropriate names through the System Configuration Editor.

If you specify the `-l` flag when starting `nch_tool`, additional output displays. Specifying a <service> lists information about each domain of that service level or lower when using the `-l` flag. For example, specifying "tertiary" displays all tertiary domains. The output is more complete in this mode and is similar to that of the `lookupdomain` command. (See the description of the `lookupdomains` command for an explanation of the output fields.)

The following example displays detailed information for a tertiary domain:

```
nch_tool> domainserved tertiary
norco:FileNet
    Service = Tertiary, Updates = 0, Agent = [0,0]
    Primary Domain Server Net Address =
    [00000000,000000000000,0000]
```

### **ECho <text string>**

Displays the text following the echo command to the output device

The echo command is useful in command files to write text to an output file.

### **EXit | Quit**

Terminates nch\_tool

### **Findserver <domain>**

Displays the network address of the server for the specified domain, as in the example below. The server address is the IP address. (See the description of IP address under the listproperties and netaddresses commands.)

```
nch_tool> findserver corona
Server address = [135.0.75.50, 32769]
```

You typically might use the findserver command to verify the existence of and the ability to communicate with a specific server in a domain. You can also use the nch\_check command tool to find this information.



## Help | ?

Displays a description of each command. For example, if you enter **help properties** or **? properties**, the names of all recognized properties display.

## Initialize <hint>

Creates an instance record in the NCH database. You would use this command only to initialize the NCH database. It is the first command executed after MKF\_ddl initialization of the NCH database.

<hint> is an object name specified as "hint:defaults:FileNet".

## LISTDomains <domain>

Lists domains known by the local or default NCH server that match the specified two-part domain name. The following command uses the wildcard character to request a listing of all known domains:

```
nch_tool> listdomains *  
corona:FileNet  
norco:FileNet
```

### **LISTOBjects <object> [<property>]**

Lists all objects matching the specified object name. Listobjects is useful for viewing the current resource entries in the NCH database. If you specify a <property> parameter, it restricts the list to objects matching the pattern and having the specified property. (See the listproperties command for a list of valid property names.) An example of listobjects output follows:

```
nch_tool> listobjects *  
Bes:corona:FileNet  
Cachela:corona:FileNet  
DefaultIMS:corona:FileNet  
DocServer:corona:FileNet  
FormsServer:corona:FileNet  
IndexServer:corona:FileNet  
Index_DB:corona:FileNet  
Index_RLa:corona:FileNet  
Index_RLb:corona:FileNet  
OsarServer1:corona:FileNet  
Perm_DB1:corona:FileNet  
Perm_RL1a:corona:FileNet
```

### **LISTORgs <organization>**

Lists organizations known by the local or default NCH server that match the specified one-part organization name (usually "FileNet").

```
nch_tool> listorgs *  
FileNet
```

### **LISTProperties <object> [<property>]**

Lists all properties and property values associated with the specified objects and properties. Each NCH object has one or more associated

---

properties. Properties contain the data values that describe the object; therefore, properties are characteristics of an NCH object. Use the listobjects command to display a list of object names.

**Tip** To get a more detailed display of information from the listproperties command, specify the local flag (-l) when starting nch\_tool. If you do not start nch\_tool with the -l flag, listproperties shows properties for only one object.

---

In the following example, nch\_tool was not started with the local option. The listproperties displays only the properties for the **first** object matching the object pattern specified.

```
nch_tool> listproperties D*

Properties for DefaultIMS:Pubs:FileNet
(ims, "System IMS Defaults")
(IMSDesc, 256 IndexServer:Pubs:FileNet
DocServer:Pubs:FileNet 10126
FormsServer:Pubs:FileNet)
```

In the following example, nch\_tool was started with the local option specified and the listproperties command is issued with the wildcard character to display properties for all objects:

```
nch_tool> listproperties *
Properties for Bes:corona:FileNet
(addressList, [135.0.75.1,32769]
(batchEntryService, "Batch Entry Service")
(batchDesc, 1 DefaultIMS:corona:FileNet bes_cache1:corona:FileNet)

Properties for Cachela:corona:FileNet
(addressList, [135.0.75.1,32769]
(dataSet, "Cache partition")

Properties for DefaultIMS:corona:FileNet
(ims, "System IMS Defaults")
(IMSDesc, 256 IndexServer:corona:FileNet DocServer:corona:FileNet 10000
  FormServer:corona:FileNet)
```

The example above shows properties for three NCH objects: Bes, Cache1a, and DefaultIMS. It associates several properties with each object. (See [“Property Names List” on page 861](#) for a list of valid property names.)

The Properties line of the output displays the three-part NCH object name.

The first line of addressList output displays the IP (Internet Protocol) address (135.0.75.1) followed by the port number. In the example given for Bes:corona:FileNet, the port is 32769, which is Courier using transmission control protocol (tcp). A daemon (in this case, the Courier daemon) "listens" on this socket for incoming requests for connection to the socket.

The remaining lines of the display describe other properties of each object, such as the name of the service and a description, as defined by the FileNet Configuration Editor. For Bes:corona:FileNet in the example above, the name of the service is followed by the description line, which consists of a property version level (in this case, 1), the Image Services name associated with the service (in this case, the default IS), and the batch entry cache name (bes\_cache1:corona:FileNet).

Other information, based on the property type, could also appear, such as dataset names (as shown for Cache1a:corona:FileNet) and system serial numbers (as shown in the entry for DefaultIMS:corona:FileNet, where the system serial number is 10000).

### Property Names List

The following table lists all valid property names. For information about a particular property, specify the property name in the <property> parameter of the listproperties command. To display this list, enter **help properties** or **help \*** at the nch\_tool> prompt.

ADDRESSLIST	AGEABLECACHEDESC	ATTRIBUTEDESC
BATCHDESC	BATCHENTRYSERVICE	CACHEDESC
CACHESERVICE	DATABASEDESC	DATASET
DEFCACHEDESC	DEFDEVICEDESC	DEFSERVICE1DESC
DEFSERVICEDESC	DOCSERVDESC	DOCUMENTSERVICE

DTARSSERVICE	ENTRYCACHEDESC	FAXDESC
FILESERVICE	GROUP	ICRSERVICE
ICRSERVICEDESC	IDMIS	IMSDDESC
INDEXSERVICE	LICENSEDESC0	LICENSEDESC1
LICENSEDESC2	LICENSEDESC3	LICENSEDESC4
LICENSEDESC5	LICENSEDESC6	LICENSEDESC7
LICENSEDESC8	LICENSEDESC9	NETWORKNAMEDESC
OSARDESC	OSARSERVICE	PDBDESC
PRINTCACHEDESC	PRINTER	PRINTERDESC
PRINTSERVDESC	PRINTSERVICE	REFCOUNTCACHEDESC
REGION	RJESERVICE	SECURITYSERVICE
SNAGATEWAY	SORTSERVICE	SQLSERVICE
STATION	SYSDEFDESC	SYSTEM
SYSTEMDEFAULT	TAPEDRIVE	TAPEDRIVEDESC
TAPESERVICE	USER	VWSERVER
VWSERVERDESC	VWSERVICE	WFLQUEUE
WFLQUEUEDESC	WFLQUEUESERVICE	WORKFLOQUEUESERVICE
WQSSERVICEDESC		

The following examples show useful `listproperties` commands. The first example uses a wildcard search (\*) to display the address list for all objects. The second example displays all properties associated with any object that has an object name beginning with the letter P. Because the `-l` option was not specified when `nch_tool` was started, only the first object satisfying the criteria displays.

```
nch_tool> listproperties * addresslist
Properties for Bes:corona:FileNet
  (addressList, [135.0.75.50,32769]
                    [00000000,08005a7a30b7,0005] )
  (batchEntryService, "Batch Entry Service")
  (batchDesc, 2 DefaultIMS:corona:FileNet   bes_
cache1:corona:FileNet DefaultIMS:corona:FileNet)

nch_tool> listproperties P*
Properties for Perm_DB1:corona:FileNet
  (addressList, [135.0.75.50,32769]
                    [00000000,08005a7a30b7,0005] )
  (dataSet, "Permanent Database partition")
```

## LOOKUPDomain <domain>

Displays information on the specified domain. This command is available only if you specify the `-l` flag when starting `nch_tool`. Below are two examples of `lookupdomain` output, followed by explanation of the output fields.

```
nch_tool> lookupdomain corona
  Service = Primary, Updates = 0, Agent = [0,0]
  Primary Domain Server Net Address =
    [00000000,000000000000,0000]
nch_tool> lookupdomain norco
  Service = Tertiary, Updates = 0, Agent = [0,0]
  Primary Domain Server Net Address =
    [00000000,000000000000,0000]
```

Output from the `lookupdomain` command includes the domain's service level (see the `domainserved` command for an explanation of primary and tertiary service levels), updates, the Clearinghouse agent parameters of 0,0, and the network address of the domain. Updates is an unused field at this time and always contain zero. The agent field represents authenticator values for the credentials and verifier agent parameters, used to verify the participants in a connection. At this time, the agent parameter values are always zero. NCH verifies that both agent parameters contain zero.



## LOOKUPObject <object>

Displays the full, three-part NCH name of the first object with a name that matches the specified object. You can also use wildcards as shown in the following example:

```
nch_tool> lookupobject B*  
B*:corona:FileNet --> Bes:corona:FileNet
```

## Netaddresses

Lists the network addresses associated with the local or default NCH server. NCH uses the network address to locate the server that can satisfy the request for service.

```
nch_tool> netaddresses  
[135.0.75.1,32769]
```

The above example displays the IP (Internet Protocol) address of the default server, followed by the port number. In this case, the IP address is 135.0.75.1 and the port number is 32769.

A dual-homing server in a multiserver environment is one in which there are two network adapters per server. If you have installed and configured a dual-homing server, the netaddresses command displays two IP address lines. The IP addresses are different, as shown in the example below.

```
nch_tool> netaddresses  
[135.0.75.1,32769]<—IP Address  
[135.0.75.2,32769]<—IP Address
```

Refer to your FileNet system software installation procedures for information about dual-homing server installation.

Note that the values for the address are determined from the routing information packets (RIPs) received from the network. It specifies the **order** in which the addresses display, which you can change with operating system network interface configuration commands. Refer to your operating system documentation for more information.

### Primarydomain <domain>

---

**Note** The primarydomain command is intended to be used internally only by FileNet System Configuration tools.

---

Changes the name of an existing primary domain to the specified domain name. (This command is valid only if you started nch\_tool in local mode using the -l option.)

All property values associated with the primary domain are automatically changed to reflect the new domain name.

You can use this command to change the name of a primary domain. If you define only one primary domain, it changes the name to the one specified, as well as its property values, to reflect the new primary domain. The command does not change the NCH database if you define multiple primary domains (which is very unusual).

### REAd <filename>

Directs nch\_tool to read and interpret commands from the input file specified by <filename>

**REMOVEDomain <name>**

Removes the domain specified by <name> from the NCH database

If you define multiple primary domains for the system, you can use the removedomain command to remove unwanted domains.

**REMOVERemotes**

Removes all remote domains. Only by FileNet System Configuration programs use this command.

**RESizecache [<servers> <objs> <extents> <timeout>]**

Resizes the domain server address cache and the object property value cache. NCH uses three caches: server address cache, object property value cache, and extents cache. Resizing flushes the cache contents. Running the command with no arguments resizes the caches without changing cache sizes, which provides a useful way to flush the caches.

<servers> controls the number of <domain, server, address> tuples in the domain server address cache. The default value is 4.

The remaining three parameters are related to the object property value cache.

<objects> controls the size of the objects table, that is, the maximum number of objects that can be cached concurrently.

<extents> controls the number of 64 byte extents available for storing property values and therefore is an upper bound on the number of property values that can be cached concurrently.

<timeout> specifies the maximum cache-residency, in seconds, of a property value. When the timeout period expires, NCH considers the property value stale; therefore, you can remove the entry from cache. If the timeout period is zero, it cannot age the property values out of the cache.

The default values are 50 objects, 200 extents, and a timeout value of 7200 seconds (2 hours). To disable caches, set the appropriate sizes to zero.

### **REtrieveitem <object> <property>**

Displays the property value for the specified object and property

```
nch_tool> retrieveitem DocServer docservdesc
DocServer:corona:FileNet
(docServDesc, 2 [Resident] 10000 page_cache1:Pubs:FileNet)
```

For a list of valid property names, see [“Property Names List” on page 861](#).

### **Set [<variable> [<value>]]**

Sets or displays a variable value. If you do not specify any parameters, set simply lists the values of all variables.

<variable> can be one of the following:

- domain sets the domain name variable
- organization sets the organization name variable

<value> is the two-part domain name or one-part organization name you want to set the variable to.

The program uses the domain or organization values specified with the set command as defaults when you do not specify the domain name or organization name in an nch\_tool command. If you do not set these variables with the set command, the program initializes the values for domain and organization using the default NCH domain and organization. It obtains the defaults from the nch\_domain file for UNIX platforms or the Registry database for Windows Server platforms. You can set these defaults with fn\_setup (UNIX) or the FileNet System Configuration Editor (Windows Server).

The following commands set the default domain to "rojo" and the default organization to "FileNet":

```
set domain rojo
```

```
set organization FileNet
```

```
UPDATEDomain <domain_name> <service> [network_address] [new_domain_name]
```

Updates a domain record

---

**CAUTION**

Updatedomain is intended for use only by FileNet system configuration programs to update the service level and network address associated with a domain.

---

<network\_address> is only necessary for tertiary domains on nonlocal networks, in which case you should specify the network address of the NCH server for that domain.

If you specify <new\_domain\_name>, the program changes the domain name and if it is primary, the program updates all property values stored in that domain to reflect its new name.

**CAUTION** Changing a domain name requires changing many files, so you should not manually change domain names. If you need to change a domain name, use FileNet system configuration utilities (fn\_setup for UNIX platforms; System Configuration Editor for Windows Server platforms).

---

#### **UPDATENetwork <pattern> <network>**

Updates network number fields in the addressList property values for those objects specified by <pattern>. This command is used **only** by FileNet system configuration programs.

### **Checklist**

Before you use nch\_tool, be aware of the following:

- To use nch\_tool, you must be logged on as a member of the fnop or fnadmin group.
- The NCH database must be up when running nch\_tool.
- You must call your service representative if you need to make changes to the NCH database or NCH cache.
- To display detailed listings, start nch\_tool with the -l flag.

---

**Note** You can issue the -l flag only from the Root server on which the NCH database resides.

---

## Procedure

- 1 Start the program.

Enter **nch\_tool** at the command line. The following prompt displays:

```
nch_tool>
```

- 2 Enter nch\_tool commands at the prompt.

As you examine the output from commands, you could discover a problem within an NCH resource (such as an incorrect queue, database, or printer description). Call your service representative for further instructions. The NCH database could require modification or re-initialization.

---

### **CAUTION**

Changes to the NCH objects and properties must be made with FileNet system configuration utilities. Use nch\_tool only to diagnose problems.

---

## Related Topics

[\*\*“nch\\_check” on page 832\*\*](#)

[\*\*“nch\\_flush” on page 839\*\*](#)

System Configuration Editor online help (for Windows Server platforms) or fn\_setup online help (for UNIX platforms)

See your system software installation procedures for an explanation of network installation.

## NLT\_build\_maps

### Description

The NLT\_build\_maps tool allows you to create a character set translation map which maps a user-defined or non-FileNet system character set to the system default character set. You can define the system character set using the FileNet Configuration Editor. It represents the default character set the system uses to represent all character-based data. The newly created character set map allows you to import foreign data from non-FileNet systems, such as EBCDIC from mainframes, to a FileNet system. After importing the data, you can display and store it on the FileNet system.

To begin the translation process, you must create ASCII-based maps in source format. The source translation map consists of 256 records. The format of each record identifies character mapping from the source character set to the destination character set's corresponding hexadecimal value for that character. The records are representations of each character in the character set in the following arrangement:

```
0xc1 0x41 capital letter A
```

where

- the first column is the hexadecimal offset into a specified source character set
- the second column is the destination character set's corresponding hexadecimal value for the character
- the third column is optionally used for comments. Use this space to describe the character being mapped.



- Any record beginning with a # or "white space" (spaces or tabs) is considered a comment line.

The FileNet software provides six sample map files in the following directory:

/fnsw/lib/nltmapsFor UNIX platforms

\fnsw\lib\nltmapsFor Windows Server platforms

Two of the files provide source code for translation of the EBCDIC character set. In addition, the FileNet software provides the binary normal and inverse maps for translations of EBCDIC and ISO 8859-1. The table below describes each file:

File	Description
eb_fn_src	Source for translation of EBCDIC to FileNet International character set
eb_8859-1_src	Source for translation of EBCDIC to the ISO 8859-1 character set
ebcdic_8859-1	Binary map of EBCDIC to ISO 8859-1 translation
8859-1_ebcdic	Binary map of binary map of ISO 8859-1 to EBCDIC translation (This is the inverse map of ebcdic_8859-1.)
ebcdic_filenet	Binary map of EBCDIC to FileNet International character set translation
filenet_ebcdic	Binary map of FileNet International character set to EBCDIC translation (This is the inverse map of ebcdic_filenet.)

**Note** When NLT\_build\_maps builds the binary maps, it places them in the local (current) directory. If the local (current) directory is other than /fnsw/local/nltmaps (or \fnsw\_loc\nltmaps for Windows Server), a message informs you to move the maps to /fnsw/local/nltmaps. This is necessary because the NLT program that performs the translation looks

for and uses the map located in /fnsw/local/nltmaps even if a map of the same name exists in /fnsw/lib/nltmaps.

---

In addition to the binary file of the map, NLT\_build\_maps creates a file consisting of the inverse mappings of the characters. You provide the name of this inverse map in the `-i` option. For example, if you created a map that translates EBCDIC to the FileNet International character set, the inverse map provides translation from the FileNet International character set back to EBCDIC.

We strongly suggest that you choose names for your normal and inverse map files that describe the type of mapping that is taking place. This is the file name displayed by the various applications (such as COLD 2) for selection, and therefore should be self-describing. For example, a file name of `ebcdic_to_8859` tells you that the source character set is EBCDIC and the destination character set is FileNet ISO 8859. Compare this to a file name of `translate_to_iso`, which tells you the destination file is ISO, but does not tell you anything about the source character set.

After the create process, you can run the verification option (`-v`) to ensure that your files have correctly mapped each character in the set. If you have not mapped the characters correctly, NLT\_build\_maps displays as much information about the record in error as possible in either an ERROR or WARNING message, similar to the following:

```
WARNING: One-to-many mapping detected (record=n,index=0xnn). Record ignored.
```

It might also display a sequence similar to the following:

```
ERROR: Many-to-one mapping detected in inverse map  
(index=0xnn).
```

```
ERROR: Can't verify inverse map due to previous errors.
```

```
NLT_build_maps: Verification of inverse map fn_eb failed!
```

where **n** is the record in error and **nn** is the index value into the character set.

One-to-many mapping errors occur when you assign a character value that is already mapped to more than one character in the destination character set. This situation causes only warning messages to display, alerting you of the condition.

Many-to-one mapping errors occur when the source map file maps more than one of the same character value. For example, in mapping the EBCDIC capital J to the FileNet International character set, you could have made a typographical error. When mapping 0xd1 to 0x4a, you typed 0xc1 instead of 0xd1. Since 0xc1 is already mapped to 0x41, in our example, verification returns a many-to-one mapping error. Verification fails on this type of mapping error.

When you have created and verified the translation maps, move the normal and inverse maps to /fnsw/local/nltmaps (or \fnsw\_loc\nltmaps for Windows Server). (NLT\_build\_maps displays a message reminding you to do this.) It is then the responsibility of the application to call NLT to retrieve the list of these user-defined names. The NLT program passes an array of the map names back to the application, and the application displays the list to the users for selection.

A typical example of this is the COLD 2 application. A COLD 2 user can import non-FileNet-supported EBCDIC data into the FileNet system for storage and/or display. COLD 2 calls NLT in the above manner to perform the translation on the data being imported.

## Use

Use NLT\_build\_maps to create a character set by translating a user-defined or non-FileNet-supported character set to the system character set. Use NLT\_build\_maps to create character sets that allow you to import non-FileNet system data (such as EBCDIC) into FileNet applications (such as COLD 2).

The FileNet software provides two sample source map files for translating the EBCDIC character set. One file translates EBCDIC to the FileNet International character set, and the other translates EBCDIC to the ISO 8859-1 character set. In addition, the FileNet software provides the binary normal and inverse maps for these translations. All these files are located in /fnsw/lib/nltmaps (or \fnsw\lib\nltmaps for Windows Server).

## Syntax

In the syntax below, you can separate an option from the map name with a space, but a space is not required.

```
NLT_build_maps {-c | -v} -s<source_map> -n<normal_map>  
                -i<inverse_map>
```

- c** Creates translation maps using translation mapping records from <source\_map>. (-c is mutually exclusive with -v)
- v** Verifies translation maps using translation mapping records from <source\_map>. To check for mapping errors, use -v after you create a translation map. (-v is mutually exclusive with -c)
- s<source\_map>** Identifies the name of the ASCII source translation file containing translation mapping records. This option is required.
- n<normal\_map>** Identifies the name of the normal translation map to be created or verified in the local directory using <source\_map> as the input file. The local directory is the current directory.  
-n or -i (or both) is required.
- i<inverse\_map>** Identifies the name of the inverse translation map to be created or verified in the local directory using <source\_map> as the input file. The local directory is the current directory.  
-i or -n (or both) is required.

## Example

The following partial list of records from the source map file `/fnsw/lib/nltmaps/eb_8859-1_src` maps the characters from the EBCDIC character set to the ISO 8859-1 character set.

```
# ebcdic_iso8859-1_src:
#
# This file contains the character mappings from the EBCDIC character set # to the
# ISO 8859-1 character set. The first column represents the EBCDIC # character's
# representation in hexadecimal. The second column
# represents the corresponding ISO 8859-1 character's representation in
# hexadecimal. The third column contains the visual description of the
# character. The sequence XXX corresponds to bit combinations that do not #
# represent graphic characters.
#
0x00    0x00    XXX
0x01    0x01    XXX
.
.
0x40    0x20    space
0x41    0xa0    no-break space
0x42    0xa1    inverted exclamation mark
0x43    0xa2    cent sign
.
.
0xc1    0x41    capital letter A
0xc2    0x42    capital letter B
0xc3    0x43    capital letter C
```

Taking a character from the above source map as an example, the following record maps the character 'A' from the EBCDIC character set to the ISO 8859-1 character set:

0xc10x41capital letter A

In EBCDIC, 'A' is represented by 0xc1. In ISO 8859-1, 'A' is represented by 0x41.

## Checklist

Before you use NLT\_build\_maps, be aware of the following:

- Before you run NLT\_build\_maps, you must map the characters in a source file.
- You should use a naming convention for your normal and inverse map files that makes it easy to tell what type of translation is being mapped in each.

## Procedure

- 1** Create the source mapping file for the character set you want to translate.
- 2** Run NLT\_build\_maps with the appropriate options to create and verify the binary normal and inverse maps from the source map file.
- 3** Move the binary normal and inverse map files to /fnsw/local/nltmaps for UNIX platforms or \fnsw\_loc\nltmaps for Windows Server platforms.

## Related Topics

Refer to online help for FileNet configuration utilities (System Configuration Editor for Windows Server and `fn_edit` for UNIX platforms) for information on the system default character set.

See the *COLD 2.0 Handbook* for information on the use of `NLT_build_maps` with the COLD 2 "import document" function.



## ntdm\_exp

### Description

Image Services provides a single user logon from a trusted Windows Server domain. This functionality uses FileNet-provided tools to first export the users names and group names from the Windows Server domain and then import them into the Image Services (IS) Security Service.

### Use

It is the use of the ntdm\_exp tool in conjunction with either the RSEC\_imp tool or the SEC\_imp tool that makes the single logon from a trusted Windows Server domain feature possible.

Run ntdm\_exp to export the Windows Server domain security information to an XML formatted file. This requires the user to run the export command twice with different options each time. The first time exports the group names and creates the group.lst control file. The second time the command is run exports groups and users to the XML formatted file using the group.lst control file created from the first running of the command.

### Syntax

First, export group names and create the group.lst control file:

```
ntdm_exp /L<domain> /O<file> [/G<op>]
```

Second, export groups and users to the XML formatted file using the group.lst control file created in the first command:

---

```
ntdm_exp /I<file> /O<file> [/E<file>] [/S]
```

**/? or /H** Help

**/L<domain>** Generate a control file containing all unique group names for the given Windows Server primary domain

**/I<file>** Export control file

**/O<file>** Output file

**/G<op>** Global group operator (a, e, sk, se, sx, dk, de, dx, x)

(a) - **Add** on groups and users.

(e) - **Expire** groups and users.

(sk) - **Sync** groups and users in the sec\_groups table. If a user is only in the sync group, **kill** actual user in sec\_object table. Otherwise, remove user from the sync group only.

(se) - **Sync** groups and users in sec\_groups table. If a user is only in the sync group, **expire** actual user in sec\_object table. Otherwise, remove user from the sync group only.

(sx) - **Sync** groups and users in sec\_groups table. If a user is only in the sync group, **no operation** on actual user in sec\_object table. Otherwise, remove user from the sync group only.

(dk) - **Delete** groups and users in sec\_groups table. If a user is only in the delete group, **kill** the actual user in sec\_object table. Otherwise, remove user from the delete group only.

(de) - **Delete** groups and users in sec\_groups table. If a user is only in the delete group, **expire** actual user in sec\_object table. Otherwise, remove user from the delete group only.

(dx) - **Delete** groups and users in sec\_groups table. If a user is only in the delete group, **no operation** on actual user in sec\_object table alone. Otherwise, remove user from the delete group only.

(x) - **No operation.**

(If /G option isn't specified, the default group operator is x)

**/E<file>** **Exclude** users from being exported.

**/S<file>** Omit groups/users that contain **special** characters (spaces or special characters).

## Procedure

- 1 Log on to the Windows Server client from which you plan to export the user and group information. You need to log on as a user with sufficient rights to complete the export.
- 2 Download the ntdm\_exp tool from the Web to the \fnsw\CLIENT\bin directory. The tool is located on the IBM Information Management support page (<http://www.ibm.com/software/data/support>).
- 3 On the Windows Server client station, go to the directory in which you want your export file(s) created, for example, \fnsw\CLIENT\tmp.
- 4 From the Windows Server client station, export the Windows Server domain security information to an XML formatted file. To do this, you need to run the ntdm\_exp command twice with different options each time:

- First to extract the group names in the Windows Server domain.
- Second to extract the users in specified groups.

The result of the second run is a file that is later imported to the particular Image Services library on which the user wants to configure the accounts. Review the ntdm\_exp options **before** running the export command. The ntdm\_exp tool must be run from a DOS environment on any Windows client machine that is in the correct domain. When ready, run the following export operations:

- a First, use the following command to export group names and create group.lst, the group list control file:

```
ntdm_exp /L<domain> /O<file> [/G<op>]
```

Example:

```
ntdm_exp /Lnt1 /Ogroup.lst /Gse
```

- b Second, use the following command to export groups and users to the XML formatted file using the group.lst file created in the first command:

```
ntdm_exp /I<file> /O<file> [/E<file>] [/S]
```

Example:

```
ntdm_exp /Igroup.lst /Oxml.dat /S
```

**Note** All error messages are logged to a file called ntdm\_exp.log in the current directory. Releases prior to IS 3.6 must use the /S option when running the command to create the XML formatted file.

---

- 5 After you have successfully created the output file, you have to import the Windows Server domain security information into the IS Security Service. This step requires the use of the XML formatted file generated from the export step. To complete this import step you have two choices:
  - Import the XML file by running the security import tool on an Image Services server. The tool used to perform this function is SEC\_imp. To review this import tool, go to [“SEC\\_imp” on page 1129](#)
  - Import the XML file by running the remote security import tool on an Windows Server machine. The tool used to perform this function is RSEC\_imp. To review this import tool, go to [“RSEC\\_imp” on page 1103](#).

## Related Topics

[“RSEC\\_imp” on page 1103](#) and [“SEC\\_imp” on page 1129](#)

# oddump

## Description

The oddump tool is a utility for analyzing storage media problems and, in certain cases, repairing media problems. Features of the tool include a hexadecimal dump option as well as a formatted print mode. You can also use oddump to display storage media data structures. In addition, oddump performs database updates when such updates must be done in conjunction with storage media changes.

The oddump tool detects when document services software is running and determines if a storage drive is disabled. If document services software is running, oddump can perform some of its functions only on a disabled drive.

If your system has more than one Storage Library server, oddump prompts you to enter a lowercase letter indicating the Storage Library server that contains the desired drive.

---

**Note** Even though oddump is an optical disk dump utility, it can be used for MSAR libraries, drives, and surfaces. All of the existing oddump options will be supported for MSAR surfaces.

---

## Use

Use oddump to investigate and repair storage media problems. Repairs could include resetting the high-water mark and creating a new doc\_hdr table.

The oddump tool runs in two modes—read-only and update/modify. Read-only mode is the default. To perform update and modify func-

tions, you must issue the ALLOWUPDATES command, which requires a password from your service representative. All oddump commands are available to update and modify database tables in addition to all read-only functions.

## Syntax

### oddump

When you enter oddump at the command line, the program prompts you to enter a drive number.

```
vicenza(fnsw)/home/fnsw> oddump

ODDUMP -- Optical disk dump.

Storage Library containing drive? (a,b,...,p):a

Logical drive number? (0,1,...,9,a,b,c or CR=none):
No drive selected

Type '?' for help
<oddump:lib a>
```

If you do not select a drive at this time (as depicted in the example above), the <oddump> prompt will only show the library selected. Once you select a drive, the drive number will also be reflected in the prompt, for example <oddump:lib a:drive 1>. At any rate, enter commands at the <oddump:lib a> prompt. See [“Commands” on page 888](#) for a list of valid commands and their use. Some commands have a number of subcommands, which are also described in the “Commands” section.

## Commands

A variety of oddump commands enable you to investigate and repair storage media problems. The following table lists each command in alphabetical order with a brief description of each. Detailed descriptions of each command follow the table.

**Tip** UPPERCASE letters in command descriptions denote the minimum number of characters you must enter on the command line to run the command. For example, to run the cachedochdr command, you can enter any of the following commands: CA, ca, CACHEDOCHDR, cachedochdr.

Command	Description
<b><u>ALLOWUPDATES</u></b>	Establishes correct permissions for running certain commands
<b><u>CA</u>chedochdr</b>	Displays a document header
<b><u>C</u>hecksum</b>	Displays the checksum computed for a page and the checksum stored for a page
<b><u>CL</u>ASSinfo</b>	Displays a list of classes, indexes, and index types for documents
<b><u>C</u>OPYFile</b>	Copies a storage media file to a magnetic disk file or copies a disk file to a storage media file
<b><u>C</u>OPYSectors</b>	Copies raw sectors on storage media
<b><u>CR</u>EAThedrfile</b>	Creates a new document header file on storage media and updates the database
<b><u>DO</u>CHDRdump</b>	Dumps document header files
<b><u>DO</u>CIDRange</b>	Reports minimum and maximum document IDs
<b><u>E</u>VENT</b>	Extracts event logs for the Plasmon (Philips) 6000- or 8000-series optical drive
<b><u>FI</u>NDDoc</b>	Finds all document headers in the image area



Command	Description
<b><u>Findunused</u></b>	Finds an unused sector past an area of written sectors
<b><u>FIRMWARE</u></b>	Downloads firmware for Plasmon (Philips) 6000- or 8000-series optical drives automatically
<b><u>Fmtdump</u></b>	Browses the hierarchical data structure on storage media
<b><u>HARdcopy</u></b>	Sends a copy of all screen output to a file
<b><u>Help</u></b>	Displays commands and their descriptions
<b><u>Highwater</u></b>	Displays the value of the next available sector on the storage medium
<b><u>HPONLINE</u></b>	Enables you to take a specific HP drive offline and back online so the HP service technician can repair and replace an optical drive with minimal impact on the FileNet system
<b><u>INITPseudo</u></b>	Erases initialized pseudo-optical media
<b><u>Loaddisk</u></b>	Loads a medium from any slot of a drive into the selected drive
<b><u>LOCKdoor</u></b>	Locks the door of an Optical Disk Unit (ODU) to prevent media removal; however, the door unlocks and ejects media automatically when the drive resides in a storage library.
<b><u>Map</u></b>	Issues a map command to the drive and reports the written/unwritten status of the area mapped
<b><u>Quit</u></b>	Exits oddump
<b><u>Rawdump</u></b>	Dumps selected sectors in hexadecimal and ASCII format
<b><u>READLoop</u></b>	Reads a range of sectors
<b><u>REady</u></b>	Issues a ready command to the drive and reports current status
<b><u>SETPseudoerr</u></b>	Sets errors on a pseudo-optical medium (Used for testing only)
<b><u>SElectdrive</u></b>	Changes a storage library or drive number
<b><u>SPinup</u></b>	Issues a spinup command to the drive

---

Command	Description
<b><u>UNLockdoor</u></b>	Unlocks the door and ejects the media of a drive that resides in an Optical Disk Unit (ODU)
<b><u>WRITELoop</u></b>	Writes a test pattern at the high-water mark of the medium and updates the high-water mark in both the surface record and the database

In the command descriptions that follow, replace items in angle brackets (< >) with an appropriate value. Items in square brackets ( [ ] ) are optional and UPPERCASE identifies characters you need to type. All commands are case insensitive; therefore, you can enter any keyword in either uppercase or lowercase. You can abbreviate keywords listed in uppercase by specifying the capitalized characters of the keyword.

### **ALLOWUPDATES**

Enables functions that write to storage media and update the database. Requires a password, available from your service representative.

### **CAchedochdr**

Displays a document header (page 0 of a document) from the page cache. You must enter a document ID. Displays the attributes of the document, including indexing information, and the size and location of each document page.

### **Checksum**

Displays the checksum computed for a page and the checksum stored for a page when given a sector number that contains the document header of a document and a page of the document. This is useful to determine if the checksum on a storage media page is invalid.

**Note** If a background job has an ODU file handle open as it sits in the wait loop because an MSAR library is in Backup mode, this command that requires an exclusive drive open access to the surface will fail with the drive busy error.

---

### CLASSinfo

Displays a list of classes, indexes, and index types for documents in the directory. You can use the command output to set up a database for media import. The list can be for either all document header files on the media or for a single file. Since output can be lengthy, consider using the hardcopy command to send output to a file. (See [“HARDcopy” on page 916.](#))

**Note** If a background job has an ODU file handle open as it sits in the wait loop because an MSAR library is in Backup mode, this command that requires an exclusive drive open access to the surface will fail with the drive busy error.

---

### COPYFile

(Requires ALLOWUPDATES)

Copies a storage media file to a magnetic disk file or copies a magnetic disk file to a storage media file. When copying a file to storage media, a file with a file name prefix of "FN\_DESCRIPTOR\_" must be a valid document header file. Files with other prefixes can contain any type of data.

**CAUTION** Copying magnetic disk files to storage media is not an officially supported feature. Use it only with approval from your service representative.

---

---

**Note** If this command attempts to modify the surface file of an MSAR library in Backup mode, it will not be allowed to complete.

---

If you select the copy to storage media option, the following prompts display:

```
Extent size is kilobytes (min=200, CR=100):  
Buffer size in kilobytes (CR=64):  
Mag disk file to read:  
Optical disk file to create and write (CR=same name):  
Begin copy (y/n):
```

Extent size should be large enough to prevent excessive allocation, which takes time, and small enough to avoid wasting space at the end of the last partially filled extent sector.

The recommended buffer size is 64KB. The magnetic disk file names must follow rules set by the native operating system. The storage media file name must follow storage media format specification rules.

If you copy to magnetic disk, the following prompts display:

```
Buffer size in kilobytes (CR=64):  
Optical disk file to read:  
Mag disk file to create and write (CR=same name):  
Begin copy (y/n):
```

---

**Note** If a background job has an ODU file handle open as it sits in the wait loop because an MSAR library is in Backup mode, this command that requires an exclusive drive open access to the surface will fail with the drive busy error.

---

## COPYsectors

(Requires ALLOWUPDATEs)

Copies raw sectors on storage media using one of three options. When you enter copysectors, a prompt for the mode displays. The following table describes these modes.

Mode Selection	Mode of Operation	Description
a	Copy storage media sectors to storage media sectors	Copies data from one location to another on the same storage medium
b	Copy storage media data to magnetic disk file	Copies data from storage media to a magnetic disk file
c	Copy magnetic disk file to storage media sectors	Copies data from a magnetic disk file to storage media

You can also use these options to copy data from one storage medium to a magnetic disk file, then to a second storage medium.

---

**Note** If this command with the **a** and **c** options attempts to modify the surface file of an MSAR library in Backup mode, it will not be allowed to complete.

---



---

**Note** Copying magnetic disk files to storage media is not an officially supported feature. Use it only with approval from your service representative.

---

## CREATEhdrfile

(Requires ALLOWUPDATES)

Creates a new document header file on the storage medium and updates the database to include this file during document writes. When executed, the following prompt displays:

```
Create new document headers file. Confirm? (y/n)
```

---

**Note** If this command attempts to modify the surface file of an MSAR library in Backup mode, it will not be allowed to complete.

---

## DOCHDRdump

Dumps document headers files

---

**Note** If a background job has an ODU file handle open as it sits in the wait loop because an MSAR library is in Backup mode, this command that requires an exclusive drive open access to the surface will fail with the drive busy error.

---

## DOCIDRange

Reports minimum and maximum document IDs on the selected storage media.

## EVENT

Extracts Plasmon (Philips) 6000- or 8000-series optical drive event log data for analysis. The log file generated provides essential troubleshooting data for service representatives.

---

**Note** Only your service representative should use the event command. The oddump help command does not list the event command.

---

When you enter the EVENT command, oddump reads the parse file associated with the type of drive selected to determine how to format the event log file. After successfully generating the event log file, oddump displays a message showing the log file name and directory in which the program stored the file.

If desired, you can redefine the format displayed in the event log file by editing the appropriate parse file using a text editor. For details, see [\*\*“Event Log Parse File” on page 906.\*\*](#)

## Generated Event Log File

The program stores log data in an ASCII file in the following directory:

/fnsw/local/logs/phil\_elog           for UNIX platforms

<drive>:\fnsw\_loc\logs\phil\_elog for Windows Server platforms

The output file name is in the following format:

o<library\_id>d<drive\_number><yyyymmdd>

where:

<library\_id> is the Plasmon (Philips) 6000- or 8000-series storage library ID

<drive\_number> is the storage media drive number

<yyyymmdd> is the year, month, and day on which the log file is generated.

For example, output file /fnsw/local/logs/phil\_elog/oad019990301 was created on March 1, 1999 and contains event log data for storage library A, drive 0.

---

**Note** A Storage Library server is sometimes still referred to as an "OSAR." The first character of the output file name is an alphabetic o for OSAR, not a numeric zero.

---



When you run the event command, oddump generates the logs listed below:

<b>Log type</b>	<b>Page Code Identifier</b>
Drive Statistics Log	1
Seek Statistics Log	2
Shuttle, Motor and Baseplate Statistics Log	3
DPC Statistics Log	4
Write Statistics Log Side A	5
Write Statistics Log Side B	6
Read Statistics Log Side A	7
Read Statistics Log Side B	8
SCSI Activity Log	9
System Event Log	A
Failure Analysis Log	B
Media Management Log	C
6000 Media Specific Disk Information, SDI Side A Log	D
6000 Media Specific Disk Information, SDI Side B Log	E
6000 Media Product Specific Information, PSI Log	F
8000 Media Specific Disk Information, SDI Side A Log	10
8000 Media Specific Disk Information, SDI Side B Log	11
8000 Media Write Power Calibration log	12

The System Event Log (page code A) is disabled by default. Output for disabled page codes is all zeros. To extract System Event Log information, you must enable the feature through the configuration options of the Philips Drive Operator Console (DOC).

Page code B might exist, depending on the firmware level installed on the storage library. If page code B exists, the log type is Log B. If Log B does not exist, oddump makes an entry in the output file that includes the Write Once Optical Drive Interface (WOODI) firmware revision value (bytes 5 through 10 of page code 1).

If oddump encounters errors in any page code, oddump logs an error message in the output file, skips the page code in error, and continues to the next. Event log extraction does not clear all event logs.

The output file lists each log, separated by the title of the log file and page code number.

## Event Log File Example

The following example shows the screen displayed when entering the oddump commands required to generate an event log file for an 8000 series Plasmon (Philips) drive, located in storage library c, logical drive number 1, on October 13, 1999.

```
borabora(fnsw)/fnsw/local/logs/phil_elog> oddump

ODDUMP -- Optical disk dump.

Storage Library containing drive? (a, b,...,p) <c>
Logical drive number? (0, 1, ..., 9, a, b, ..., or CR=none) <1>
Drive successfully opened.  Checking label . . .
Disk is labeled; Surface id is 3000.

Type '?' for help

<oddump:lib c>event
Extract LD6100/8100 access event log into file /fnsw/local/logs/phil_
elog/ocd119991013 successfully
```

The command example shown above generated the event log file presented in the following example.

---

The following example shows the primary log file headers created for this event log, but omits the complete data where indicated with an ellipsis (...).

---

LD8100 DRIVE

\*\*\*\*\* Drive Statistics Log (PAGE CODE = 1, Size = 200) \*\*\*\*\*

Drive Serial Number (byte 000 - 004): 80005  
WOODI Firmware Rev (byte 005 - 00a): 091004  
RWS/A Firmware Rev (byte 00b - 010): 691004  
RWS/B Firmware Rev (byte 011 - 016): 691004  
DPC Firmware Rev (byte 017 - 01c): 790928  
WOODI Serial Number (byte 01d - 01f): 000000  
RWS/A Serial Number (byte 020 - 022): 000013  
RWS/B Serial Number (byte 023 - 025): 000011

...

Raw Data: Drive Statistics Log (PAGE CODE = 1, Size = 200)

0000: 38303030353039313030343639313030  
0010: 34363931303034373930393238000000

...

\*\*\*\*\* Seek Statistics Log (PAGE CODE = 2, Size = 40) \*\*\*\*\*

Side A Seek Errors (byte 000 - 003): 00000000  
Side A Seek Operations (byte 004 - 007): 0000c083  
Side A Seek Retries (byte 008 - 00b): 0000011d  
Side B Seek Errors (byte 014 - 017): 00000000  
Side B Seek Operations (byte 018 - 01b): 0000d895

...

---

\*\*\*\*\* Shuttle, Motor and Baseplate Statistics Log (PAGE CODE = 3, Size = 134) \*\*\*\*\*

Information Reads On Spin Up (byte 000 - 003): 0000004c  
Focus Not Achieved On Spin Up (byte 004 - 007): 00000000  
Total Spin Ups (byte 008 - 00b): 00000026  
Total Spin Up Failures (byte 00c - 00f): 00000000  
8600 Media Insertions (byte 010 - 013): 00000000  
DI Retries (byte 014 - 017): 00000000  
Door Openings (byte 024 - 027): 00000000  
Shuttle Initializations (byte 028 - 02b): 00000000  
Shuttle Movements (byte 02c - 02f): 00000000  
Shuttle Movement Failures (byte 030 - 033): 00000000  
Media Insertion Failures (byte 034 - 037): 00000000  
Slot 1 Media Insertions (byte 038 - 03b): 00003fee  
Slot 2 Media Insertions (byte 03c - 03f): 00000000

...

Raw Data: Shuttle, Motor and Baseplate Statistics Log (PAGE CODE = 3, Size = 134)

0000: 0000004c00000000000000002600000000  
0010: 0000000000000000000000000000000000

...

\*\*\*\*\* DPC Statistics Log (PAGE CODE = 4, Size = 230) \*\*\*\*\*

DPC Commands Not Completed (byte 000 - 003): 00000000  
Shuttle Door Faults (byte 004 - 007): 00000000  
Shuttle Unload Faults (byte 008 - 00b): 00000000  
Shuttle Load Faults (byte 00c - 00f): 00000000  
Shuttle Position Faults (byte 010 - 013): 00000000  
Shuttle Initialization Faults (byte 014 - 017): 00000000  
Baseplate Open Faults (byte 018 - 01b): 00000000  
Baseplate Close Faults (byte 01c - 01f): 0000000b

...

---

Raw Data: DPC Statistics Log (PAGE CODE = 4, Size = 230)

0000: 00000000000000000000000000000000

0010: 0000000000000000000000000000000b

...

\*\*\*\*\* Write Statistics Side A (PAGE CODE = 5, Size = 300) \*\*\*\*\*

Sectors written by channel A : 00000000

Total channel A relocs : 00000000

Channel A relloc rate : 0.000000 %

Drive Not Configured Errors (byte 000 - 003): 00000000

Command Not Completed Errors (byte 004 - 007): 0000024f

Arbitrary Faults (byte 008 - 00b): 0000024f

Focus Faults (byte 00c - 00f): 00000000

...

Raw Data: Write Statistics Side A (PAGE CODE = 5, Size = 300)

0000: 000000000000024f0000024f00000000

0010: 00000000000000000000000000000000

0020: 00000000000000000000000000000000

...

\*\*\*\*\* Write Statistics Side B (PAGE CODE = 6, Size = 300) \*\*\*\*\*

Sectors written by channel B : 00000000

Total channel B relocs : 00000000

Channel B relloc rate : 0.000000 %

...

---

Raw Data: Write Statistics Side B (PAGE CODE = 6, Size = 300)

0000: 000000000000000010000000100000000

0010: 000000000000000000000000000000000

0020: 000000000000000000000000000000000

...

\*\*\*\*\* Read Statistics Side A (PAGE CODE = 7, Size = 600) \*\*\*\*\*

Drive Not Configured Errors (byte 000 - 003): 00000000

Command Not Completed Errors (byte 004 - 007): 00000865

Arbitrary Faults (byte 008 - 00b): 00000865

Focus Faults (byte 00c - 00f): 00000000

Tracking Out of Limit Faults (byte 010 - 013): 00000000

...

Raw Data: Read Statistics Side A (PAGE CODE = 7, Size = 600)

0000: 00000000000008650000086500000000

0010: 00000000000000000000000000000000

...

\*\*\*\*\* Read Statistics Side B (PAGE CODE = 8, Size = 600) \*\*\*\*\*

Drive Not Configured Errors (byte 000 - 003): 00000000

Command Not Completed Errors (byte 004 - 007): 00000371

Arbitrary Faults (byte 008 - 00b): 00000371

Focus Faults (byte 00c - 00f): 00000000

Tracking Out of Limit Faults (byte 010 - 013): 00000000

Transfer Faults (byte 014 - 017): 00000000

C Clock Faults (byte 018 - 01b): 00000000

Motor Speed Faults (byte 01c - 01f): 00000000

Read Timing Faults (byte 020 - 023): 00000000

...

---

Raw Data: Read Statistics Side B (PAGE CODE = 8, Size = 600)

0000: 0000000000000037100000371000000000  
0010: 0000000000000000000000000000000000  
0020: 0000000000000000000000000000000000  
0030: 0000000000000000000000000000000004  
...

\*\*\*\*\* SCSI Activity Log (PAGE CODE = 9, Size = 1100) \*\*\*\*\*

Mode Sense Header Byte 2: (byte 000 - 000): 00  
Mode Sense Page 20 Byte 2: (byte 001 - 001): 00  
Mode Sense Page 20 Byte 3: (byte 002 - 002): 04  
...

Raw Data: SCSI Activity Log (PAGE CODE = 9, Size = 1100)

0000: 0000041000008810700000000000000000  
0010: 031d801e0080178017800370000000036  
...

\*\*\*\*\* System Event Log (PAGE CODE = A, Size = 16000) \*\*\*\*\*

No Parse data in `/fnsw/lib/parse\_eventlog8100'

Raw Data: System Event Log (PAGE CODE = A, Size = 16000)

0000: 1001400000000000041000000003dead  
0010: face00008810700000000000000008810  
...

\*\*\*\*\* Log B (PAGE CODE = B, Size = 8000) \*\*\*\*\*

CDE Log Index (byte 000 - 001): 0002  
CDB Log (byte 1-4) (byte 002 - 005): 02470248  
...



---

Raw Data: Log B (PAGE CODE = B, Size = 8000)

0000: 00020247024802480000000000000000

0010: 00000000000000000000000000000000

...

\*\*\*\*\* Media Management Log (PAGE CODE = C, Size = 400) \*\*\*\*\*

Side A Media ID (byte 000 - 008): 38 39 33 38 30 34 31 37 41

...

Side B Relocations Left (byte 08e - 091): 00000000

Raw Data: Media Management Log (PAGE CODE = C, Size = 400)

0000: 383933383034313741000000000000d01

0010: 000036f3000000000000000000000000

...

\*\*\*\*\* LM 8100 Media Specific Disk Info. (SDI) Side B Log (PAGE CODE = 11, Size = 2048) \*\*\*\*\*

SDI Revision Code (byte 000 - 000): 01

Product Identifier (byte 001 - 001): 08

Servo - Writer Number (byte 002 - 002): 01

...

Raw Data: LM 8100 Media Specific Disk Info. (SDI) Side B Log (PAGE CODE = 11, Size = 2048)

0000: 0108010107cf0a0100000000d3214788c

0010: 6464646464646969695e5e5e3e416483

0020: 64646464646464646464646464646464

...

---

## Event Log Parse File

The oddump tool event command formats each log file as defined in the parse file associated with the selected drive. You can modify the format displayed in the log file by editing the appropriate parse file prior to running the command.

The parse file is located in the following directory:

/fnsw/lib            for UNIX platforms

<drive>:\fnsw\lib    for Windows Server platforms

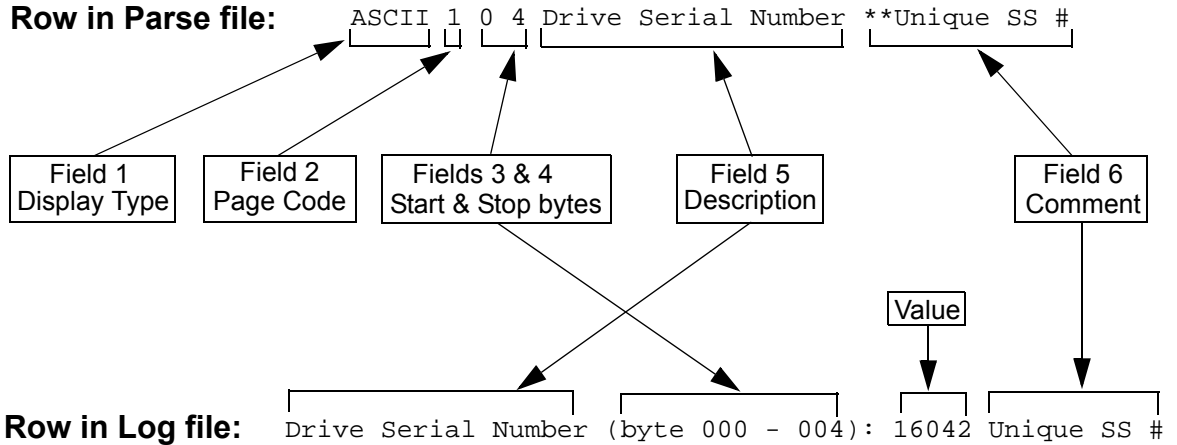
The name of the parse file identifies the associated Plasmon (Philips) drive type:

parse\_eventlog6100   for LMS 6000-series drives

parse\_eventlog8100   for LMS 8000-series drives

When generating the event log file, the oddump event command reads up to six fields in each line and formats the information accordingly. It ignores all comments, preceded by a pound sign (#).

The following illustration identifies each field of the first line oddump reads from the parse file and points out how each field displays in the corresponding row of the event log file.



---

The following example shows a partial listing of the parse file used to define the formatting of an event log file generated for the LMS 8000-series drive. This example omits the complete data where indicated with an ellipsis (...). For a complete data description, see the SCSI Interface Specification for your Plasmon Infinity 6000- or 8000-Series drive (part number 97662164 for the 8000-series drives).

---

```
#
#   Format of parse file:
#
# Any entry preceded by a pound sign, "#" is considered to be a comment and
# will be ignored.
# field 1 - display type field - Currently "ASCII" is the only supported
#           optional display type. If no display type is specified, the data will
#           be displayed in HEX. - (optional field). This entry must start in the
#           first column.
#           (Note: Currently, only five entries use this optional field.)
# field 2 - pagecode in HEX - (mandatory field) This entry must be in the
#           first column if there is no display type.
# field 3 - start byte in DECIMAL - (mandatory field)
# field 4 - end byte in DECIMAL - (mandatory field)
# field 5 - Field description in ASCII - (mandatory field)
# field 6 - Additional comment field ASCII. This should always be preceded by two
#           asterisks "***" - (optional field)
#
#   The fields must be seperated by a space.
#
```

```
#####  
#  
#Drive Statics Log (Page code 1)  
#  
#####  
ASCII 1 0 4 Drive Serial Number  
ASCII 1 5 10 WOODI Firmware Rev  
...  
1 32 34 RWS/A Serial Number  
...  
#####  
#  
#Seek Statistics Log (Page code 2)  
#  
#####  
2 0 3 Side A Seek Errors  
2 4 7 Side A Seek Operations  
...  
#####  
#  
#Shuttle, Motor and Baseplate Statistic Log ( Page code 3)  
#  
#####  
3 0 3 Information Reads On Spin Up  
3 4 7 Focus Not Achieved On Spin Up  
...  
#####  
#  
#DPC Statistic Log (Page code 4)  
#  
#####  
4 0 3 DPC Commands Not Completed  
4 4 7 Shuttle Door Faults  
...
```

```
#####  
#  
#Write Statistic Log for Side A (Page code 5)  
#  
#####  
5 0 3 Drive Not Configured Errors  
5 4 7 Command Not Completed Errors  
5 8 11 Arbitrary Faults  
...  
#####  
#  
#Write Statistic Log for Side B (Page code 6)  
#  
#####  
6 0 3 Drive Not Configured Errors  
6 4 7 Command Not Completed Errors  
6 8 11 Arbitrary Faults  
...  
#####  
#  
# Read Statistics Side A Log (Page code 7)  
#  
#####  
7 0 3 Drive Not Configured Errors  
7 4 7 Command Not Completed Errors  
...  
#####  
#  
# Read Statistics Side B Log (Page code 8)  
#  
#####  
8 0 3 Drive Not Configured Errors  
8 4 7 Command Not Completed Errors  
...
```

```
#####
#
# SCSI Activity Log (Page code 9)
#
#####
9 0 0 Mode Sense Header Byte 2:
9 1 1 Mode Sense Page 20 Byte 2:
9 2 2 Mode Sense Page 20 Byte 3:
...
#####
#
# Failure Analysis Log (Page code B)
#
#####
b 0 1 CDE Log Index
b 2 5 CDB Log (byte 1-4)
...
#####
#
# Media Management Statistic Log (Page code C)
#
#####
c 0 8 Side A Media ID
c 10 13 Side A Raw Media Sectors Left
...
#####
#
# Media Management Statistic Log (Page code D)
#
#####
d 0 8 Media ID
d 9 11 Datae media was tested
d 12 12 Media Tested number
...
```

```
#####  
#  
#   Media Management Statistic Log (Page code E)  
#  
#####  
e 0 8 Media ID  
e 9 11 Datae media was tested  
e 12 12 Media Tested number  
...  
#####  
#  
#   Media Management Statistic Log (Page code F)  
#  
#####  
f 0 0 Media Family  
f 1 1 LM 6000 Identifier  
...  
#####  
#  
#   LM 8000 Media Specific Disk Info Side A (Page code 10)  
#  
#####  
10 0 0 SDI Revision Code  
10 1 1 Product Identifier  
...  
#####  
#  
#   LM 8000 Media Specific Disk Info Side B (Page code 11)  
#  
#####  
11 0 0 SDI Revision Code  
11 1 1 Product Identifier  
...
```



## FINDDoc

Finds all document headers in the image area, which is not a directory of the media. The program prompts you to enter a starting and ending sector number. It then scans the disk in the range specified for document headers. It displays the SSN, document ID, and sector offset of each document header. FINDDoc can also find documents that are not completely written. It does not find any headers in a directory if the directory is in the range.

---

**Note** Since document write operations use a "skip ahead" method of error recovery, using this command does not reliably guarantee finding all documents on a medium.

---

## Findunused

Finds an unused sector past an area of written sectors. The program displays the following prompts:

```
Begin search at? (CR=7000)
End search at? (CR=1280299)
```

If you do not enter beginning and ending search criteria, the search begins at default sector 7000 and ends at default sector 1280299. Sector 7000 is the first sector to which a document can be written and sector 1280299 is the last used sector on a storage medium.

When the find is complete, the following message displays:

```
Search from xxxxx to yyyy found unused at zzzzz
```

where the first unused sector within the specified range (xxxxx to yyyy) was found at zzzzz.

## **FIRMWARE**

(Requires ALLOWUPDATES)

Downloads firmware (also called "microcode") for Plasmon (Philips) 6000- or 8000-series optical drives automatically.

Before using this utility to download the firmware for your Plasmon (Philips) drive, you must load the microcode you received for your drive from Philips, perhaps on a floppy disk or copied to a directory on your server through a file transfer session using ftp or kermit.

Run oddump on the server that contains the Philips drive, issue the ALLOWUPDATES command, and enter the password when prompted. After you enter the FIRMWARE command, the program prompts:

Firmware version (such as c09, please note this is case sensitive)

Enter the firmware file name, identified by the firmware level with extension numbers. For example, for a series with codes beginning with "C01," you would enter:

**C01**

Once you've entered a valid microcode file name, the system prompts you for the directory path containing the firmware files:

New firmware files location (full path name, CR=current directory)

Enter the directory path where the firmware files reside. For example:

**/usr/fnsw/philips**

As the program downloads the firmware files, it displays its progress on the screen, as shown in the following example:

```
Starting DATA DOWNLOAD ....
  Downloading c09.1 ....
  Downloading c09.f ....
  Downloading c09.10 ....
  Downloading c09.11 ....
  Downloading c09.12 ....
  Downloading c09.13 ....
  Downloading c09.14 ....
  Downloading c09.15 ....
  Downloading c09.16 ....
  Downloading c09.17 ....
  Downloading c09.18 ....
  Downloading c09.19 ....
  Downloading c09.1a ....
  Downloading c09.1b ....
  Downloading c09.1c ....
  Downloading c09.30 ....
  DATA DOWNLOAD is completed
  Starting DRIVE DOWNLOAD ....
```

If you attempt to download firmware having the same microcode version as the one you are installing, the program displays the following error message:

```
Starting DATA DOWNLOAD ....
  Downloading c09.1 ....
  LaserDrive is already at the desired series code level c09
  A microcode download is not required.
```

## Fmtdump

Allows you to browse the hierarchical data structure on a storage medium.

---

**Note** If a background job has an ODU file handle open as it sits in the wait loop because an MSAR library is in Backup mode, this command that requires an exclusive drive open access to the surface will fail with the drive busy error.

---

## HArdcopy

Sends a copy of all screen output to a file.

## Help

Displays oddump commands and a brief description of each.

## Highwater

(Requires ALLOWUPDATES)

Displays the value of the next available sector on the media according to the surface record and the surf\_dyn\_info table. These values should be the same unless the medium is in a drive and is currently being written to.

Use this command **only** to increase the value of the next available sector. Do **not** decrease the value of the next available sector unless an error occurred when increasing the value using this command.

After it displays the values for high-water from the tables, the program prompts:

Calculate what HW should be? (y/n):

If you enter **y**, the program inspects the event log and starts the Findunused command from the appropriate point to determine the high-water value. This high-water value should match the value in the tables.

You can then change the value of the next sector in the surface record, the surf\_dyn\_info table, and the event log of the medium. Make changes in the tables only, or in all three places. Changing the value in all three places is recommended. If you do not set the value in the event log and the medium is used on another system, the second system cannot write on the medium at the right location.

---

**Note** If this command attempts to modify the surface file of an MSAR library in Backup mode, it will only calculate the highwater mark of an MSAR file and will not offer a user to update the highwater mark in the surf\_info database.

---

---

**Note** If a background job has an ODU file handle open as it sits in the wait loop because an MSAR library is in Backup mode, this command that requires an exclusive drive open access to the surface will fail with the drive busy error.

---

## HPONLINE

(Requires ALLOWUPDATES)

Enables you to take a specific HP drive offline (library a in the example below) and back online so the HP service technician can repair and replace an optical drive with minimal impact on the FileNet system.

The following example shows the screen display that appears when you enter the hponline command at the oddump command prompt.

---

```
<oddump:lib a>HPONLINE
```

```
Options are:
```

- a. Prepare for drive(s) state change. Run this option  
BEFORE drive state changed to either  
== "Online Pending" or  
== "Offline Good Pending" (only for 330/600fx or up)
- b. Take a specific disabled drive to "Offline Failed"
- c. Bring drive(s) to "Online" or  
"Offline Good" (only for 330/600fx or up)

```
Choice? < >
```

To select an hponline command option, type its option letter in the space provided with the system prompt for choice, as shown in the following example.

```
Choice? <a>  
Successfully access storage library a to detect online drive repair state change
```

The following table describes the options available:

Option	Description
a	Prepares for drive state change. You should run this option before changing the state of a specified drive to either: "Online Pending" or "Offline Good Pending."
b	Takes a specified disabled drive to: "Offline Failed"
c	Brings online drive to "Online" or "Offline Good"

The Image Services System Administrator uses the hponline options to change the states of the drive as needed while the HP service technician repairs and replaces the failed drive.

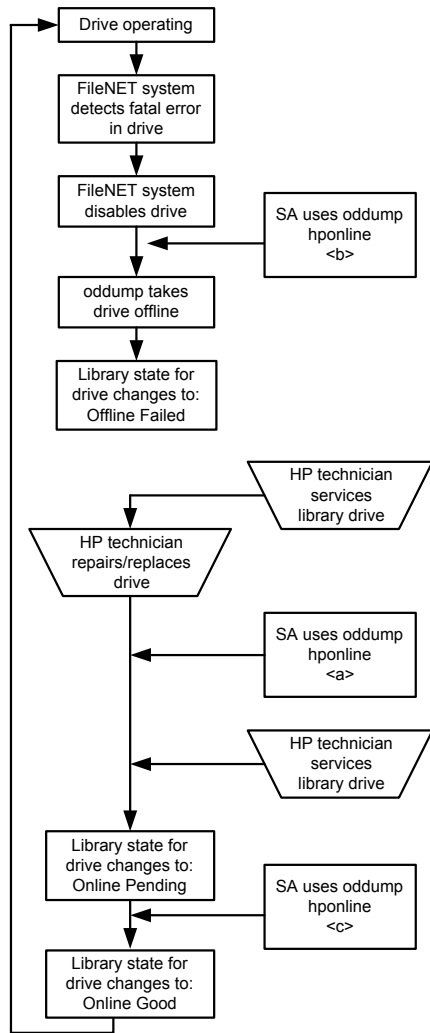
At any time, you can check the current state of the drive through the console display of the HP jukebox. For control panel operational procedures, see the user's guide that came with your HP Optical Jukebox.

The following table identifies which actions to take in response to the possible drive states displayed on the storage library's front panel.

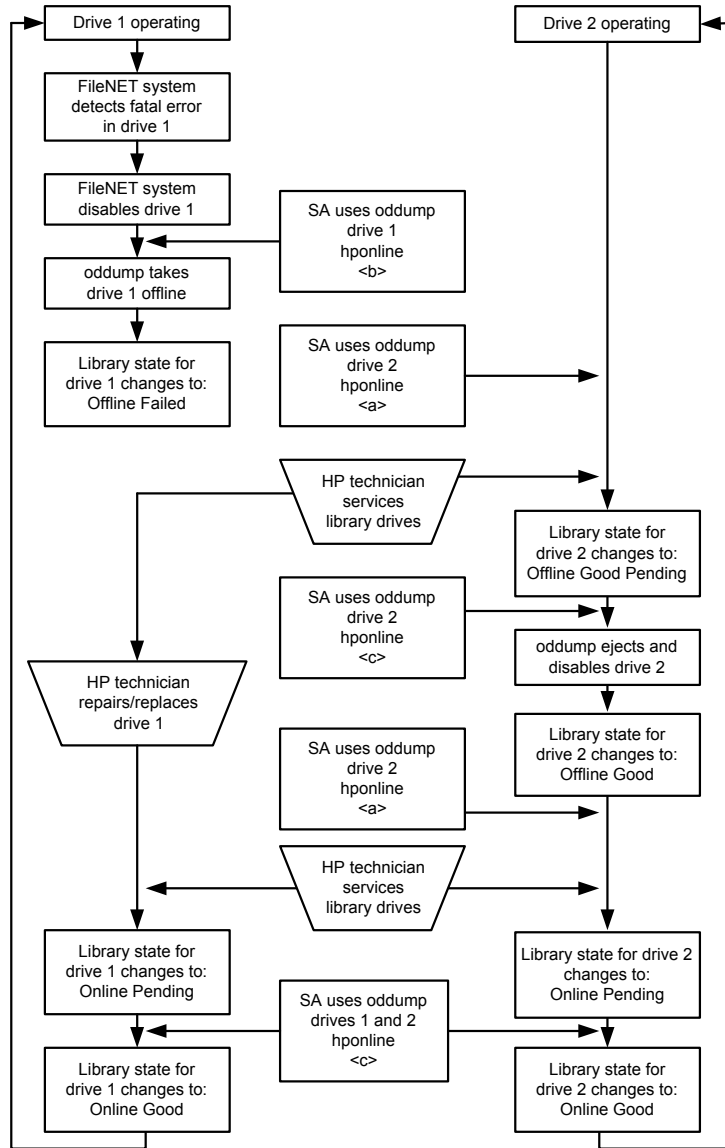
<b>Current State</b>	<b>Action</b>
Online Good	None
Online Pending	Change to Online Good. This applies to either the single- or dual-drive bracket jukebox.
Offline Failed	None
Offline Good Pending	Eject disk, disable drive, and change to Offline Good. This applies only to the dual-drive bracket jukebox.
Offline Good	None

The following diagrams show typical procedures and operations involved when using this tool during an HP service call for a single-drive and dual-drive storage library.





Drive State Diagram for Single-Drive Bracket Storage Library



Drive State Diagram for Dual-Drive Bracket Storage Library

**INITPseudo**

(Requires ALLOWUPDATES)

Erases initialized pseudo-optical media (optical media simulated on magnetic disk media).

**Loaddisk**

Loads a medium from any slot of a drive into the specified drive. Updates the media location tables so that the location of the medium is known to the system.

**LOCKdoor**

(Requires ALLOWUPDATES)

Locks the door of an Optical Disk Unit (OCU) to prevent media removal. When the drive resides in a storage library, the program unlocks the drive and ejects the media automatically.

## Map

Issues a map command to the drive and reports the written/unwritten status of the mapped area. The following prompts display:

```
Sector number (CR=1)
Sector count (CR=32)
Detailed (y/n)?
```

Sector number is the first sector to map. Sector count is the number of 1 KB sectors to map.

If you enter **y** at the Detailed prompt, it displays the status for the low 512 bytes and the high 512 bytes for each sector. If you enter **n**, it assigns only one status: the more significant of the two 512-byte status reports.

For example, if one sector has an error status and another has a written status, the combined status shows an error.

Status bytes in the map indicate the following:

-	unwritten sector
W	written sector
D	written deleted sector
d	deleted written
E	bad sector
e	ecc error
G	gray written, maybe deleted
g	gray written, maybe deleted
P	partially bad
r	sector out of range

---

T	track off
?	unknown

---

**Note** Written and deleted sectors could show up with either a W or a D status when you map multiple sectors but shows a D status when you request a single sector map (sector count = 1).

---

## Quit

Exits oddump

## Rawdump

Dumps selected sectors in hexadecimal and ASCII format. Reads 1 KB of data at a time and displays .25 KB. When you start rawdump, the following prompt displays:

```
R)ead, N)xt, P)rev, F)wd, B)ck, T)ypecast, U)nusedsrch
W)rt, M)dfy, E)rase, (CR=exit)?
```

---

**Note** If you do not use ALLOWUPDATES, some options do not display.

---

The options are:

**Read** prompts for a sector number, reads the sector, and displays the first 256 bytes.

**Next** displays the first 1/4 of the next sector.

**Previous** displays the first 1/4 of the previous sector.

**Forward** displays the next 1/4 of the current sector, or the first 1/4 of the next sector if the last 1/4 of the current sector is displayed.

---

**Back** displays the previous 1/4 of the current sector, or the last 1/4 of the previous sector if displaying the first 1/4 of the current sector.

**Typecast** displays the contents of the buffer as one of several media structures. Select the structure from the following prompt:

L)abel, O)ldlbl, U)pdatelbl, M)sarlbl, I)nitExt, T)ermExt, F)ileDirEnt, D)ochdr, or E)ventEnt?

Selection	Description
Label	2.2 or newer format label
Oldlbl	1.8 format label
Udatelbl	Label for updated short descriptor files
Msarlbl	MSAR label
InitExt	Extent initiator
TermExt	Extent terminator
FileDirEnt	File directory entry
Dochr	Document header (page zero of a document)
EventEnt	Event log entry

---

**Note** Only development Engineering should use the typecast option.

For MSAR surfaces, the label can be read through the typecast option.

---

**Unusedsrch** performs a linear search for unwritten sectors. You must select a drive before you can use unusedsrch. The program prompts you to enter a starting sector number and number of sectors to search:

```
<odddump:lib a>selectdrive
Logical drive number? (0, 1, ..., or CR=none) <1>

Drive successfully opened.  Checking label . . .
<odddump><odddump>r
R)ead, N)ext, P)rev, F)wd, B)ck, T)ypecast, U)nusedsrch
(CR=exit)? <u>
Linear search for unwritten sector
Start search at sector number 1
Number of sectors to search  100
```

**Write** prompts for a sector number and then writes that sector with the contents of the buffer. Before the write operation occurs, you must confirm your selection. (See the caution below.)

---

**CAUTION**

Use the write option carefully. Once you write a sector, you can never modify it.

---

**Modify** allows you to change the data currently in the buffer with ASCII or hexadecimal input of data. You can input decimal numbers by selecting hex input and then suffixing each number with a decimal point. After modifying the buffer, you can check the change by entering **t** for typecast and print as a structure, or by typing **s** for a redisplay of the data in hexadecimal/ASCII.

**Erase** allows you to erase sectors, if the media supports the erase command.

**Note** If this command attempts to modify the surface file of an MSAR library in Backup mode, only the Read, Next, Prev, Fwd, Bck, Typecast and Unusedsrch options will display. Subsequent to these commands this command will not show any options to write, modify or delete an MSAR surface.

---

## **READLoop**

Reads a range of sectors. When you start readloop, the program prompts you to enter the number of sectors in an I/O, the number of times to issue the I/O, and the first sector to read. An I/O error during the read does not abort the readloop. However, you can manually abort the read operation using the Control+c key sequence.

## **REady**

(Requires ALLOWUPDATES)

Issues a ready command to the drive and reports current status of the drive.

## **SETPseudoerr**

Sets errors on a pseudo-optical medium so when a client reads or writes to a sector, the system returns an error to the client attempting the I/O. Used for testing only.

## **SElectdrive**

Changes to a new Storage Library server number or drive number and attempts to read the volume label of the storage medium in that drive. Use this when you manually swap a medium from the drive or when you want to select a different drive.



**SPinup**

Issues a spinup command to the drive. After a storage medium is manually inserted into the drive, use this command to pull the medium into the drive.

**UNLockdoor**

(Requires ALLOWUPDATES)

Unlocks the door and ejects the media of the drive that resides in an Optical Disk Unit (ODU).

**WRITELoop**

(Requires ALLOWUPDATES)

Writes a test pattern at the high-water mark of the media and updates the media high-water mark in both the surface record and the database. Use this command to test the ability to write to a medium without damaging the structures previously written. This command aborts if an I/O error occurs or you enter the Control+c key sequence.

**Note**

---

If this command attempts to modify the surface file of an MSAR library in Backup mode, it will not be allowed to complete.

---

## Checklist

Before using oddump, be aware of the following:

- You must run oddump on the Storage Library server. If your system has more than one Storage Library server, the program prompts you to enter the specific server.
- To run oddump, you must be logged on as a member of the fnadmin group or have a valid fnlogon session with SysAdmin privileges.
- To perform any changes to the storage media files, you must execute the ALLOWUPDATEs command. ALLOWUPDATEs requires a password only available to your service representative.
- You do not need to place a medium in the drive before the program starts; you can use the loaddisk command to move media in and out of the drive. The same command also updates the media location tables to avoid system errors from misplaced media.
- If you manually remove a medium and insert another while running oddump, you should reselect the drive with the selectdrive command. The oddump program caches certain information about a medium (such as surface ID). The selectdrive command updates the cached information with current information.

## Procedure

If you plan to make changes to media files, call your service representative for the ALLOWUPDATES password in preparation for Step 2 below.

- 1 Enter **oddump** at the system prompt. If the server has multiple Storage Library servers, the program prompts you to enter a lower case letter indicating the Storage Library server that contains the desired drive.

After you respond to the user prompts, the prompt changes to <oddump:lib *n*>.

- 2 If you plan to change storage media files, run ALLOWUPDATES and enter the required password.
- 3 Enter commands and options to perform the required tasks.
- 4 Enter **Q** (quit) to exit oddump.

## Related Topics

**[“fnlogon” on page 489](#)**

See the *System Administrator’s Handbook* for information about the Storage Library Control program to enable or disable optical drives. To download IBM FileNet documentation from the IBM support page, see **[“Accessing IBM FileNet Documentation” on page 21](#)**.

# odrecover

## Description

The odrecover tool initiates a docimport background job to start importing documents from storage media. This works the same as calling the stdocimp tool with selected options. It takes the following actions:

- Updates the next available document ID in the scalar\_numbers table, if necessary.

If you have to restore your system from backup tapes, the current document ID on the backup tapes usually does not match the document ID that was current when the database was damaged.

- Avoids importing deleted documents when only one of the document tables (DOCS or DOCTABA) is damaged.
- Verifies that a document is in both document tables.

If the document is in only one table, odrecover adds it to the other table. If the document is not in either table, odrecover does not add the document to either table. If both document tables are damaged, use the `-d` flag to import all documents on the storage media.

- Prevents adding documents to WorkFlo queues.

Re-imported documents should already be in the appropriate WorkFlo queues (or already processed by WorkFlo).

You can have odrecover verify that the associated document classes in the importing and exporting systems match during import. By default, odrecover does **not** require consistent document classes.

**Note**

---

If odrecover encounters an error (such as mismatched document class) during the import of a document into DOCTABA, odrecover still corrects an inconsistency in the databases based on the DOCTABA table in the following manner. If the document exists in DOCTABA but not in DOCS prior to the import, the document is imported to the DOCS table. If the document exists in DOCS but not in DOCTABA prior to the import, odrecover **deletes** the document from the DOCS table. Check for message tuple 90,0,18 in the following logs:

- system event log
- import log file

/fnsw/local/logs/bkglog/ImpLog.xxxxxx for UNIX platforms

<drive>:\fnsw\_loc\logs\bkglog\ImpLog.xxxxxx for Windows Server platform where xxxxxx is the odrecover job number.

To recover the deleted documents, run odrecover `-d`.

---

If you run odrecover with the exact match flag (`-e`), odrecover issues message 90,0,18 when a mismatch exists between the index entries of the importing and exporting systems. The program logs messages in both the system error log and in the Background Job Control import log (ImpLog.xxxxxx) for the odrecover job.

---

Samples from both these logs are shown below:

```
97/02/12 16:23:10.000 90,0,18 <fnswg> docimport 111 (0xa4.84) ...  
Docimport Job#=111, surf=3344; doc 1167956, ssn 8008, Can't get index  
description, name='stringA'
```

```
97/02/12 16:23:10.000 90,0,18 <fnswg> docimport 111 (0xa4.84) ...  
Docimport Job#=111, surf=3344: Translate descriptor to DIR failed.  
Document id=1167956, ssn=8008
```

```
97/02/12 16:23:10.000 90,0,18 <fnswg> docimport 111 (0xa4.84) ...  
Docimport Job#=111: Import: DB insert failed. Document id=1167956,  
ssn=8008
```

```
97/02/12 16:23:46.000 76,0,262 <fnswg> docimport 111 (0xa4.84) ...  
Document import Job 111 completed
```

The following entries are from the Background Job Control import log file:

```
Information for import job number 111 started at Wed  
Feb 12 16:22:40 1997
```

```
Job parameters:
```

```
Input surface id: 3344  
Import from both sides: no  
Import all files: yes  
Document class must match exactly: yes  
Reset surface statistics: no  
Insert into doctaba: yes  
Security options: from document class  
Update scalar numbers tables: yes  
Update WorkFlo queue: no  
Skip import of deleted documents: yeys  
Number of documents to skip on media: 0  
High priority: no
```

```
Errors encountered:
```

```
document id: 1167956, ssn: 8008, error: <90,0,18>
```

```
End of job information:
```

```
Number of errors encountered: 1  
Number of documents read from input media: 610  
Number of documents imported: 0  
Number of duplicate documents in docs table: 1  
Job completed at Wed Feb 12 16:23:46 1997
```

## Use

Use odrecover to import documents from media and rebuild the permanent and index databases when either or both databases have become damaged or corrupted.

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3)

The output of the odrecover tool can report some confusing information when being used with the FileNet P8 Content Federation Services functionality. Since the function of odrecover is to initiate a docimport background job to start importing documents from storage media, a catalog entry is inserted into DOCTABA for each document. If the affected document class and indexes are mapped to Content Engine, then the new CE catalog entry is inserted into the CE database and contains the mapped index values from the media, this can cause a problem when the documents are stored on the Image Services system but indexed on the Content Engine system. In this scenario, the images could be erroneously catalogued twice in the CE catalog.

Additionally, in the instances where documents are indexed on the Image Services system and the odrecover tool is run, existing DOCTABA indexes are overwritten with data from the media. This can overwrite the existing document properties in the CE catalog, if the properties are mapped. Refer to the table below to determine when documents on the IS server will not have the index information associated with them (the **Ns**).

Document Activity	Index on CE	Index on IS
New documents entered via Content Engine	Y	<b>N</b>
New documents entered via Image Services		
• Indexed on both IS and C	Y	Y
• Indexed only on CE	Y	<b>N</b>
Existing document images transferred to CE		
• Indexed on both IS and CE	Y	Y
• Index deleted from IS	Y	<b>N</b>



## Syntax

odrecover [-b] [-d] [-e] <surface ID>

- b** Imports both sides of the specified medium
  - d** Imports all documents on the specified media, regardless of whether the documents are in the DOCS or DOCTABA table. Imports all documents, even those that have been deleted from the databases. This flag is usually specified if both document tables are corrupted.
  - e** Verifies that the importing and exporting systems have an exact document class match for each document to be imported. If mismatch occurs, odrecover logs an error message and the document is imported or deleted based on the DOCTABA table. If you do not specify this option, odrecover continues with the operation even if the document classes do not match.
- <surface ID>** Specifies the media surface to import (for example, 3002)

## Checklist

Before using odrecover, be aware of the following:

- Entering odrecover with no flags displays online help that includes a description of odrecover and a definition of each parameter.
- When you run odrecover, you must import documents from all media that were written since the last backup was made. If you do not import all media, the assignment of duplicate document IDs could result.
- After running odrecover, rescan documents that were in page cache (that is, those documents not yet written to media).
- The odrecover tool does not import documents into WorkFlo queues.
- The default behavior of odrecover is equivalent to running stdocimp with `-nonexactclass`, `-noworkfloqueue`, `-updatesnt`, and `-nondeleted docs` options. You must specify the `-e` option to request an exact document class match and the `-d` option to import the deleted documents from media.

## Procedure

Use the following command to import both sides of a medium and place missing or damaged documents in the DOCS or DOCTABA table:

**odrecover -b <surface ID>**

Use the -b option if documents are missing or damaged in one table, but not the other.

To import all documents, including deleted documents, use odrecover -d:

**odrecover -d <surface ID>**

The -d flag imports the documents even if they already exist in the DOCS or DOCTABA table.

---

**Tip** The odrecover program initiates a document import job. You can monitor the document recovery process like any other document import using Background Job Control.

---

## Related Topics

**[“stdocimp” on page 1249](#)**

See the “Background Job Control” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see **[“Accessing IBM FileNet Documentation” on page 21](#)**.

## oraloglist

**UNIX**

The oraloglist tool is available only on a UNIX-based Image Services server.

### Description

The oraloglist tool provides a simple way of running the commands required to determine the status (enabled or disabled) of the Oracle RDBMS archive logging mode.

Normally, you must run a set of commands, including the "connect internal" command, to determine this setting. These commands require special permissions — only a database administrator should run them. With oraloglist, any user can easily determine the archive logging status.

### Use

Use oraloglist to determine the archive logging status. For example, use oraloglist to verify that the database log mode is set to ARCHIVELOG mode before a backup. (Refer to your *System Administrator's Companion for UNIX* or *System Administrator's Companion for Windows Server* for more information. To download IBM FileNet documentation from the IBM support page, see [\*\*"Accessing IBM FileNet Documentation" on page 21.\*\*](#))

FileNet software must be active or the index database must be up to use this command.

### Syntax

**["oraloglist" on page 940](#)**

## Sample Output

Enter the oraloglist at the command line as shown in the sample output below:

```
corona(fnsw)/usr/fnsw> oraloglist

Database log mode           No Archive Mode
Automatic archival         Enabled
Archive destination        /fnsw/local/ora_arch/arch
Oldest online log sequence 27
Current log sequence       28
```

## Checklist

Before you use oraloglist, be aware that FileNet software must be active or the index database must be up.

## Procedure

No specific procedure is required.

# osschk

## Description

The osschk (optical subsystem check) tool provides summary information contained in FileNet error logs for optical storage and retrieval (OSAR) libraries. The osschk tool normally resides in the following directory of the optical subsystem server:

`/fnsw/support`

The system writes output from osschk to the directory where the elogs are found:

`/fnsw/local/logs/elogs/`

The first time you execute osschk on the server, it prompts for a site name used to identify that particular system. The site name is used only by osschk and is used on the osschk summary report. The program automatically configures itself to match the number of storage libraries and optical drives. On multi-server Image Services configurations, the program will analyze the elogs on one server at a time. The program will prompt for the station number of the server.

In response to the osschk command, the program prompts for starting and ending dates of the period of the FileNet error logs to check.

The osschk tool summarizes all OSAR and optical information into the following two files:

- ogndetmmdd\_n     Contains only OSAR and optical information from error logs.
- ognsummmdd\_n    Contains OSAR configuration, optical subsystem module stamps, and a breakdown of each of the errors encountered by type.

The osschk tool generates file names using the following format:

ogntypmmdd\_n

where:

- og    =    Prefix to file
- n     =    Station number
- typ   =    File type, either detail (det) or summary (sum)
- mm   =    Month generated (01-12)
- dd   =    Day generated (01-31)
- \_n   =    A if the -x parameter is used to generate the summary report.
- \_n   =    Run of the day (1-4)

For example, a file named **og1sum0325\_1** is the osschk summary file generated on station number 1 on March 25 during the first run of the day.

When finished generating the files, the program pauses to allow you to view the summary and detail files. If you enter **yes** (default is no) to view the files, it displays a less prompt, enabling forward and backward viewing of the summary file first. A **q** ends the less mode.

After displaying the summary file, the program pauses again to allow you to view the detail file. If you enter **yes** (default is no) to view the detail file, it displays a less prompt, enabling forward and backward viewing. A **q** ends the less mode and quits the program.

---

**Note** If the **-x** argument is used, then the program will NOT give a prompt or allow the viewing of the summary or detail file.

---

Any summary or detail files remain on the server for approximately two months before the program automatically removes them. When the osschk program is run, it looks for summary and detail files that are within a two-month-old range and erases them. For example, if the program is run in June (Month 06) all the summary and detail files made in the April (Month 04) of the same year are erased.

## Use

Use osschk to determine the hardware performance of the Image Services server's optical subsystem. You can analyze the generated output to check the subsystem performance and look at any points requiring attention or repair.

The osschk utility can summarize elog data in different situations:

- To summarize last week's errors:

> **osschk -x**

Enter the dates of the last week and today.



- To summarize elogs that were transferred from some other machine and placed in the /sitea/elog directory:  
  
    > **osschk -ka /sitea/elog**  
    Enter the dates of the time period to scan.
- To summarize the error for the day, assuming it is close to the end of the day (11:30, etc.)  
    > **osschk -x**

## Syntax

osschk [-v] [-s] [-d] [-k] [-a <elog directory path>] [-x] [-h]

---

**Note** The options for the osschk utility are mutually exclusive, unless noted otherwise.

---

- v Displays previously generated summary and detail files.
- s Generates a file of the storage library configuration and related optical subsystem module stamps.
- d Disables the summary and detail files. The program will prompt the user and delete the files as directed.
- k Tells the program **NOT** to use the stamp and configuration information from the server where the utility is running. osschk will use the data in the scanned elogs to derive the apparent configuration to allow the generation of the summary.

---

**Note** When the -k option is used, stamp data for the various system modules will **NOT** be reported.

---

**-a <elog directory path>** Allows the user to specify in the next parameter the directory of the elogs to be analyzed.

---

**Note** Sometimes elogs from one system will be transferred into a subdirectory on another system for analysis. The -a option can be specified with the -k option and will cause the program to examine the elogs in a subdirectory and to report only on the OSAR libraries that have errors in the elog files.

---

**-x** Used when the osschk utility is included in a script run automatically from another script. The option utilizes no screen input from the user and provides no screen output to the user. The elog that is summarized is from the day it is run. In addition, the files produced will have the \_A suffix in the file name.

**-h** Displays a summary of the options that control osschk.

## Sample Output

The following example shows a summary file generated for a site with one OSAR library on station 1. It does not show the detail file, but contains only lines from the error log files which involve the optical sub-system. The example has filtered out other lines.

```
cat og1sum032503_1
+++++
Site_Name..  A Site Name

+++++
Platform: AIX   OS: 4.2.1.0   IMS: 3.5.0
- - - - -
Modules in.. /fnsw/bin
    dtp
system 3.5.0.224(0) (userid 8, Thu Nov 15 12:33:14 2001)
SubSys: os, Rel_type: rel_aix, SCR#: 143325, mode: 100755, size: 302856
    dtp_tran
system 3.5.0.240(3) (userid 8, Thu Mar  7 11:31:57 2002)
SubSys: os, Rel_type: rel_aix, SCR#: 150069, mode: 100755, size: 70034
    dsched
SubSys:
    oddump
system 3.5.0.241(3) (userid 8, Wed Mar 20 14:52:21 2002)
SubSys: os, Rel_type: rel_aix, SCR#: 150770, mode: 100755, size: 248335
    fnsod
system 3.5.0.149(0) (userid 8, Fri May 26 13:55:35 2000)
SubSys: kd, Rel_type: rel_aix, SCR#: 109458, mode: 100755, size: 163694
    Companion Module in.. /etc/drivers
        /etc/drivers/fnsod
system 3.5.0.149(0) (userid 8, Fri May 26 13:55:35 2000)
SubSys: kd, Rel_type: rel_aix, SCR#: 109458, mode: 100755, size: 163694
    FNPoll
system 3.5.0.122(3) (userid 8, Tue Nov  2 16:42:48 1999)
SubSys: kd, Rel_type: rel_aix, SCR#: 97690, mode: 100755, size: 60501
```

```

Xslc
system 3.5.0.233(0) (userid 8, Tue Jan 22 09:51:08 2002)
SubSys: xc, Rel_type: rel_aix, SCR#: 146770, mode: 100755, size: 1388323

```

Associated Module in.. /fnsw/lib/X11/uid

```

Xslc.uid
system 3.5.0.180(0) (userid 8, Thu Jan 11 14:13:06 2001)
SubSys: xc, Rel_type: rel_aix, SCR#: 121594, mode: 100644, size: 344179

```

Modules in.. /fnsw/lib/shobj

```

ARM
SubSys:
ODX
SubSys:
ODU
system 3.5.0.217(0) (userid 8, Thu Sep 27 10:43:51 2001)
SubSys: os, Rel_type: rel_aix, SCR#: 139584, mode: 100755, size: 291534
FCL
system 3.5.0.173(2) (userid 8, Mon Nov 13 12:38:15 2000)
SubSys: mv, Rel_type: rel_aix, SCR#: 118027, mode: 100755, size: 40631
SysV
system 3.5.0.254(25) (userid 8, Mon Aug 12 15:58:20 2002)
SubSys: mv, Rel_type: rel_aix, SCR#: 159279, mode: 100755, size: 408603

```

-----  
The number of Servers configured on this system: 1

OSAR Configuration for... stimpv (AIX)  
-----

Library Unit: 1 (a): FileNet Optical Library 50GTL/GTS/HTL/HTS Slots: 50  
Mode: Normal

```

Device Addr: 0 5 6 0
osarb is linked to.. /dev/fnsod.0,5,6,0

```

Drive 1: Plasmon LMS LD8100 30 GB Optical Drive

```

Device Addr: 0 5 1 0
oddb1 is linked to.. /dev/fnsod.0,5,1,0

```

-----  
+++++

---

Command line : /fnsw/support/osschk

This osschk report was run at..  
Tue Mar 25 15:14:12 PST 2003

Report Period Start.. 01/01/2003 End.. 01/30/2003

Site\_Name.. A Site Name

Directory of scanned elogs: /fnsw/local/logs/elog  
Deriving the list of OSARs from the error logs..  
(No system configuration files were used to  
generate this report.)

OSAR Server.. stimp  
List of OSAR Libraries found by scanning elogs : A

\*\*\*\*\* Summary Info for Storage Library.. 'A' \*\*\*\*\*

Library A wake-up [RF389] Qty: 0

Total RFxxx error count for Library A = 0

Total Backup Mode count for Library A = 6

RB153 Qty: 1 RB153\_\_\_\_\_The Gripper 2 Slot Align Sensor  
did not detect the alignment target after  
completion of the mechanism move. A routine  
that hunts for the target over a limited area  
was successful in finding it however and the  
command completed normally.

RB261 Qty: 5 RB261\_\_\_\_\_Rack 1 Interrupter Sensor was  
not blocked while a cartridge was moved  
between the gripper and the Storage slot.

Backup mode counts at Drives:

Drive 1 = 0

Drive 2 = 1

Drive 3 = 0

```

Backup mode counts at Slots :
  Slots 00 to 09 .. Qty: 1
  Slots 10 to 19 .. Qty: 4
  Slots 20 to 29 .. Qty: 0
  Slots 30 to 39 .. Qty: 0
  Slots 40 to 49 .. Qty: 0
  Slots 50 to 59 .. Qty: 0
  Slots 60 to 69 .. Qty: 0
  Slots 70 to 79 .. Qty: 0
  Slots 80 to 89 .. Qty: 0
  Slots 90 to 99 .. Qty: 0
  Slots 100 to 109 .. Qty: 0
  Slots 110 to 119 .. Qty: 0
  Slots 120 to 129 .. Qty: 0
  Slots 130 to 139 .. Qty: 0
  Slots 140 to 149 .. Qty: 0

```

```

Backup mode counts at I/O Station :
  Input and IX.. Qty: 0
  Output      .. Qty: 0

```

```

Fault counts at I/O Station :
  Input.. Qty: 0
  Output. Qty: 0

```

Times Slot(s) were disabled. = 0

```

Storage Library.. A
  Enabled = 0      Disabled = 0  using XSLC
  Enabled = 0      Disabled = 0  using DOC_tool

```

The following describes disable & enable counts.  
 XSLC = Request to Disable/Enable using XSLC.  
 DOC = Request to Disable/Enable using DOC\_tool.  
 S/W = Disabled by software due to previous/motion error.

Requested	----	Disable via	----	or	-Enable via-
Drive #	XSLC	DOC	S/W		XSLC DOC
Drv_1	0	0	5		4 0
Drv_2	0	0	5		3 0
Drv_3	0	0	4		3 0

```

---- Current drive status:
New   Disabled Configured
Drive 1:  Disabled Configured
Drive 2:  Disabled Configured
Drive 3:  Disabled Configured
Drive 4:  Disabled Configured
Drive 5:  Disabled Configured

```

```

-----
Move count totals ( ALL libraries combined ).
[ Manually view logs for specific library. ]
    Gripper -> Slot   = 0
    Gripper -> Drive  = 0
    Gripper -> I/O    = 0
    Slot/Dr  -> Grip  = 0
    Slot -> Drive    = 6
    Slot -> I/O     = 0

```

```

-----
FileNet elog totals ( All libraries combined ).
Occurrences of Lost disk.. = 0
Occurrences of Found disk. = 0

```

```

-----
                          SCSI OSAR Drive Number:
Sys_Error   SCSI_Sense  Drv 1   Drv 2   Drv 3   Drv 4   Drv 5   Drv 6
'disabled'  no_sns_data   5       5       4       0       0       0
'baddata'   03,09,00,0e  0       0       4       0       0       0
'drivedown' 04,44,00,38  5       6       2       0       0       0
'drivedown' 04,44,00,39  1       0       1       0       0       0
'drivedown' no_sns_data   5       3       6       0       0       0
'drivenotrd 02,04,00,36  0       5       0       0       0       0
'drivenotrdy' 02,04,00,62  0       2       0       0       0       0
'drivenotrdy' 04,44,80,ff  0       1       0       0       0       0
'drivereset' 04,44,00,39  0       0       1       0       0       0
'drivereset' 06,44,80,36  0       0       1       0       0       0
'eof'       no_sns_data   0       2       0       0       0       0
'unusedsect' no_sns_data   0       0       1       0       0       0

Totals for OSAR Library A   16      24      20      0       0       0
+++++
----- End of Report -----

```

## Procedure

**Note** Document Services needs to be up and running on the server for osschk to run.

---

- 1 Enter **osschk** at the OSAR server's prompt.
- 2 If this is the first time osschk is run on a server, respond to the prompt to enter the name of the server.
- 3 Respond to prompts for starting and ending dates of the period of FileNet error logs you want to check.
- 4 Enter **yes** to view the summary file and use the less function to scroll through the displayed file.
- 5 Enter **q** to quit viewing the summary file.
- 6 Enter **yes** to view the detail file and use the less function to scroll through the displayed file.
- 7 Enter **q** to quit viewing the detail file and end the program.

To view these files again or any previously generated files, use **osschk -v**.



## Related Topics

[“DOC\\_tool” on page 326](#)

[“getreports” on page 523](#)

[“oddump” on page 886](#)

[“perf\\_report” on page 968](#)

---

## perf\_mon

### Description

The perf\_mon tool collects statistical data that is kept by the kernel, and the following shared libraries ("abstracts"): counter (CNT), Network Clearinghouse (NCH), Multi-keyed Files (MKF), WorkFlo Queue Services (WQS), Configuration Database (CDB), Security (SEC), and WorkFlo (WQM). It gathers statistical data from the following areas: cpu usage for kernel and non-kernel activities; overall file system activities; kernel file system activities; network I/O statistics; database server counts; document services counts, RPC statistics, and security statistics.

When FileNet software starts or restarts, perf\_mon starts automatically with a script file

```
perfmon -fperfmon.script
```

The script file resides in:

```
/fnsw/lib/perf/perf_mon.script            UNIX platforms
```

```
<drive>:\fnsw\lib\perf\perf_mon.script    Windows Server platforms
```

The script controls the default sampling intervals to run every 15 minutes during the normal 8:00 a.m. to 6:00 p.m. day and every two hours during the night. On the weekends, samples also run every two hours.

Use an editor to view the script to see how automated scheduling works.

You can change parameters defined in the perflog file, as described in **[“SET \[<variable>\] \[<value>” on page 961.](#)** If you want to change

---

only the number of records stored in the perflog file, see [“log\\_create” on page 696](#) for details on changing the maximum number of records.

The perf\_mon tool collects the samples in a circular log file:

`/fnsw/local/sd/1/perflog` for UNIX platforms

`<drive>:\fnsw_loc\sd\1\perflog` for Windows Server platforms

You can also run perf\_mon from the command line to debug performance problems. If you run perf\_mon from the command line, the log file in which the program saves the collected data is the one you name in your capture command (for example, sample <filename> 60 10).

You can set up a schedule of monitoring activity. Use the perf\_mon schedule command to set up daily or weekly times when you want perf\_mon to run automatically. First, set up the daily schedules. Then, link each weekday with one daily schedule. Finally, invoke the poll command. Poll determines the current time and weekday, and monitors the system according to the schedule that you linked to each weekday. When you use scheduling, you must account for a full 24-hour period, starting at midnight. Poll does not wrap around to the next day.

For example, the following commands (with appropriate comments) set the system to monitor at 30-minute intervals from 8:00 a.m. to 5:00 p.m. and at two-hour intervals from 5:00 p.m. to 8:00 a.m. the next day.

```
"schedule 1 00:00:00 02:00:00" -- from midnight on, monitor every two hours
```

```
"schedule 1 08:00:00 00:30:00" -- from 8 a.m. on, monitor every 30 minutes
```

```
"schedule 1 17:00:00 02:00:00" -- from 5 p.m. on, monitor every two hours
```

After the program collects the data, you can run `perf_report`, the performance report writing tool, to gather and format the statistics into reports. (Refer to [“perf\\_report” on page 968.](#))

## Use

When you start or restart FileNet software, `perf_mon` starts automatically and collects performance data. However, you can run `perf_mon` from the command line with special commands and parameters to debug performance problems. You can run `perf_mon` interactively from your terminal or run it using a command file.

## Syntax

You can use one of the following two command forms to start `perf_mon`:

- `perf_mon`

`perf_mon` starts and runs in interactive mode.

- `perf_mon -f<script> [&]`

`perf_mon` starts and accepts input from a command file.

**-f<script>** Reads `perf_mon` input commands from the file specified by `<script>` instead of from standard input

**&** Runs `perf_mon` in the background  
You must use the `-f<script>` option to run `perf_mon` in the background.

## Commands

There are two categories of commands you can use with perf\_mon. Capture Commands collect specific information. Command Dispatcher Commands operate within perf\_mon.

To list these commands interactively, type ? at the <perf\_mon> prompt, as shown in the example below. A description of each command follows the example.

---

```
perf_mon> ?
```

### Capture Commands

```
=====
+ filename                Sample system state
ADD_EVENT event           Add an event category
ADD_QUEUE systemname queueName  Add workflo queue to the list
LINK [day# sched#]       Link a day to a schedule
POLL filename [n_samples] Capture according to the schedule
SAMPLE filename delay n   Capture at a regular interval
SCHEDULE # [time intv]    Inspect or modify schedule
SHOW_DISKS                Show the disk list
SHOW_EVENT                Show enabled event categories
SHOW_QUEUE                Show the Workflo queue list
UNSCHEDULE # [time]       Clear or delete schedule entry
WATCH delay n_samples [> filename] Watch at a regular interval
```

### Command Dispatcher Commands

```
=====
! command line            Issue commands to a shell
?                          List this information
ECHO                       Echo command line
EXIT                       Exit this program
HELP                       List this information
QUIT                       Exit this program
READ file                  Redirect input from 'file'
SET variable [value]      Set or inspect a variable
```

---

## Capture Commands

### **+ <filename>**

Takes one sample of system data and saves it to the specified file

### **ADD\_EVENT <event>**

Adds an event category beyond those listed in the events list when perf\_mon starts

### **ADD\_QUEUE <systemname> <queuename>**

In addition to the WorkFlo general queue, adds one specific WorkFlo queue <queuename> to the list to be monitored. <systemname> is the name of the WorkFlo system. You can invoke add\_queue on up to 5 different WorkFlo queues.

### **LINK [<day#> <sched#>]**

Link a day to a schedule, where day# is the number of the weekday. Sunday is day 0, Saturday is day 6.

sched# is the schedule number as defined with the SCHEDULE command. For example, to link Monday and Tuesday to schedule 1 (as set up with the schedule command), use the following commands:

```
link 1 1
```

```
link 2 1
```

### **POLL <filename> [<n\_samples>]**

Begins data capture according to the schedule. <filename> is the name of the file where the system saves the data. <n\_samples> is a decimal value for number of samples. Poll uses the schedules that are

currently set up by the schedule and link commands. For example, the following command uses schedule 1, monitors 50 times and stores the data in a file called savefile:

**poll savefile 50**

**SAMPLE <filename> <delay> <n>**

Captures data at a regular interval specified in seconds by the <delay> parameter, for a specified number of times defined by the <n> parameter. The data is stored in the specified file. For example, the following command saves data in a circular log file named savefile. Data collection occurs every 5 seconds for 10 times:

**sample savefile 5 10**

**SCHEDULE # [<time intv>]**

Adds an entry to a schedule of monitoring activity. You can also use this command to inspect or modify an existing schedule. You can specify a maximum of 7 schedules. Each schedule can have a maximum of 6 entries.

**#** is a new or existing schedule number.

**<time intv>** is the start time followed by the interval time.

For example, to add an entry to schedule 1 to begin monitoring at 10:00 a.m. and monitor at one-hour intervals, enter the following command:

**schedule 1 10:00:00 01:00:00**

Use the LINK command to map days of the week with the schedule number.

**SHOW\_DISKS**

Shows the disk list being monitored

**SHOW\_EVENT**

Shows the enabled event categories

**SHOW\_QUEUE**

Shows the Workflo queue list being monitored

**UNSCHEDULE # [<time>]**

Clears or deletes a schedule entry from the schedule number specified by #

<time> is the time to be deleted from the schedule, in the form hh:mm:ss. If you do not specify [time], perf\_mon deletes the entire schedule.

**WATCH <delay> <n\_samples> [> <filename>]**

Displays CPU activity at a regular interval

<delay> is the interval (in seconds) to invoke the read/write of CPU activity.

<n\_samples> is the number of samples to "watch."

> <filename> redirects the output to the specified file.



## Command Dispatcher Commands

### **! command line**

Issues commands to a shell

### **?**

Lists help information

### **ECHO**

Echoes the last command entered on the command line

### **EXIT**

Exits perf\_mon

### **HELP**

Lists help information

### **QUIT**

Exits perf\_mon

### **READ <filename>**

Redirects input from the specified file name

### **SET [<variable>] [<value>]**

Sets or inspects a variable having the value specified in <value>. If you do not specify <value>, the current value of that variable displays. If you do not specify <variable>, perf\_mon inspects all variables.

The following table defines each of the set command variables, identifies the default and available values, and notes values you should not change.

Variable	Description	Default	Options
Alarm	<p>Switch that turns perf_mon's alarm mechanism on or off. When set to TRUE, perf_mon triggers an alarm under one of these conditions:</p> <p>The system write rate is greater than the value of Max_Syswrite.</p> <p>The system read rate is greater than the value of Max_Sysread.</p> <p>The system idle time is less than the value of Min_Sysidle.</p>	FALSE	TRUE or FALSE
Action	The action to take when an alarm condition is detected. The value of this variable is used as the argument to a system command.	"(date; ps alx) > AlarmLog"	Applicable System Command String
Def_Interval	The default interval value (in seconds) used by perf_mon when polling (see the <b>POLL &lt;filename&gt; [&lt;n_samples&gt;]</b> command) if a schedule entry isn't found.	3600	Number in Seconds
EventLogging	Switch that determines whether perf_mon will log events. When set to TRUE, perf_mon collects ELA (RPC) event data.	TRUE	TRUE or FALSE

Variable	Description	Default	Options
Listener	Turns on the FileNet System Manager listener agent. For more specifics on the Listener variable, go to <a href="#">“FileNet System Manager” on page 414</a> . For an example of temporarily turning on the FileNet System Manager listener agent, go to <a href="#">“Make the System Manager Listener Agent Run Temporarily” on page 415</a> .	FALSE	TRUE or FALSE
Max_Syswrite	System write I/O alarm threshold, in writes per second, if non-zero. (See Alarm description.)	0	Number Writes per Second
Max_Sysread	System read I/O alarm threshold, in reads per second, if non-zero. (See Alarm description.)	0	Number Reads per Second
Min_Idle	System idle time alarm threshold, in percent, if non-zero. (See Alarm description.)	0	Number in Percent
MKF_Stats	Switch that determines whether perf_mon will collect MKF statistics. The default value is TRUE if the server has any MKF databases configured on it.	TRUE	TRUE or FALSE
Num_Records	Maximum number of records allowed in perf_mon log file	2048	Number of Records
Record_Length	Number of bytes allowed for each record stored in perf_mon log file	1024	DO NOT CHANGE
Verbose	Additional output, primarily used for debugging. If TRUE, perf_mon adds this data to the log file.	FALSE	TRUE or FALSE
Workflo	Switch that determines whether perf_mon will collect WorkFlo Queue Services (WQS) data. The default value is TRUE if the server has WorkFlo Queue Services configured on it.	TRUE	TRUE or FALSE

## Examples

The set command displays a list of all current variable settings:

```
perf_mon> set
Alarm = FALSE
Action = "(date; ps alx) > AlarmLog"
Def_Interval = 3600
EventLogging = TRUE
Max_Syswrite = 0
Max_Sysread = 0
Min_Idle = 0
MKF_Stats = TRUE
Num_Records = 2048
Listener = TRUE
Record_Length = 1024
Verbose = FALSE
Workflo = TRUE
```

The following sample command captures a sequence of two data collections at a constant interval of two seconds and saves the data to file testfile. The return of the perf\_mon prompt indicates that the data collection is complete. The quit command terminates perf\_mon.

```
perf_mon> sample testfile 2 2
perf_mon> quit
```

To see the data in file testfile, run perf\_report:

**perf\_report -rf testfile**

(For more information, see [“perf\\_report” on page 968.](#))

The following commands establish monitoring schedules 1 and 2, linking Monday through Friday to schedule 1 and Saturday and Sunday to schedule 2. The last command, poll, limits monitoring to no more than 300 times and saves the data to a file named savefile.

```
schedule 1 00:00:00 02:00:00
schedule 1 07:55:00 00:15:00
schedule 1 18:15:00 02:00:00
schedule 2 00:00:00 03:00:00
link 1 1
link 2 1
link 3 1
link 4 1
link 5 1
link 6 2
link 0 2
poll savefile 300
```

## Checklist

Before you use perf\_mon, be aware of the following:

- perf\_mon starts automatically at system start or restart. However, you can run it from the command line at any time to capture data related to a specific performance problem.
- If you use scheduling, you must account for a full 24-hour period, starting at midnight.
- To generate reports with data perf\_mon has collected, run perf\_report. You can view the generated report files using a text editor.

## Procedure

- 1 Start perf\_mon in the foreground or the background.
  - To run perf\_mon in the foreground, enter **perf\_mon** at the command line. At the <perf\_mon> prompt, enter perf\_mon Capture or Command Dispatcher commands.
  - To run perf\_mon in the background, create a script file (for example, pmon\_script) containing the commands just as you would enter them interactively. Then, invoke perf\_mon as follows:

**perf\_mon -fpmon\_script &**

The example script name is pmon\_script. Use your script name when you start perf\_mon.

- 2 When data collection is complete, run perf\_report to view the captured data.

## Related Topics

[“log\\_create” on page 696](#)

[“log\\_dir” on page 699](#)

[“log\\_extract” on page 702](#)

[“perf\\_report” on page 968](#)

[“FileNet System Manager” on page 414](#)

# perf\_report

## Description

The perf\_report tool generates reports you can use to analyze performance. It provides report generation capabilities for historical data logged in the system performance log file by the perf\_mon tool. (Use the log\_dir tool to see the entries in the log file.) The data logged by ELA (the event logging abstract) includes Remote Procedure Call (RPC) data and some local events. ELA logs data on a per-server basis.

You can request default reports, canned soft reports, or user-defined soft reports. The perf\_report tool generates reports that summarize system activity and/or a set of data items for various subsystems such as document services and WorkFlo. Default report types are:

- **abs** Full Absolute Report
- **ful** Full Report

For details on report types, see [“Soft Reports” on page 972](#).

If you want to define your own custom soft reports, you can:

- Define new data variables from the set of basic system variables
- Do arithmetic
- Define reports that are tables, or collections of tables, containing basic or user-defined variable data

The **perf\_report -a** command writes a collection of reports to text files stored in the reports directory. After generating reports, perf\_report



displays the file path to the reports directory, followed by a list of generated reports.

You can view these reports using view commands or text editors available through your server's operating system.

The perf\_report tool calculates the difference between two records in the log. In the full report and most canned soft reports, perf\_report prints the delta value. An exception is the absolute report (see the `-rabs` option). For soft reports, you can choose between delta and absolute values for each data item in the record. With the `-i` option, you can also specify an interval number to generate reports between every `n`th record. Options are available so you can limit the total number of reports or limit the number of reports to generate each day.

## Use

Use perf\_report to generate reports (using data gathered by the perf\_mon tool and written to a log file) to assist you with performance analysis tasks.

## Syntax

To generate default or specific soft reports, use the following format with appropriate options as defined in "Options."

```
perf_report <logfile> [-i#] [-m#] [-ph] [-pd] [-d#] [-tz#] [-tsv] [-csv]
[-p] [-r<rname1>] [-r<rname2>] [-fd<ela_dir>] [-fr<reports_file>]
[-br<date>+<time>] [-er<date>+<time>] [-b<time>] [-e<time>]
```

---

**Note** No spaces are allowed between an option and its parameter.

---

## Options

- <logfile>** Specifies the name of the performance monitor log file from which to extract log records  
A log file name is required.
- i#** Sets reporting interval to the value specified by #  
If not specified, default interval is 1. perf\_report generates a report every **n** records found in the performance log, where **n** is the value specified in #. The default of 1 causes perf\_report to use every entry when generating reports.
- m#** Sets maximum number of generated reports to the value specified by # (for example, m4)
- ph** Sets printout to display in hourly increments
- pd** Sets printout to display in daily increments
- d#** Sets maximum number of generated daily reports to the value specified by # (for example, d2)
- tz** Specifies delta in hours between local time and capture log time (for example, -24, 24)
- tsv** Formats report using tabbed characters as column separators. This enables you to import generated reports into applications, such as Excel spreadsheets.
- csv** Formats report using commas as column separators. Used for Scout Import.
- r<rname1>** Produces the default and/or soft reports specified in <rname1>.  
**-r<rname2> ...** You can specify a maximum of 32 report names

- 
- <name> =** abs | ful
- rabs** Produces Full Absolute Report
- rful** Produces Full Report
- r<soft\_rtype>** Produces the specified soft report (for example, `-rds` produces a Documents Services report). For a description of predefined soft report types, see [“Predefined Soft Reports” on page 973](#).
- s[<name>]** Displays symbolic names of the specified category. If no category is specified, all categories display.
- b<time>** Sets statistics daily begin time. Specify <time> in numeric <HH>[:<MM>] format.
- e<time>** Sets statistics daily end time. Specify <time> in numeric <HH>[:<MM>] format.
- br<date>+<time>** Sets reporting interval lower bound. Specify <time> in numeric <HH>[:<MM>] format. Specify date using one of these formats:
- MM/DD/YYYY (years from 1970-2037)
  - MM/DD/YY (years from 70-99 for 1900's or 00-37 for 2000's)
  - MM/DD (system uses current year)
  - yesterday
  - today
  - (none specified uses today's date, but requires time)
- For example, to specify <date>+<time>, you can enter the following option with the `perf_report` command:
- `-br09/14/1998+14:00`

- 
- er<date>+<time>** Sets reporting interval upper bound (uses the same date and time formats described for -br)
- fd<ela\_dir>** Specifies the event dictionary file where <ela\_dir> is the full path name to the file. If a file is not specified, the default file is:
- /fnsw/lib/perf/ela/elacodes for UNIX platforms  
<drive>:\fnsw\lib\perf\ela\elacodes for Windows Server platforms
- fr<reports\_file>** Specifies report description file

## Soft Reports

A "soft report" contains performance statistics defined in a report description file that specifies the report format, parameters, and variable constructs. "Canned" soft reports are available in the default reports file:

/fnsw/lib/perf/reports for UNIX platforms  
<drive:>\fnsw\lib\perf\reports for Windows Server platforms

When generating soft reports, perf\_report scans the report description file until it finds the specified report type.

To generate individual predefined reports, you can enter the following command:

```
perf_report -r<soft_rtype>
```

where: <soft\_rtype> is the report type.

For example, to generate a CSS Summary Report, you would enter the following command:

## perf\_report -rcss

**Predefined Soft Reports**

The following table shows a list of predefined soft reports. For report samples, see [“Sample Reports” on page 1000](#).

## Predefined Soft Reports

Type	Report Name
css	CSS Summary Report
asum	AIX Processor Summary Report
hpux_sum	HPUX Processor Summary Report
sol_sum	Solaris Processor Summary Report
osum	OLI_5000 Processor Summary Report
ntsum	NT Server Processor Summary Report (used by all Windows Server processors)
usum	Unisys Processor Summary Report
ds	Document Services Summary
ol	OSAR Load Summary
which	Which SCoUT Services
scout	Scout Overlay Sheet Data
sv	Service Activity
rsv	Client Service Activity
csv	Cumulative Service Activity
rcsv	Remote Cumulative Service Activity
bs	Batch Entry Report
rbs	Batch Entry Server Report
csm	Cache Services Report
doc	Document Services Report

## Predefined Soft Reports

Type	Report Name
inx	Index Services Report
nch	Network Clearinghouse Report
pri	Print Services Report
psm	Printer Services Report
prt	Print Activity Report
pf	Image Prefetch Activity Report
cpr	Client Page Request Report
pa	Performance Measurements
fs	File Services Report
sq	SQL Services Report
sec	Security Services Report
fb	Fast Batch Throughput Report
wfl	WorkFlo Activity Report
aix_net	AIX Net Activity
aix_vmm	AIX Virtual Memory Activity
sol_vmm	Solaris Virtual Memory Activity
hpux_vmm	HPUX Virtual Memory Activity
oli_vmm	Olivetti Virtual Memory Activity
vw	Visual WorkFlo Activity
transdb_io	Transient DB I/O Tuning Report
transdb_trans	Transient DB Transaction Tuning Report
permdb_io	Permanent DB I/O Tuning Report
permdb_trans	Permanent DB Transaction Tuning Report

---

### Predefined Soft Reports

Type	Report Name
secdb_io	Security DB I/O Tuning Report
secdb_trans	Security DB Transaction Tuning Report

### User-Defined Soft Reports

You can also define your own soft reports. To create a user-defined report, you must create a report description file containing statements that build a table of data values. Once built, the perf\_report tool uses this table to generate the type of report you need in a format that you have prescribed. You specify your report description file with the `-fr` option.

A single report supports a maximum of 128 columns, subject to a limit on the total size of all expressions. Each operand, operator, and function name count as one element. In addition, each function, column or compound variable counts as one element. It supports a maximum of 2048 elements.

Each column of the table is defined by an arithmetic expression and a title string. In addition, a column can be defined with optional print format information and information about the type of summary desired for the column.

The table rows correspond to a time frame and the data that occurred in the time frame. The report displays records based on the perf\_report options you select, such as a particular service, date, time, etc. (See the descriptions in the Options section above).

For example, you can specify hourly intervals (using the `-ph` option) to display rows of data in hourly intervals, even if the data records were collected every 15 minutes.

To generate soft reports, `perf_report` reads two files: the directory of Event Log Abstract (ELA) event categories and a reports file. The default locations of each are:

ELA event categories default directory:

<code>/fnsw/lib/perf/ela/elacodes</code>	for UNIX platforms
<code>&lt;drive&gt;:\fnsw\lib\perf\ela\elacodes</code>	for Windows platforms

Reports default file path:

<code>/fnsw/lib/perf/reports</code>	for UNIX platforms
<code>&lt;drive&gt;:\fnsw\lib\perf\reports</code>	for Windows platforms

The ELA directory contains the symbolic names for ELA event categories and the minor events associated with each category. System (i.e. nonELA) variable names are also organized into categories in this directory. The definitions in the report file consist, in part, of these ELA names and other system variable names.

Use the following command to display a list of all ELA and nonELA event categories:

**`perf_report -s`**

Soft report descriptions in the reports file have two constructs: the single-statement "variable" construct and the multi-statement "report" construct.



The variable construct defines a compound variable. The perf\_report tool supports a maximum of 128 user-defined variables, subject to the limit on the total size of all expressions as described above. Arithmetic expressions are supported in a C-like syntax. The basic operator set consists of add (+), subtract (-), divide (/) and multiply (\*). Operands are system variables, integer constants or floating point constants. Variables can be basic or compound (user-defined). (See the description of [“Basic Variables” on page 981](#) and [“Compound Variables” on page 983](#).)

Report descriptions can also support a set of functions. (Refer to [“Functions” on page 985](#).)

The following example taken from a reports file shows the definition of a variable called "cputime":

```
variable cputime = (sys.idle + sys.wait + sys.kernel + sys.user)/100.0
```

The multi-statement report construct contains other statements but can also include "variable" statements. The construct carries a statement over to a new line using the backslash (\) symbol. The multi-statement report construct terminates with an "end" statement.

## Report Description File

The following example shows the format definition for the CSS Summary Report, taken from a report description file.

```
report css "CSS Summary Report"
  column idle title "%I" format "d3" average
  column kernel title "%K" format "d2" average
  column user title "%U" format "d2" average
  column wait title "%W" format "d2" average
  group "CPU"
  column disk.ops.rate title "OPS/SEC" format "d8" average
  group "DISK"
  column sec.CurrentSLUs title "CURRENT" format "d8" average
  column sec.PeakSLUs title "PEAK" format "d8"
  column sec.MaximumSLUs title "MAX ALLOWED" format "d8"
  column sec.SLURejects title "REJECTS" format "d8" average sum
  column sec.SLUSoftLimit title "SOFT CUTOFF" format "d8"
  column sec.SLUSoftViolations title "SOFT VIOLATIONS" format "d8" average sum
  group "SLU"
end
```

The example above describes a CSS Summary Report that consists of three groups of columns. Each column has its own title and format. Each group name appears immediately after the columns describing the information included in the specified group.

The CSS report format defines the format for three groups of information:

- CPU describes the columns that display the CPU-related performance statistics. The column titles present the column formats that identify the average percentage of time the CPU spends on activities during the data collection time, including idle (%I), kernel (%K), user (%U), and wait (%W) time.
- DISK describes the column containing the disk-related information. This column has a title of OPS/SEC which reflects the operations per second.
- SLU consists of six columns with titles of CURRENT, PEAK, MAX ALLOWED, REJECTS, SOFT CUTOFF, and SOFT VIOLATIONS. These define the statistics collected regarding the SLAC key usage. (For details, see [\*\*“Customer Service Report” on page 1005.\*\*](#))

The end statement terminates this report construct.

---

The following sample is a complete report definition for a Service Activity report. The report definition is from the reports file in /fnsw/lib/perf.

---

```
# perf_report report descriptions #
variable cputime = (sys.idle + sys.wait + sys.kernel + sys.user)/100.0
variable idle = sys.idle/cputime
variable wait = sys.wait/cputime
variable user = sys.user/cputime
variable kernel = sys.kernel/cputime
report sv "Service activity"
    column idle title "%I" format "d3" average
    column kernel title "%K" format "d2" average
    column user title "%U" format "d2" average
    column wait title "%W" format "d2" average
    group "CPU"
    column INX:$ title "COUNT" sum average
    column INX:$.dura/(1000*INX:$) title "DURA" format "f4.1" average
    group "INX"
    column WQS:$ title "COUNT" sum average
    column WQS:$.dura/(1000*WQS:$) title "DURA" format "f4.1" average
    group "WQS"
    column SQI:$ title "COUNT" sum average
    column SQI:$.dura/(1000*SQI:$) title "DURA" format "f4.1" average
    group "SQI"
average
    group "BES"
    column CSM:$ title "COUNT" sum average
    column CSM:$.dura/(1000*CSM:$) title "DURA" format "f4.1" average
    group "CSM"
    column DOC:$ title "COUNT" sum average
    column DOC:$.dura/(1000*DOC:$) title "DURA" format "f4.1" average
    group "DOC"
end
```

---

## Basic Variables

To build your report file, you use a set of basic variables to define the information you want retrieved from the performance log. Basic variables are either ELA data variables, nonELA variables, or special variables.

Both ELA and nonELA data variable names have two main parts, a category name and an item name. An ELA category name is separated from an item name by a colon (:). A period (.) separates nonELA category names from item names. In addition, both ELA and nonELA variable names can have one or two option suffixes. (See [“Suffixes” on page 984.](#)) A period (".") separates the name from its suffix(es). Some nonELA variables also use optional variable-dependent parameters.

---

**Tip** To see a list of variable-dependent parameters, run `perf_report -s` and locate the parameters under the nonELA category descriptions in the output listing.

---

Special variables have only one part and can be program constants, configuration variables, or time values. (See [“Special Variables” on page 983](#) for more information.)

The following are names and short descriptions of ELA event names of remote procedures for which you can collect data on both server and client systems (client procedures are identified with a suffix of 'c'):

ELA Event Names	Description
<b>BES/BESc</b>	Batch Entry Service RPCs
<b>CSM/CSMc</b>	Cache Services Manager RPCs

ELA Event Names, Continued	Description, Continued
<b>DOC/DOCc</b>	Document Services RPCs
<b>INX/INXc</b>	Index Services RPCs
<b>PSM/PSMc</b>	Print Services Manager RPCs
<b>PQM/PQMc</b>	Printer Service RPCs
<b>SEC/SECc</b>	Security Service RPCs
<b>WQM/WQMc</b>	WorkFlo Queue Manager RPCs
<b>WQS/WQSc</b>	WorkFlo Services RPCs

NonELA variable categories are listed below with a brief description:

NonELA Variable Categories	Description
disk	Magnetic disk drive statistics
ds	Local Document Services statistics, both front and back end
ic	Miscellaneous variables and configuration parameters (see <a href="#">“Special Variables” on page 983</a> )
mkf	Multikeyed database statistics
net	Network statistics
ppm	Server Process Manager statistics
sec	Security Service SLU statistics
sys	Corresponds to the kernel sysinfo structure
vmm	Virtual memory statistics
wfl	WorkFlo use statistics

## Compound Variables

A "variable" statement followed by a variable name and any legal arithmetic expression(s) defines a compound variable. You can define a compound variable anywhere in a report description file. No colon (:) or period (.) can appear in a compound variable name.

The format of a compound variable description is

```
variable <variable_name> = <exp>
```

The following is an example of a compound variable. The variable named "pkts" is defined to represent the SPP packet rate per hour.

```
variable pkts = (net.SPPpktsin.rate+net.SPPpktsout.rate) * 3600
```

## Special Variables

Special variable names have only one part and, in most cases, cannot be used with suffixes. Special variables can be time variables, configuration-related variables, or program constants. Special variables include all variables of the nonELA category "ic." Use the `-s` option of the `perf_report` command for a complete list of these variables, as shown below.

```
corona(root)/> perf_report -sic
Data symbols for category: ic (Misc. and Configuration variables)
*****

stanum      t          ndisks     nqueues    nevents     nmkfdbs
sv_root     sv_boot   sv_file    sv_index   sv_doc       sv_batch
sv_cache    sv_print  sv_osar    sv_wfl     sv_dtars    sv_nch
sv_sort
```

Some useful special variables are listed below:

Special Variable	Description
t	Time interval in seconds between the two data record currently in use. Division by this variable is equivalent to using the "rate" suffix. Use t instead of rate whenever there is an expression with a linear combination of rates.
stanum	Server number on which the record was created
ndisks	Number of magnetic drives in the configuration
nqueues	Number of WorkFlo queues created (if any)
nevents	Number of ELA events enabled for the station
nmkfdbs	Number of MKF databases on the station

## Suffixes

Suffixes are optional additions to names, which you can use whenever they are semantically valid. Both ELA and nonELA variable names can have one or two option suffixes. A period (.) separates a name from its suffixes. There are two types of suffixes—data type and time function. Semantically incorrect data type suffixes used with nonELA variables are reported as parsing errors.

The following are the data type suffixes:

Data Type Suffix	Description
count	Usage count for the item (default data type suffix)
dura	Usage duration for the item
size	Byte count for the data item
dumax	Maximum duration since software restart



Data Type Suffix, Continued	Description, Continued
dumin	Minimum duration since software restart
szmax	Maximum size since software restart
szmin	Minimum size since software restart
duimax	Maximum duration in the sample interval
duimin	Minimum duration in the sample interval
szimax	Maximum size in the sample interval
szimin	Minimum size in the sample interval

The following are the time function suffixes:

Time Function Suffix	Description
count	Usage count for the item (default data type suffix)
cumu	Absolute (cumulative) value of the data item
delta	Incremental change (delta) between two data records (delta is the default time function suffix)
rate	Rate of change per second between two records

**Note** The use of the rate suffix is equivalent to explicitly dividing the data item by the **t** variable. See [“Special Variables” on page 983](#).

## Functions

General functions are supported in expressions that you use in your reports description file. Function arguments are general expressions, and are separated by commas. The perf\_report tool supports a maximum of 128 function arguments, subject to the limit on the total size of

all expressions (see the discussion on expression limits under **“Soft Reports” on page 972**). The supported functions are listed below:

Function	Description
sum(x1,x2,x3,...)	Computes the sum of the arguments represented by x
max(x1,x2,x3,...)	Computes the maximum of the arguments represented by x
min(x1,x2,x3,...)	Computes the minimum of the arguments represented by x
log(x)	Computes the natural logarithm of x
log10(x)	Computes the logarithm of x to base 10
exp(x)	Computes the exponent of x
pow(x,y)	Computes x to the y power
sqrt(x)	Computes the square root of x
int(x)	Truncates data by the value of x
round(x)	Rounds off the value of x to the nearest integer

The following examples show the use of functions. The first example computes the sum of the values represented by the rate of network SPP packets received and the rate of network SPP packets sent:

**sum(net.SPPpktsin/t,net.SPPpktsout/t)**

The example below computes the maximum of the two values represented by sys.user and sys.kernel:

**max(sys.user,sys.kernel)**

---

## Report Construct

Once you have defined variables in the report description file, you can define the report construct itself. The report construct consists of one or more column statements, and optional group and variable statements.

A "column" statement binds an expression and title string to a column in the output report. You can use a group statement to group columns under an optional subtitle. A variable statement within the report construct defines compound variables that have scope only within the given report from the next statement following the variable to the end of the reports file.

The format of the report construct is as follows:

```
report <report_name> <title_string>
column <exp> title <title_string> [format
<fmt_string>] [sum] [average]
[group] [<title_string>]
[column]
[group]
...
end
```

where

**title\_string** is a text string used as the title for a report or for columns or groups. A report title is limited to one line. Column or group titles are limited to 3 lines. Separate each line of a multiple-line title with the backslash (\) character. The formatted string is centered over the report.

In the case of columns, the string is centered in the column width. Column width is computed as the maximum of the column title string width and the data format width. In the case of a group, the string is centered over the group's width. Group width is computed as the sum of the respective column widths. Group titles extending beyond this width are truncated.

**fmt\_string** is an optional three-part string that defines a left margin, right margin, and a data format for the column. Margins default to a value of zero. The three-part string is defined as follows:

```
[l#] [<data_fmt>] [r#]  
<data_fmt>=<int_fmt> | <float_fmt>  
<int_fmt>=#.#]d  
<float_fmt>=#.#f
```

The components of the format string are defined as follows:

- **l#**  
Left margin position, where # is number of positions (default is 0).
- **r#**  
Right margin position, where # is number of positions (default is 0).
- **data\_fmt**  
Minimum width field expressed as an integer (<int\_fmt>) or floating point number (<float\_fmt>). The minimum width field can be followed by an optional period and second width field. The width field(s) are followed by a data type symbol for integer (d) or floating point (f). Data values are printed right justified.
- **int\_fmt**  
Minimum width field expressed as an integer. The second width

field, if used, specifies the maximum data field width. (If not specified, the default is 5d.)

- **float\_fmt**  
Minimum width field expressed as a floating point number. The second width field, if used, specifies the maximum number of places print to the right of the decimal point. (If not specified, the default is 5.1f.)

**sum** generates optional summaries per column, printing the column total at the bottom of the report

**average** generates optional summaries per column, printing the average per row (record) at the bottom of the report.

**group** creates a group of columns starting from the first column or from the first column after the last "group" statement. It can provide an optional title string to print centered over the group of columns. If used, the optional title string must be three lines or less. Title lines are truncated to fit over the total width of the column. Total width of the column is determined by the width of each column in the group. The width of each column is the maximum of the width of the column title and the maximum data format width, truncated to a maximum of 16 characters.

## Sample Output

To create the following sample report, perf\_mon was run to collect performance data for a specified interval. The captured data was written to

a log file /fnsw/tmp/log12096. The commands used to perform these steps follow:

```
corona(root)/> perf_mon
perf_mon> sample /fnsw/tmp/log12096 5 10
perf_mon> quit
```

The next step was to create reports using the data collected and stored in the log. This sample uses the Full Report option (`-rful`) and the default report description file. This combination creates a large amount of data, sent to the standard output device (usually the terminal from which you invoked `perf_report`). A better way to view the data is to redirect the output reports to a file and use the `less` command to scroll through the reports. The set of commands to perform those steps follow:

```
corona> perf_report /fnsw/tmp/log12096 -rful > /fnsw/
tmp/testperf
corona(root)/> ls /fnsw/tmp/t*
testperf
corona(root)/> less /fnsw/tmp/testperf
```

The following pages show a partial listing of the reports. Since this example captured data every 5 seconds for 10 intervals, ten reports are in the file. This example shows only one of those reports.

---

New\_Day (00:00-00:00) (1) HQ-HAMLET

Base file : zz

Date-time : Wed Aug 31 11:32:35 2005

Date-time : Wed Aug 31 11:33:35 2005

Services: INDEX DOC BATCH CACHE PRI OSAR  
WorkFlo NCH SQL RmtFile Security

NT Server hardware configuration:

CPUs	Type	PageSize	Disks
4	Pentium	4096	0

CPU Utilization:

priority	% total	% n-idl	ticks	tot-cpu-time
idle	99.367	-----	5962	59.620
kernel	0.450	71.053	27	0.270
user	0.183	28.947	11	0.110
wait	0.000	0.000	0	0.000
	100.000	100.000	6000	60.000

NT System Activity

PSWITCHES	SYSCALLS	INTERRUPTS	PQUELEN
36134	652999	21955	0

NT File System Activity

READS	READ KB	WRITES	WRITE KB
1412	20581	330	5013

SEC SLU Statistics:

Current Number of SLUs = 1

SLU High Water Mark = 1  
 Maximum Allowed SLUs = 99999  
 Logon Rejections = 0  
  
 Soft Limit of SLUs = 99999  
 Soft Limit Violations = 0

## Counts:

PPM Process Starts = 0  
 PPM No Process = 0

## Document Services -- Incremental Report for Wed Aug 31 11:33:35 2005

	Docs	Pages	Cache	Disk	Errors
	-----	-----	-----	-----	-----
Committals:	0	0			
Fast Batch:	0	0			
Migrations (H):	204	204	204	0	0
Migrations (M):	0	0	0	0	0
Migrations (L):	0	0	0	0	0
Imports:	0				0

## Summary of DOC Calls

-----  
 Prefetch Calls: 0  
 Migrate Calls: 204  
 Async Migrate Calls: 0  
 Fast Batch Committals: 0  
 Client Page Requests: 204  
 Client Page Cache Hits: 204  
 Client Cache Hit Ratio %: 100.000

PageRD	PageRD	PageRD	PageWR	Reads	Writes	Reads	Writes	Loads
(H)	(M)	(L)		(KB)	(KB)			
-----	-----	-----	-----	-----	-----	-----	-----	-----
=====	=====	=====	=====	=====	=====	=====	=====	=====

SUM:



Queue Removals: 0 RD(H), 0 RD(M), 0 RD(L), 0 WR, 0 BG

Arm Moves Op Loads Op Unloads  
-----

Workspace : General WorkfloQueue : General

Operation Name Count Duration OraCount OraDurat LockCount LockDurat  
-----

MKF Base Data File = C:\FNSW\_LOC\SD\NCH\_DB0

160 finds (52 single op)  
0 inserts (0 single op)  
0 updates 0 deletes 0 set\_pos

27 begin\_trans 21 end\_trans 6 abort\_trans  
250 short\_start 244 short\_compl 6 short\_abort  
0 long\_start 0 long\_compl 0 long\_abort  
0 short->long 0 successful 0 deadlocks

341 readblocks 341 cachehits (100.00% hit ratio, 32 buffers)

	Reads	Blks	Writes	Blks
	=====	=====	=====	=====
AIJ:	0	0.00	0	0.00
BIJ:	0	0.00	0	0.00
DB:	0	0.00	0	0.00

0 shutdowns 3 open 0 close  
15 open\_cursor 0 close\_cursor  
84 get\_item\_num 72 bind  
0 open\_desc 0 close\_desc  
0 get\_param 0 set\_param  
0 get\_file 0 set\_file  
0 get\_record 0 set\_record  
0 get\_item 0 set\_item  
0 get\_aij\_bsn 0 update\_avsp  
0 trans\_state 0 set\_abortmode

```

0 getbuffer      341 divests      314 divestcompletely
160 search:      160 firsttry      0 forupdate
187 searchtable  0 insertkey        0 deletekey
0 merge calls:   0 table merges     0 table splits

```

MKF Base Data File = C:\FNSW\DEV\1\PERMANENT\_DB0

```

204 finds (204 single op)
0 inserts (0 single op)
0 updates      0 deletes      0 set_pos

0 begin_trans  0 end_trans     0 abort_trans
226 short_start 226 short_compl 0 short_abort
0 long_start   0 long_compl    0 long_abort
0 short->long  0 successful    0 deadlocks

408 readblocks 396 cachehits ( 97.06% hit ratio, 312 buffers)

```

	Reads	Blks	Writes	Blks
	=====	=====	=====	=====
AIJ:	0	0.00	0	0.00
BIJ:	0	0.00	0	0.00
DB:	12	1.00	0	0.00

```

0 shutdowns      1 open      0 close
1 open_cursor    0 close_cursor
5 get_item_num   16 bind
0 open_desc      0 close_desc
0 get_param      0 set_param
0 get_file       0 set_file
0 get_record     0 set_record
0 get_item       0 set_item
0 get_aij_bsn    0 update_avsp
0 trans_state    0 set_abortmode

```

```

12 getbuffer      408 divests      408 divestcompletely
204 search:      204 firsttry      0 forupdate
204 searchtable  0 insertkey        0 deletekey
0 merge calls:   0 table merges     0 table splits

```

MKF Base Data File = C:\FNSW\DEV\1\TRANSIENT\_DB0

```

616 finds (1 single op)
  0 inserts (0 single op)
204 updates      0 deletes      0 set_pos

411 begin_trans   409 end_trans     2 abort_trans
498 short_start  496 short_compl   2 short_abort
  0 long_start    0 long_compl      0 long_abort
  0 short->long    0 successful      0 deadlocks

1789 readblocks  1781 cachehits ( 99.55% hit ratio, 312 buffers)

```

	Reads	Blks	Writes	Blks
	=====	=====	=====	=====
AIJ:	204	2.87	204	2.87
BIJ:	0	0.00	0	0.00
DB:	27	1.00	19	1.00

```

  0 shutdowns      2 open          0 close
  8 open_cursor    0 close_cursor
14 get_item_num    64 bind
  0 open_desc      0 close_desc
  0 get_param       0 set_param
  0 get_file        0 set_file
  0 get_record      0 set_record
  0 get_item        0 set_item
  0 get_aij_bsn     0 update_avsp
204 trans_state    0 set_abortmode

390 getbuffer     1611 divests    1407 divestcompletely
972 search:       616 firsttry    356 forupdate
972 searchtable   178 insertkey   178 deletekey
  0 merge calls:   0 table merges  0 table splits

```

MKF Base Data File = C:\FNSW\DEV\1\SEC\_DB0

```

39 finds (36 single op)
  0 inserts (0 single op)
  3 updates      0 deletes      0 set_pos

  2 begin_trans    2 end_trans     0 abort_trans

```

```

416 short_start      416 short_compl      0 short_abort
  0 long_start       0 long_compl         0 long_abort
  0 short->long       0 successful          0 deadlocks

89 readblocks       85 cachehits ( 95.51% hit ratio, 150 buffers)

```

```

      Reads   Blks      Writes   Blks
=====  =====  =====  =====
AIJ:         2   2.50           2   2.50
BIJ:         0   0.00           0   0.00
DB:          4   1.00           0   0.00

```

```

  0 shutdowns        3 open          0 close
33 open_cursor       0 close_cursor
51 get_item_num      294 bind
  0 open_desc         0 close_desc
  0 get_param          0 set_param
  0 get_file           0 set_file
  0 get_record         0 set_record
  0 get_item           0 set_item
  0 get_aij_bsn        0 update_avsp
  0 trans_state        0 set_abortmode

  7 getbuffer         89 divests        80 divestcompletely
39 search:           39 firsttry        0 forupdate
48 searchtable        0 insertkey        0 deletekey
  0 merge calls:      0 table merges    0 table splits

```

## Event Data for Category CSM (0) (4 of 40)

MINOR CODE	COUNT	AVG DUR	IVAL	MAX	AVG SIZE	TOT DUR	KB/SEC
( 6) OpenObject	204	0.000	0.000		0	0.00	
( 7) CloseObject	203	0.000	0.016		0	0.03	
(15) ReadObject	408	0.004	0.031		37303	1.50	10132.91
(30) Logon1	1	0.015	0.015		0	0.01	

## Event Data for Category DOC (1) (2 of 50)

MINOR CODE	COUNT	AVG DUR	IVAL	MAX	AVG SIZE	TOT DUR	KB/SEC
(26) MigrateWSSNFromO	204	0.002	0.063		0	0.33	
(30) Logon1	1	0.032	0.032		0	0.03	

Event Data for Category INX (2) (0 of 85)

MINOR CODE	COUNT	AVG DUR	IVAL	MAX	AVG SIZE	TOT DUR	KB/SEC
------------	-------	---------	------	-----	----------	---------	--------

Event Data for Category PRI (3) (0 of 25)

MINOR CODE	COUNT	AVG DUR	IVAL	MAX	AVG SIZE	TOT DUR	KB/SEC
------------	-------	---------	------	-----	----------	---------	--------

Event Data for Category BES (4) (0 of 75)

MINOR CODE	COUNT	AVG DUR	IVAL	MAX	AVG SIZE	TOT DUR	KB/SEC
------------	-------	---------	------	-----	----------	---------	--------

Event Data for Category SEC (20) (4 of 60)

MINOR CODE	COUNT	AVG DUR	IVAL	MAX	AVG SIZE	TOT DUR	KB/SEC
(19) FindSystemDefaul	2	0.000	0.000		0	0.00	
(52) FindObject2	2	0.000	0.000		0	0.00	
(53) FindEndpt2	1	0.000	0.000		0	0.00	
(55) Logon2	1	0.063	0.063		0	0.06	

Event Data for Category SQI (23) (0 of 20)

MINOR CODE	COUNT	AVG DUR	IVAL	MAX	AVG SIZE	TOT DUR	KB/SEC
------------	-------	---------	------	-----	----------	---------	--------

Event Data for Category FIL (24) (0 of 12)

---

```

      MINOR CODE          COUNT  AVG DUR  IVAL MAX  AVG SIZE  TOT DUR  KB/SEC
=====

```

Event Data for Category WQS (26) (0 of 45)

```

      MINOR CODE          COUNT  AVG DUR  IVAL MAX  AVG SIZE  TOT DUR  KB/SEC
=====

```

Event Data for Category CPT (29) (1 of 8)

```

      MINOR CODE          COUNT  AVG DUR  IVAL MAX  AVG SIZE  TOT DUR  KB/SEC
=====

```

Event Data for Category NCH (61) (2 of 127)

```

      MINOR CODE          COUNT  AVG DUR  IVAL MAX  AVG SIZE  TOT DUR  KB/SEC
=====
( 7)      ListObjects          6    0.000    0.000        36    0.00
(38) RetrieveTimedIte        15    0.000    0.000         0    0.00

```

Event Data for Category PSMc (74) (0 of 10)

```

      MINOR CODE          COUNT  AVG DUR  IVAL MAX  AVG SIZE  TOT DUR  KB/SEC
=====

```

end report

---

## Event Data Definitions

As an example, Event Data for Category NCH (61) (2 of 127) bold in the sample above - This line represents the name of the Major event category for Network Clearinghouse, which is major event #61 in the

elacodes file. It has been assigned 127 minor events (or RPCs) and of those 127, 2 have been active since software restart (possibly occurring before the report was started).

**Event** - The minor event code from the elacodes file that was active in this reporting interval.

**MINOR CODE** - The minor event name from the elacodes file that was active in this reporting interval.

**COUNT** - The number of this specific event that happened during the reporting interval.

**AVG DUR** - The average duration of this event during the reporting interval.

**IVAL MAX** - The maximum duration of this event that occurred in this reporting interval. This is only accurate if the reporting interval is the same as the sample interval. If you run a report with the "-rful" option and no reporting interval change options, the sample interval equals the reporting interval. However, if you run **perf\_report -rful -m1 -n99**, which writes one report interval for all 99 samples, the IVAL MAX will be the maximum of the last sample interval.

**AVG SIZE** - The average size of this event (if applicable) during the reporting interval.

**TOT DUR** - The total duration across all occurrences of this event that happened during the reporting interval.

**KB/SEC** - The number of Kilobytes per second (if applicable) related to this event that happened during the reporting interval.

## Sample Reports

The following reports provide you with a sample of output for each report type. These particular samples were generated on an AIX Combined server. Reports are similar across supported platforms.

### Scout Report

To use the perf\_report output directly with Scout, use the following syntax:

```
perf_report -rscout -ph -csv > scout.txt
```

The **-csv** option is a requirement.

This example illustrates how to use a combination of options to get the report into a format that can be imported into Scout.

The following example shows the report generated when running perf\_report to generate the scout report. This report provides the baseline demand for a system being upgraded using the Scout sales tool. Using the **-ph** option collects the data hourly in chronological order, as the Scout product expects to see the data. The **-tsv** option formats the information in columns separated by tabs, making it easy to import into an Excel Spreadsheet.

---

```
elcapitan(fnsw)/usr/fnsw> perf_report -rscout -ph -tsv
Log file defaulting to /fnsw/local/sd/1/perflog
```

```
Scout Overlay Sheet Data for Wednesday, September 15, 1999 (00:00-00:00) (1) eln
```

```
Daily CPU, Network, Osar, and Optical Drive Demands
```

```
-----
AS CSM  Appl    BES      DOC      INX      NCH      Osar CSM      PRI
```



00:00	0.00	0.00	*	*	*	*	*	*
01:00	0.00	0.00	*	*	*	*	*	*
02:00	0.00	0.00	*	*	*	*	*	*
03:00	0.00	0.00	*	*	*	*	*	*
04:00	0.00	0.00	*	*	*	*	*	*
05:00	0.00	0.00	*	*	*	*	*	*
06:00	0.00	0.00	*	*	*	*	*	*
07:00	0.00	0.00	*	*	*	*	*	*
08:00	0.00	0.00	*	*	*	*	*	*
09:00	0.00	0.00	*	*	*	*	*	*
10:00	0.00	0.00	*	*	*	*	*	*
11:00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00
12:00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00
13:00	0.00	0.00	*	*	*	*	*	*
14:00	0.00	0.00	*	*	*	*	*	*
15:00	0.00	0.00	*	*	*	*	*	*
16:00	0.00	0.00	*	*	*	*	*	*
17:00	0.00	0.00	-	-	-	-	-	-
18:00	0.00	0.00	-	-	-	-	-	-
19:00	0.00	0.00	-	-	-	-	-	-
20:00	0.00	0.00	-	-	-	-	-	-
21:00	0.00	0.00	-	-	-	-	-	-
22:00	0.00	0.00	-	-	-	-	-	-
23:00	0.00	0.00	-	-	-	-	-	-
SUM:								
AVG:			0.03	0.00	0.00	0.00	0.00	0.00

+ Data item(s) for this column are not relevant to the station type

- Data item(s) for this column were not logged

## Batch Services Report

The batch services (batch entry) report (cmb1.bs.Aug31.txt) displays information about batch activity.

```
Batch Entry Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan
```

	CPU				CreateBatc		Open Batch		CommitBatc		CreatImage		WriteImage		Read Image		Create Doc		Update Doc	
%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	
10:00	72	1	3	24	25	0.0	26	0.0	2	0.0	2284	0.0	2291	0.2	14	0.0	2426	0.0	16	0.0
09:45	95	0	1	4	11	0.0	0	*	0	*	249	0.0	248	0.2	0	*	0	*	0	*
09:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
09:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
09:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:15	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:30	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
04:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
02:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
01:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
SUM:					36		26		2		2533		2539		14		2426		16	
AVG:	98	0	0	2	2	0.0	1	0.0	0	0.0	127	0.0	127	0.2	1	0.0	121	0.0	1	0.0

**Note** You can generate this report using the perf\_report -rbs command.

## Cache Services Report

The cache services report (cmb1.csm.Aug31.txt) displays information about the activity of objects in cache.

Cache Services Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

CPU            Create Obj   Open Obj   Close Obj   Delete Obj   Find Objs   Read Obj   Write Obj   Copy Obj   Move Obj  
CSM Hits

-----  
-----  
%I   %K   %U   %W   COUNT   DURA   COUNT   DURA   COUNT   DURA   COUNT   DURA   COUNT   DURA   COUNT   DURA   COUNT   DURA   COUNT   DURA   COUNT  
DURA   RATIO

```

--- -----
10:00 72 1 3 24 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
* *
09:45 95 0 1 4 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
* *
09:30 100 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
09:15 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
09:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
08:45 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
08:30 100 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
08:15 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
08:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
07:45 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
07:30 100 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
07:15 100 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
07:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *

```

---

```

06:45 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * *
06:30 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * *
06:15 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * *
06:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * *
04:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * *
02:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * *
01:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * *
-----
-----
SUM:          0      0      0      0      0      0      0      0
AVG:  98 0 0 2 0      0      0      0      0      0      0      0

```

---

**Note** You can generate this report using the perf\_report -rcsm command.

## Customer Service Report

As shown in the following example (cmb.css.Apr27.txt), the css report identifies the collection times at the far left. It displays three groups of information collected exactly at the time specified. The first two groups summarize CPU and disk performance. The last group summarizes simultaneous logged on users (SLUs). (See [“CSS Report SLU Column Descriptions” on page 1007.](#))

CSS Summary Report for Tuesday, April 27, 1999 (00:00-00:00) (1) redondo

	CPU				DISK			SLU			
	%I	%K	%U	%W	OPS/SEC	CURRENT	PEAK	MAX ALLOWED	REJECTS	SOFT CUTOFF	SOFT VIOLATIONS
16:30	19	7	18	56	200	2	12	99999	0	99999	0
16:15	20	6	15	59	198	3	12	99999	0	99999	0
16:00	21	6	15	58	198	3	12	99999	0	99999	0
15:45	19	7	16	58	197	3	12	99999	0	99999	0
15:30	20	7	16	57	95	4	12	99999	0	99999	0
15:15	45	14	23	18	44	3	12	99999	0	99999	0
15:00	38	7	47	8	18	2	12	99999	0	99999	0
14:45	26	9	28	36	123	3	12	99999	0	99999	0
14:30	17	9	22	53	197	3	12	99999	0	99999	0
14:15	18	7	18	57	198	1	12	99999	0	99999	0
14:00	20	6	16	58	198	1	12	99999	0	99999	0
13:45	20	6	16	58	196	1	12	99999	0	99999	0
13:30	20	7	15	58	188	1	12	99999	0	99999	0
13:15	21	6	16	58	196	1	12	99999	0	99999	0
13:00	47	3	44	6	16	1	12	99999	0	99999	0
12:45	62	5	27	5	11	1	12	99999	0	99999	0
12:30	22	5	16	57	201	1	12	99999	0	99999	0

---

12:15	21	5	16	58	204	1	12	99999	0	99999	0
12:00	21	5	16	58	204	1	12	99999	0	99999	0
11:45	53	5	10	32	108	1	12	99999	0	99999	0
11:30	58	10	10	21	65	1	12	99999	0	99999	0
11:15	19	7	15	59	203	1	12	99999	0	99999	0
11:00	20	6	15	60	202	1	12	99999	0	99999	0
10:45	18	7	15	60	203	1	12	99999	0	99999	0
10:30	31	5	26	38	127	1	12	99999	0	99999	0
10:15	46	5	41	8	18	1	12	99999	0	99999	0
10:00	23	7	17	53	175	1	12	99999	0	99999	0
09:45	19	6	15	60	203	1	12	99999	0	99999	0
09:30	16	11	17	56	215	1	12	99999	0	99999	0
09:15	20	6	15	59	201	1	12	99999	0	99999	0
09:00	20	6	15	59	201	1	12	99999	0	99999	0
08:45	20	6	15	60	201	1	12	99999	0	99999	0
08:30	25	5	19	51	173	1	12	99999	0	99999	0
08:15	50	4	43	4	6	1	12	99999	0	99999	0
08:00	31	6	22	41	131	1	12	99999	0	99999	0
07:45	20	5	15	60	203	1	12	99999	0	99999	0
07:30	20	5	15	60	203	1	12	99999	0	99999	0
07:15	20	5	14	60	201	1	12	99999	0	99999	0
07:00	20	5	14	60	201	1	12	99999	0	99999	0
06:45	21	5	14	60	200	1	12	99999	0	99999	0
06:30	21	5	14	60	200	1	12	99999	0	99999	0
06:15	46	4	41	9	26	1	12	99999	0	99999	0
06:00	25	5	19	51	171	1	12	99999	0	99999	0
04:00	25	5	19	51	170	1	12	99999	0	99999	0
02:00	28	5	21	46	150	1	12	99999	0	99999	0
-----											
SUM:									0		0
AVG:	27	6	20	47	159	1			0		0

---

---

**Note** You can generate this report using the `perf_report -rcss` command.

---

#### CSS Report SLU Column Descriptions

Heading	Description
CURRENT	Displays the number of users logged onto the Image Services system at the end of the report sample interval
PEAK	Displays the highest number of users logged on since the Image Services system was last recycled. This parameter is called the "maximum usage" in the report displayed with the SEC_tool license command.
MAX ALLOWED	<p>License management policy enables users to obtain Universal SLAC keys that set this value to 99,999. Although other system constraints would prevent this many users from logging onto an Image Services server at the same time, this number indicates that up to 99,999 users can log on simultaneously.</p> <p>Customers are still bound not to exceed the license quantity purchased; however, the Universal SLAC keys do not lockout connections. Comparison of report results with customer license configurations is a sales team activity.</p> <p>This field is called the "hard SLU limit" in the report displayed with the SEC_tool license command.</p>
REJECTS	This field does not apply to the current software release.

## CSS Report SLU Column Descriptions

Heading	Description
SOFT CUTOFF	<p>License management policy enables users to obtain Universal SLAC keys that set this value to 99,999. Although other system constraints would prevent this many users from logging onto an Image Services server at the same time, this number indicates that up to 99,999 users can log on simultaneously.</p> <p>Customers are still bound not to exceed the license quantity purchased; however, the Universal SLAC keys do not lockout connections. Comparison of report results with customer license configurations is a sales team activity.</p> <p>This field is called the "soft SLU limit" in the report displayed with the SEC_tool license command.</p>
SOFT VIOLATIONS	This field does not apply to the current software release.



## Document Services Report

The document services report (cmb1.doc.Aug31.txt) displays detailed information about document activity.

Document Services Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

Batch	CPU	Commit	Doc	MigToOD	MigFromOD	Prefetch	Creat	Anno	Get	Annot	Batch	Crt	Batch	Wrt						
Cls																				
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----						
%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	
COUNT	DURA																			
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----						
10:00	72	1	3	24	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
09:45	95	0	1	4	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
09:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
09:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
09:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
08:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
08:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
08:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
08:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
07:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
07:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
07:15	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			
07:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																			

---

```

06:45 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
06:30 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
06:15 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
06:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
04:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
02:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
01:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
-----
----
SUM:          0          0          0          0          0          0          0          0
AVG:  98 0 0 2 0          0          0          0          0          0          0          0          0

```

---

**Note** You can generate this report using the `perf_report -rdoc` command.

## Document Services Summary Report

The document services summary report (cmb1.ds.Aug31.txt) displays summary information about document activity.

```
Document Services Summary for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan
```

	CPU				DS Frontend							DS Backend					
	%I	%K	%U	%W	BESwrs	BESrds	CMTS	CSMrds	MIGSH	MIGSM	MIGSL	IMPTS	RDS_H	RDS_M	RDS_L	WRTS	LOADS
NFAIL																	
10:00	72	1	3	24	2291	14	6	0	0	0	0	0	0	0	0	60	4
09:45	95	0	1	4	248	0	0	0	0	0	0	0	0	0	0	0	0
09:30	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
09:15	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
09:00	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
08:45	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
08:30	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
08:15	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
08:00	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
07:45	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
07:30	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
07:15	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
07:00	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
06:45	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
06:30	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
06:15	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
06:00	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
04:00	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
02:00	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01:00	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SUM:					2539	14	6	0	0	0	0	0	0	0	0	60	4
AVG:	98	0	0	2	127	1	0	0	0	0	0	0	0	0	0	3	0

**Note** You can generate this report using the **perf\_report -rds** command.

## Processor Summary Report

The processor summary report (cmb1.aix\_sum.Aug31.txt) displays general information about system activity.

```

AIX Processor Summary Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

```

	CPU				DISK		SYSTEM ACTIVITY (per sec)					READS(per sec)			WRITES(per sec)			NET(K)	
	%I	%K	%U	%W	ops/s	KB/s	SWTCH	CALLS	FORKS	EXECS	SEMAS	CALLS	CHAR	PHYS	CALLS	CHAR	PHYS	TCP	NOPR
10:00	72	1	3	24	31	377	206	775	0	0	37	14	92	13	16	94	16	6	0
09:45	95	0	1	4	4	63	58	275	0	0	3	3	7	1	2	6	1	0	0
09:30	100	0	0	0	0	2	28	249	0	0	1	1	0	0	0	0	0	0	0
09:15	99	0	0	0	0	2	28	400	0	0	1	1	1	0	0	0	0	0	0
09:00	99	0	0	0	0	2	28	399	0	0	1	1	0	0	0	0	0	0	0
08:45	99	0	0	0	0	2	28	401	0	0	1	1	1	0	0	0	0	0	0
08:30	100	0	0	0	0	2	28	399	0	0	1	1	0	0	0	0	0	0	0
08:15	99	0	0	0	0	2	28	398	0	0	1	1	0	0	0	0	0	0	0
08:00	99	0	0	0	0	2	28	398	0	0	1	1	0	0	0	0	0	0	0
07:45	99	0	0	0	0	2	28	401	0	0	1	1	1	0	0	0	0	0	0
07:30	100	0	0	0	0	2	28	399	0	0	1	1	0	0	0	0	0	0	0
07:15	100	0	0	0	0	2	28	398	0	0	1	1	0	0	0	0	0	0	0
07:00	99	0	0	0	0	2	28	398	0	0	1	1	0	0	0	0	0	0	0
06:45	99	0	0	0	0	2	28	401	0	0	1	1	1	0	0	0	0	0	0
06:30	99	0	0	0	0	2	28	398	0	0	1	1	0	0	0	0	0	0	0
06:15	99	0	0	0	0	3	28	400	0	0	1	1	1	0	0	0	0	0	0
06:00	99	0	0	0	0	2	28	400	0	0	1	1	1	0	0	0	0	0	0
04:00	99	0	0	0	1	7	30	400	0	0	1	1	3	0	0	0	0	0	0
02:00	99	0	0	0	0	3	28	402	0	0	1	1	1	0	0	0	0	0	0
01:00	99	0	0	0	0	2	28	400	0	0	1	1	1	0	0	0	0	0	0
---																			
SUM:																			
AVG:	98	0	0	2	2	24	38	405	0	0	3	2	6	1	1	5	1	0	0

**Note** You can generate this report using the **perf\_report -raix\_sum** command.

## Index Services Report

The index services report (cmb1.inx.Aug31.txt) displays indexing activity.

Index Services Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

CPU	Create	Dir	Get	Dir	Update	Dir	Delete	Dir	Find	Dirs	File	Doc	Unfile	Doc	Find	Folds
CreateFold																

%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA
COUNT	DURA																

10:00	72	1	3	24	0	*	6	0.0	0	*	0	*	0	*	0	*	0	*
0	*																	
09:45	95	0	1	4	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
09:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
09:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
09:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
08:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
08:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
08:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
08:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
07:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
07:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
07:15	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	
07:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*
0	*																	

---

```

06:45 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
06:30 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
06:15 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
06:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
04:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
02:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
01:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 *
-----
SUM:          0          6          0          0          0          0          0          0
AVG:  98 0 0 2 0          0 0.0 0          0          0          0          0          0

```

---

**Note** You can generate this report using the `perf_report -rinx` command.

## Network Clearinghouse (NCH) Report

The network clearinghouse report (cmb1.nch.Aug31.txt) displays activity for NCH objects, properties, and items.

```

Network Clearinghouse Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

```

	CPU	LookupObj	ListObjs	ListProps	RetrItem	CreateObj	DeleteObj	AddItem	DeleteProp											
--	-----																			
%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	
-----																				
10:00	72	1	3	24	0	*	31	0.0	0	*	7	0.0	0	*	0	*	0	*	0	*
09:45	95	0	1	4	0	*	21	0.0	0	*	6	0.0	0	*	0	*	0	*	0	*
09:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
09:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
09:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:15	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:30	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
04:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
02:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
01:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
-----																				
SUM:					0		52		0		13		0		0		0		0	
AVG:	98	0	0	2	0		3	0.0	0		1	0.0	0		0		0		0	

**Note** You can generate this report using the **perf\_report -rnch** command.

## Storage Library (OSAR) Load Report

The storage library (OSAR) load summary (cmb1.ol.Aug31.txt) displays loading activity in the storage libraries.

```
OSAR Load Summary for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan
```

	CPU				OSAR Cartridge Loads									Total
	%I	%K	%U	%W	OSAR A	OSAR B	OSAR C	OSAR D	OSAR E	OSAR F	OSAR G	OSAR H		
10:00	72	1	3	24	4	0	0	0	0	0	0	0	4	
09:45	95	0	1	4	0	0	0	0	0	0	0	0	0	
09:30	100	0	0	0	0	0	0	0	0	0	0	0	0	
09:15	99	0	0	0	0	0	0	0	0	0	0	0	0	
09:00	99	0	0	0	0	0	0	0	0	0	0	0	0	
08:45	99	0	0	0	0	0	0	0	0	0	0	0	0	
08:30	100	0	0	0	0	0	0	0	0	0	0	0	0	
08:15	99	0	0	0	0	0	0	0	0	0	0	0	0	
08:00	99	0	0	0	0	0	0	0	0	0	0	0	0	
07:45	99	0	0	0	0	0	0	0	0	0	0	0	0	
07:30	100	0	0	0	0	0	0	0	0	0	0	0	0	
07:15	100	0	0	0	0	0	0	0	0	0	0	0	0	
07:00	99	0	0	0	0	0	0	0	0	0	0	0	0	
06:45	99	0	0	0	0	0	0	0	0	0	0	0	0	
06:30	99	0	0	0	0	0	0	0	0	0	0	0	0	
06:15	99	0	0	0	0	0	0	0	0	0	0	0	0	
06:00	99	0	0	0	0	0	0	0	0	0	0	0	0	
04:00	99	0	0	0	0	0	0	0	0	0	0	0	0	
02:00	99	0	0	0	0	0	0	0	0	0	0	0	0	
01:00	99	0	0	0	0	0	0	0	0	0	0	0	0	
SUM:					4	0	0	0	0	0	0	0	4	
AVG:	98	0	0	2	0	0	0	0	0	0	0	0	0	

**Note** You can generate this report using the **perf\_report -rol** command.



## Image Prefetch Activity Report

The image prefetch activity report (cmb1.pf.Aug31.txt) displays statistics on image retrieval from optical storage libraries to cache and image prefetching activity.

Image Prefetch Activity Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

CPU	Doc Migration (High,Med Pri)								Doc Migration (Low Pri,incl Prefetches)						
	%I	%K	%U	%W	Reqsts (DOCS)	Reqsts (PAGES)	CacheHits (PAGES)	%HitRatio	DiskReads (PAGES)	Reqsts (DOCS)	Reqsts (PAGES)	CacheHits (PAGES)	%HitRatio	DiskReads (PAGES)	
10:00	72	1	3	24	0	0	0	*	0	0	0	0	*	0	
09:45	95	0	1	4	0	0	0	*	0	0	0	0	*	0	
09:30	100	0	0	0	0	0	0	*	0	0	0	0	*	0	
09:15	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
09:00	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
08:45	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
08:30	100	0	0	0	0	0	0	*	0	0	0	0	*	0	
08:15	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
08:00	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
07:45	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
07:30	100	0	0	0	0	0	0	*	0	0	0	0	*	0	
07:15	100	0	0	0	0	0	0	*	0	0	0	0	*	0	
07:00	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
06:45	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
06:30	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
06:15	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
06:00	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
04:00	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
02:00	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
01:00	99	0	0	0	0	0	0	*	0	0	0	0	*	0	
SUM:					0	0	0		0	0	0	0		0	
AVG:	98	0	0	2	0	0	0		0	0	0	0		0	

**Note** You can generate this report using the **perf\_report -rpf** command.

## Client Page Request Report

The client page request report (cmb1.cpr.Aug31.txt) displays statistics on image retrieval from optical storage libraries and cache from the IDM Desktop Client point of view.

Client Page Request Report for Tuesday, August 31, 2004 (00:00-00:00) (1) elcapitan

```

CPU
-----
%I  %K %U %W Client Page Client Cache  Cache Hit Ratio Disk Reads
      Requests  Hits (PAGES)          (PAGES)
-----
22:00 16  8 11 65    3767         0         0.00      6102
20:00 16  8 11 65    3781         0         0.00      8024
18:00 16  9 11 64    3759         0         0.00      6080
16:00 18  9 11 63    3763         0         0.00      6022
14:00 16  8 11 65    3758         0         0.00      5886
12:00 16  8 11 65    3785         0         0.00      6086
10:00 17  8 11 64    3722         0         0.00      6044
08:00 18  9 11 63    3774         0         0.00      8086
06:00 16  8 11 65    3769         0         0.00      6036
04:00 16  8 11 65    3798         0         0.00      6048
02:00 16  8 11 65    3719         0         0.00      5866
00:00 18  9 11 63    3744         0         0.00      6111
-----
SUM:                49139         0         0.00      76391
AVG:   17  8 11 64    4095         0         0.00      6366

```

**Note** You can generate this report using the **perf\_report -rcpr** command.

## Print Activity Report

The print activity report (cmb1.prt.Aug31.txt) displays information on print services activity.

```
Print Activity Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan
```

	CPU				PrintDocs		PrintData		PSM:PrntPg		PSM:Cancel		Complete	
	%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA
10:00	72	1	3	24	0	*	0	*	0	*	0	*	0	*
09:45	95	0	1	4	0	*	0	*	0	*	0	*	0	*
09:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*
09:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*
09:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*
08:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*
08:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*
08:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*
08:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*
07:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*
07:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*
07:15	100	0	0	0	0	*	0	*	0	*	0	*	0	*
07:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*
06:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*
06:30	99	0	0	0	0	*	0	*	0	*	0	*	0	*
06:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*
06:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*
04:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*
02:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*
01:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*
SUM:					0		0		0		0		0	
AVG:	98	0	0	2	0		0		0		0		0	

**Note** You can generate this report using the **perf\_report -rprt** command.

## Security Services Report

The Security Services report (cmb1.sec.Aug31.txt) displays information about security services activity. The following sample output is an abbreviated example:

```
Security Services Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan
```

	CPU	Logon	Logoff	ChkAccess	ChkMember	GetSecInfo	GetMemberL	GetSysInfo	SetSysInfo												
--	-----																				
%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA
-----																					
10:00	72	1	3	24	0	*	7	0.0	0	*	0	*	0	*	0	*	0	*	0	*	
09:45	95	0	1	4	0	*	3	0.0	0	*	0	*	0	*	0	*	0	*	0	*	
09:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
09:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
09:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
08:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
08:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
08:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
08:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
07:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
07:30	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
07:15	100	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
07:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
06:45	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
06:30	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
06:15	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
06:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
04:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
02:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
01:00	99	0	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*	
-----																					
SUM:					0		10		0		0		0		0		0		0		
AVG:	98	0	0	2	0		1	0.0	0		0		0		0		0		0		

**Note** You can generate this report using the **perf\_report -rsec** command.

## SQL Services Report

The SQL services report (cmb1.sql.Jan31.txt) displays information on SQL services activity.

```
SQL Services Report for Friday, January 31, 1999 (00:00-00:00) (1) corona
```

	CPU				ExecSQL		CreateCurs		ExecCursor		Fetch		DropCursor		BeginT rans		Cancel		Logon	
	%I	%K	%U	%W	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA	COUNT	DURA
09:15	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
09:00	95	4	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:45	96	4	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:30	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:15	95	4	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
08:00	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:45	95	4	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:30	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:15	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
07:00	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:45	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:30	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:15	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
06:00	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
04:00	95	4	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
02:00	96	3	1	0	0	*	0	*	0	*	0	*	0	*	0	*	0	*	0	*
SUM:					0		0		0		0		0		0		0		0	
AVG:	96	3	1	0	0		0		0		0		0		0		0		0	

**Note** You can generate this report using the **perf\_report -rsql** command.

## Service Activity Report

The service activity report (cmb1.sv.Aug31.txt) displays information about service process activity in the FileNet system.

Service activity for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

VWR	CPU VWS	BES WQS	CSM	DOC	INX	NCH	PRI	SEC	SQI		
-----											
%I %K %U %W COUNT DURA COUNT DURA COUNT DURA COUNT DURA COUNT DURA COUNT DURA COUNT DURA											
-----											
10:00	72	1 3 24	14561	0.0	0 *	8 0.0	17 0.0	38 0.0	0 *	38 0.0	0 *
-	-	-	0	*							
09:45	95	0 1 4	789	0.1	0 *	0 *	0 *	27 0.0	0 *	23 0.0	0 *
-	-	-	0	*							
09:30	100	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
09:15	99	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
09:00	99	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
08:45	99	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
08:30	100	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
08:15	99	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
08:00	99	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
07:45	99	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
07:30	100	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
07:15	100	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							
07:00	99	0 0 0	0	*	0 *	0 *	0 *	0 *	0 *	0 *	0 *
-	-	-	0	*							

---

```

06:45 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
- - - - 0 *
06:30 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
- - - - 0 *
06:15 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
- - - - 0 *
06:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
- - - - 0 *
04:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
- - - - 0 *
02:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
- - - - 0 *
01:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
- - - - 0 *
-----
SUM:          15350      0      8      17      65      0      61      0
0
AVG:  98 0 0 2 768 0.1 0      0 0.0  1 0.0  3 0.0  0      3 0.0  0
0

```

---

**Note** You can generate this report using the **perf\_report -rsv** command.

---

## WorkFlo Activity Report

The WorkFlo activity report (cmb1.wfl.Aug31.txt) displays activity associated with the WorkFlo queues and database. In the sample report below, the last "Dura" column heading of the report has been truncated to accommodate available space. Your actual output display presents all field headings.

WorkFlo Activity Report for Tuesday, August 31, 1999 (00:00-00:00) (1) elcapitan

OPEN	CPU UPDT SEL	READ MISC	UPDATE	DELETE	INSERT	DELandINS	COUNT	STARTDUMP	READDUMP
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
%I	%K %U %W	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA
COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA	COUNT DURA
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
10:00	72 1 3 24	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
09:45	95 0 1 4	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
09:30	100 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
09:15	99 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
09:00	99 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
08:45	99 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
08:30	100 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
08:15	99 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
08:00	99 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
07:45	99 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
07:30	100 0 0 0	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *
0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *	0 *



---

```

07:15 100 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 * 0 *
07:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
06:45 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
06:30 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
06:15 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
06:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
04:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
02:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
01:00 99 0 0 0 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 * 0 *
0 * 0 * 0 *
-----
SUM:          0      0      0      0      0      0      0
0            0
AVG:  98 0 0 2 0      0      0      0      0      0      0
0            0

```

---

**Note** You can generate this report using the `perf_report -rwfl` command.

## Checklist

Before you use perf\_report, be aware of the following:

- The performance monitoring and data collection tool, perf\_mon, starts automatically when you start the FileNet software. You can run perf\_report against the existing data collected by perf\_mon and written to the log file (the UNIX default is /fnsw/local/sd/1/perflog; Windows Server default is \fnsw\_loc\sd\1\perflog). However, you can also run perf\_mon from the command line to collect specific or time-dependent performance data.
- Output from perf\_report can be lengthy. You can redirect output to a file or pipe the output to the less command for ease of viewing. For example, the following command redirects output:

```
perf_report /fnsw/tmp/log112196 -rful > /fnsw/tmp/myfile /  
less /fnsw/tmp/myfile
```

and this command pipes the output to the less command:

```
perf_report /fnsw/tmp/log112196 -rful | less
```

---

### Note

The full report with no other options (as used above) generates a full report for every sample taken.

---

If you want to write one report for all samples, run the following command:

```
perf_report -ml -i<n samples> /fnsw/tmp/log112196 >  
/fnsw/tmp/myfile
```

## Procedure

- 1 Select performance data source.

You can use the existing data automatically collected by the perf\_mon tool or run perf\_mon from the command line to retrieve new performance data.

- 2 Run perf\_report.

Enter the perf\_report command with the parameters required to format and display performance data.

## Related Topics

[“getreports” on page 523](#)

[“log\\_dir” on page 699](#)

[“log\\_extract” on page 702](#)

[“perf\\_mon” on page 954](#)

## ppm\_log

### Description

When Protocol Process Manager (PPM) logging is enabled with the debug option of PPMOI, information is written to a circular log buffer inside PPM. When activated, the ppm\_log tool writes the contents of the log buffer to a file.

If logging is activated but ppm\_log is not running, the logged information just cycles through the log buffer. If PPM should hang or if you want to see the log information, you can run ppm\_log at any time. Once started, ppm\_log runs in a loop, logging information from the log buffer to the log file once per second.

The ppm\_log tool runs in the background. You can terminate ppm\_log at any time using either the kill command with either the interrupt (INT) or terminate (TERM) signal. Use the ps command to obtain the ppm\_log process ID, which is required input to the kill command.

---

**CAUTION**

Do not use kill -9 to terminate ppm\_log. Doing so can hang your Image Services system.

---

### Use

Use ppm\_log to write PPM log information to a file for viewing with a text editor. The log information can help you to investigate PPM hang conditions or to analyze problems in OCOR\_listen or PPM software modules.

## Syntax

ppm\_log <logfilename>

**<logfilename>** The full path name of the file to receive logging information. The default for UNIX platforms is /fnsw/local/logs/cor/ppm\_log and for Windows Server platforms is <drive>:\fnsw\_loc\logs\cor\ppm\_log.

## Checklist

Before you use ppm\_log, be aware that you should not use **kill -9** to terminate ppm\_log. Doing so can hang the Image Services system.

## Procedure

No specific procedure is required.

## Related Topics

**[“PPMOI” on page 1030](#)**

Use the UNIX **man kill** command to display the related on-line manual pages.

# PPMOI

## Description

PPMOI is the operator interface to the Protocol Process Manager (PPM) shared library. Use this tool's options to analyze server processes. In addition to displaying the current status and configuration of each FileNet subsystem, use PPMOI to test changes before making them permanent.

## Use

Use PPMOI (Protocol Process Manager Operator Interface) to perform the following tasks:

- Temporarily increase the allowable number of request handlers started on a server
- Debug or temporarily alleviate problems with server processes
- Obtain Remote Procedure Call (RPC) counts and performance statistics
- Obtain statistics for RPC connection queuing

Use the connection service and queuing statistics of PPMOI to properly tune the number of request handlers available on your Image Services server.

**UNIX**

Connection queuing is implemented for UNIX platforms only.

---

You can also capture information with a debugging option. PPM stores log information in a circular buffer. You can write its contents to a file using the ppm\_log tool. (See [“ppm\\_log” on page 1028.](#))

Choosing the appropriate number of service request handlers (also called "server stubs") enhances overall FileNet system performance. You must allocate enough request handlers to provide good subsystem response time. However, specifying too many request handlers for any of the server processes could needlessly use up resources.

PPMOI can help you tune service request handlers when you are troubleshooting a sluggish FileNet system or one that often stops operating due to system overload.

To permanently increase the number of request handlers, you must edit the serverConfig file.

---

**CAUTION**

Before editing the serverConfig file, make a backup copy of the original configuration. This prevents you from losing important configuration details if your new file becomes corrupt during your editing session.

---

**UNIX**

For UNIX platforms, PPMOI attempts to locate the serverConfig file in /fnsw/local/sd. If it doesn't find the file, PPMOI looks for it in the default directory, /fnsw/etc.

**WIN**

On a Windows Server platform, PPMOI searches the following equivalent directories:

```
<drive>:\fnsw_loc\sd  
<drive>:\fnsw\etc
```

For a detailed description of serverConfig file parameters, see the most current *System Reference Guide*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

Before changing the maximum number of request handlers for any subsystem, see [“Service Request Handlers” on page 1048](#).

## Syntax

### PPMOI

When you start PPMOI, the PPMOI> prompt displays at which you can enter PPMOI commands.

To display a list of PPMOI commands, enter **PPMOI** (uppercase) at a system command line prompt, as shown below:

```
corona(root)/> PPMOI
FileNet PPM Operator Interface COMMANDS
Cleanup - Initiates PPM's process cleanup routine.[cleanup|c]
Debug   - Turns PPM_DEBUG on and off.[debug|d]
debugger- Allows server stub to start by itself.[debugger|db]
Help    - Gets you this list.[help|h]
Modify  - Change max or opt value of one entry in table.[modify|m]
Resume  - Restores all max values to original setting.[resume|r]
SHow    - Displays all the entries in the PPM table.[show|sh]
Status  - Displays server stub process status.[status|st]
CULstatus - Displays CUL status.[culstatus|cul]
SUSpend - Changes all max values in the PPM table to 0.[suspend|su]
RPCqueue- Display entries in RPC queues.[rpcqueue|rp]
QPerf   - Display RPC queue performance statistics.[qperf|qp]
Quit    - Terminates the Operator Interface session.[quit|q]
```

See the following descriptions of each PPMOI command.



---

## Commands

This section lists PPMOI commands in alphabetical order. Uppercase letters of the command indicate the minimum number of characters that you can enter to run the command.

### Cleanup

Initiates a PPM cleanup function to get rid of all unused processes. As an option, you can initiate the cleanup any time using this command. The system automatically initiates this function when:

- All server stubs are busy, or
- Before Cormon displays data to avoid displaying information about server processes that have already exited.

### Debug

Turns the debug option in PPM on and off. When toggled on, PPM logs diagnostic messages using the ppm\_log program.

The **debug** option displays the current status followed by a prompt to maintain the current setting or change it:

```
PPMOI> debug
                PPM_DEBUG is OFF
Change PPM_DEBUG to 'on' 'off' or <CR>:
```

Press Enter to maintain the current selection; enter **on** to turn on the debugger; enter **off** to turn it off.

## DEBUGGER

Enables you to turn the debugger on or off.

When you enter **debugger**, a warning message displays informing you whether the debugger is on or off followed by a prompt:

```
PPMOI> debugger
      WARNING WARNING WARNING WARNING WARNING WARNING
WARNING
      The debugger option is for FileNet programming use
only.

              debugger option is OFF

Change debugging option to 'on' 'off' or <CR>:
```

Press Enter to maintain the current selection; enter **on** to turn on the debugger; enter **off** to turn it off.

## Help

Displays a list of PPMOI commands with a brief description of each.

## Modify [<entry#>]

Temporarily changes the maximum and optimum numbers for any of the server processes started by PPM. Use the **SHow [<entry#>]** or **SStatus [<entry#>]** command to obtain a specific PPM table entry number to modify. For example, to change the maximum number of server processes for PPM table entry 16:

```
PPMOI> modify 16
Modify Table - Enter Entry Number or <CR>: 16
Enter maximum value or <CR>: 32
```

To permanently change the maximum values assigned to a request handler, you must edit the serverConfig file.

---

**Note** See also [“Event Log Output” on page 1047](#) and [“Service Request Handlers” on page 1048](#) for more information about when to consider permanent changes to the serverConfig file.

---

## QPerf [<entry#>]

Displays RPC queue performance statistics. QPerf displays the queuing statistics for all remote programs or if requested, for a single entry.

To obtain the number of a PPM entry, run the show or status command. Select a PPM entry number from the **Entry** column of the show report or the **#** column of the status report. See [“QPerf Command Output” on page 1044](#) for a sample of QPerf output and a description of the fields in the report.



Connection queuing is implemented for UNIX platforms only.

**Quit**

Terminates the PPMOI tool.

**Resume**

Resets the maximum processes to the values currently set in the serverConfig file. Use resume after a suspend command to allow new requests for service to be processed. (See [“SUsuspend” on page 1037.](#))

**RPcqueue [<entry#>]**

Displays the state of the connection queue at a particular instant for a single entry or all entries in the PPM table. If you do not specify an entry, this command displays all entries. To obtain the number of a PPM entry, run the [SHow \[<entry#>\]](#) or [SStatus \[<entry#>\]](#) command. Select a PPM entry number from the **Entry** column of the show report or the **#** column of the status report.

**SHow [<entry#>]**

Lists the maximum and optimum numbers for a single entry or all entries in the PPM tables. If you do not specify an entry, this command displays all entries. To obtain the number of a PPM entry, run the show or status command. Select a PPM entry number from the **Entry** column of the show report or the **#** column of the status report. See [“Show Command Output” on page 1038](#) for a sample of show output and a description of the fields in the report.

**S**Status [<entry#>]

Displays connection service statistics by server process, by remote program (such as Batch Entry Services or WorkFlo Queues Services), and by total for all remote programs or a single PPM entry. To obtain the number of a PPM entry, run the show or status command. Select a PPM entry number from the **Entry** column of the show report or the **#** column of the status report. See [“Status Command Output” on page 1041](#) for a sample of status output and a description of the fields in the report.

**C**ULstatus

Used to display the customer embedded Oracle user license status which is now obsolete with the use of the Universal SLAC Key. If this command is entered, the following displays:

```
PPMOI> culstatus
#   Type   MaxNumber   HiWaterNum   CurrentNum   RejectNum
==  =====  =====
There is no CUL enforced
```

**S**Uspend

Changes the maximum value for all entries in the PPM table to zero, which suspends any new requests for service.

---

**Note** Although this command does change all request handlers to zero, it does not prevent existing ones from servicing requests. It only works when issued before PPM has allocated any requests handlers.

---

## Sample Output

The following sections show sample output from issuing PPMOI commands.

### Show Command Output

PPMOI> **show**

Entry	Program Name	Program #	Ver #	Max	Total	Idle	NoPrc	NoCul	MaxQsz
=====	=====	=====	=====	=====	=====	=====	=====	=====	=====
0:	NCHs	2	2	12	0	0	0	0	64
1:	CSMs	134231040	1	12	0	0	0	0	64
2:	DOCs	134231041	1	12	0	0	0	0	64
3:	INXs	134231042	1	12	0	0	0	0	64
4:	PRIs	134231043	1	12	0	0	0	0	64
5:	BESs	134231044	1	8	0	0	0	0	64
6:	PSMs	134231050	2	4	0	0	0	0	64
7:	RMKs	134231051	1	1	0	0	0	0	64
8:	OSIs	134231052	1	3	0	0	0	0	64
9:	DLIs	134231053	1	5	0	0	0	0	64
10:	OSSs	134231054	1	3	0	0	0	0	64
11:	DLsS	134231055	1	8	0	0	0	0	64
12:	BRBs	134231056	1	1	0	0	0	0	64
13:	BRRs	134231057	1	1	0	0	0	0	64
14:	TAPs	134231058	1	1	0	0	0	0	64
15:	dbintf	134231059	1	12	0	0	0	0	64
16:	SECS	134231060	1	12	0	0	0	0	64
17:	SKFs	134231061	1	12	0	0	0	0	64
18:	SQIs	134231063	1	10	0	0	0	0	64
19:	FILs	134231064	1	6	0	0	0	0	64
20:	WQsS	134231066	1	12	0	0	0	0	64
21:	CPTs	134231069	1	32	0	0	0	0	64
22:	RJEs	134231070	1	3	0	0	0	0	64
23:	OSCs	134231072	1	3	0	0	0	0	64
24:	FSMs	134231073	1	6	0	0	0	0	64
25:	CDBs	134231074	1	6	0	0	0	0	64
26:	BRDs	134231075	1	12	0	0	0	0	64

---

27:	BRTs	134231076	1	12	0	0	0	0	64
28:	BRRMKs	134231077	1	12	0	0	0	0	64
29:	VWSs	134231078	1	64	0	0	0	0	64
30:	VWRs	134231079	1	64	0	0	0	0	64

---

### Program Name

The request handler's program name

### Program #

A system-assigned program number

### Ver #

The program's version number

### Max

The currently defined maximum number of request handlers

- If you changed this value using PPMOI, the Max field identifies that temporary value. (See [“Modify \[<entry#>\]” on page 1035](#) and [“SUsuspend” on page 1037](#).)
- If you reset the server or restore the permanent values, this field identifies the maximum number of request handlers set in the serverConfig file. (See [“Resume” on page 1036](#).)

### Total

The total number of request handlers started up since the last reboot of the system

**Idle**

The number of request handlers currently loaded into memory

**NoPrc**

The number of "no process" occurrences for this type of request handler (that is, the number of times this process was not available because the maximum number had been reached)

**NoCul**

The number of RPCs rejected due to the number of times a database management system (DBMS) CUL license limit was exceeded. If this value is not zero, more DBMS licenses should be obtained from the DBMS vendor.

**MaxQsz**

The maximum connection queue size. The value is obtained from the serverConfig file in /fnsw/local/sd or in /fnsw/etc. If the value is not present in serverConfig, the default maximum of 64 is assumed. Queue sizes greater than 64 are lowered to 64.



## Status Command Output

```

PPMOI-3> st
#      Name      Number  Vers MaxProc No Proc No Culs
=====
 0 NCHs          2     2     12      0      0
      Pid      Status      Conns  Avg Time  Min Time  Max Time  RPC  Max Time Stamp
      =====
      190488 available/idle      27   0.0198   0.0027   0.0253  116  2005/06/01 17:54:24
 1 CSMs          1     1     12      0      0
      Pid      Status      Conns  Avg Time  Min Time  Max Time  RPC  Max Time Stamp
      =====
      201950 available/idle      15   5.8323   0.0013   73.8635   5  2005/06/01 18:22:31
 2 DOCs          1     1      1      0      0
      Pid      Status      Conns  Avg Time  Min Time  Max Time  RPC  Max Time Stamp
      =====
      193536 available/idle      16   0.0617   0.0013   0.6568   35  2005/06/01 18:03:25
 3 INXs          1     1      1      0      0
      Pid      Status      Conns  Avg Time  Min Time  Max Time  RPC  Max Time Stamp
      =====
      194906 available/idle      28   0.4814   0.0012   11.0283  12  2005/06/01 18:24:30
16 SECs          1     1      1      0      0
      Pid      Status      Conns  Avg Time  Min Time  Max Time  RPC  Max Time Stamp
      =====
      38448 available/idle      14   0.0261   0.0189   0.0494   0  2005/06/01 17:54:24

      Grand Totals      Conns  Avg Time  Min Time  Max Time
      =====
                100   1.0285   0.0012   73.8635

```

Each entry in the status listing includes the following information:

**#**

The entry number in the PPM table

**Program name**

The request handler name

**Program number**

The request handler's Courier program number

**Vers**

The Courier version number of the request handler

**MaxProc**

The currently defined maximum number of request handlers

- If you changed this value using PPMOI, the MaxProc field identifies that temporary value. (See [“Modify \[<entry#>\]” on page 1035](#) and [“SUsuspend” on page 1037.](#))
- If you reset the server or restore the permanent values, this field identifies the maximum number of request handlers set in the serverConfig file. (See [“Resume” on page 1036.](#))

**No Proc**

The number of "no process" occurrences for this request handler (that is, the number of RPCs rejected when this process was not available because the maximum number had been reached)

**No Culs**

The number of RPCs rejected due to the number of times a database management system (DBMS) CUL license limit was exceeded. If this value is not zero, more DBMS licenses should be obtained from the DBMS vendor.

---

## Pid

The process IDs of the request handlers that are loaded into memory

## Status

The status of the request handler. The table below lists each status and its description that you can see in the report:

Status	Description
available/idle	The request handler is idle and available for an incoming request.
active	The request handler is active.
conInProgress	The request handler is starting up.
terminating	The request handler is terminating.

## Conns

Number of connections processed by this pid

## Avg Time

Average process time of all connections

## Min Time

The fastest process time for any connection

## Max Time

The duration of the longest Remote Procedure Call (RPC)

**RPC**

The remote procedure number of the longest RPC

**Max Time Stamp**

The date and time at the end of the longest RPC

**Totals**

Totals for each column of the report for the named process (for example, CPTs)

**Grand Totals**

Totals for each column of the report for all processes for all programs running at the time of this report

**QPerf Command Output**

```
PPMOI> qp 28
```

#	Name	On	Max	Alloc	OnQ	Hiwt	DQ	# QOps	Avg Time	Min Time	Max Time	
===	=====	==	=====	=====	=====	=====	==	=====	=====	=====	=====	
28	CPTs	1	64	64	0	4	0	795961	0.5247	0.1057	45.8849	
								Totals	# QOps	Avg Time	Min Time	Max Time
								=====	=====	=====	=====	
								795961	0.5247	0.1057	45.8849	

**#**

The entry number in the PPM table

**Name**

The request handler name

**On**

Flag indicating the status of connection queuing. Flag values are:

- 0** Queuing is disabled.
- 1** Queuing is enabled.

**Max**

Maximum number of connections allowed on a connection queue

**Alloc**

Number of connections queue structures that have been allocated

**OnQ**

Number of connections currently on the queue

**Hiwt**

The high water mark indicating the maximum number of queued connections

**DQ**

Flag indicating that a connection is in the process of being dequeued

**# QOps**

The number of queue operations that have been performed since the Image Services software started

**Avg Time**

The average time that a connection spent on the queue

**Min Time**

The minimum time that a connection spent on the queue

**Max Time**

The maximum time that a connection spent on the queue

## Event Log Output

PPMOI is most often used to address problems identified from entries in the event log. Events and errors are logged to the following files:

/fnsw/local/logs/elogs/elog<yyyymmdd> for UNIX servers

<drive>:\fnsw\_loc\logs\elogs\elog<yyyymmdd> for Windows servers

The following is an example of a log entry:

```
1997/04/01 13:23:46 17, 1, 8196 <fnsw>
PPM_GetProcess: no available process, program = 134231061
version = 1.
```

This message indicates that the number of requests for the request handler, identified by the program number 134231061, exceeded the number that the system is configured for. Use the Modify option to temporarily increase the maximum value until the higher demand for that process is reduced.

PPM uses an algorithm to queue connections if all server processes for a given program are busy. Queued connections offer much better system robustness when processing heavy bursts of traffic.

However, if the number of server processes is set too low and connections are consistently queued, it significantly affects server response time. Use the status command and the QPerf command to obtain the total number of connections processed and the total number of connections queued for the remote programs in question.

---

**Note** See also **[“Service Request Handlers” on page 1048.](#)**

---

## Checklist

Before you use PPMOI, be aware of the following:

- PPMOI is directory independent. You can run it from any working directory.
- After restarting FileNet software, any values changed using PPMOI return to their current settings as defined in the serverConfig file. You can also use the Resume option to restore all values to their original settings.

## Procedure

- 1 Enter **PPMOI** at the system command line.

The PPMOI> prompt displays.

- 2 At the PPMOI> prompt, enter commands to display or modify information.
- 3 Enter **quit** when finished.

## Service Request Handlers

Client PCs make requests to the Image Services in the form of a Remote Procedure Call (RPC). A group of related RPCs is called a "remote program," such as the Image Services subsystems DOC, INX, and WQS.

A service request handler is a process on the Image Services that responds to an RPC from a client. There are separate request handlers for each unique remote program. The Courier networking software is the mechanism that delivers RPC requests from clients to the request



handlers for processing, and delivers the responses from the server back to the clients.

PPM is the Courier component that manages service request handlers. PPM creates new request handlers as necessary and maintains status as to whether a given request handler is available or busy. As the "operator interface" for PPM, PPMOI enables you to temporarily alter the maximum number of request handlers allocated for a given remote program.

Once you have temporarily tuned the maximum number of request handlers to enhance system performance, you can make those changes permanent by editing the `/fnswh/etc/serverConfig` file.

Changing the number of service request handlers can profoundly impact Image Services system performance. For example:

- Allocating too few request handlers can cause slow response time or ignored requests.
- Allocating too many request handlers sometimes consumes more resources than available on the server. This can result in longer response times and dropped requests. In extreme cases, it can cause the Image Services server to crash.

## Collecting Statistics

Before you attempt to change the number of request handlers for any remote program, use the appropriate PPMOI command to gather the following information:

Required Data	PPMOI Command
Maximum number of request handlers allowed for the given remote program	<b><u>SHow [&lt;entry#&gt;]</u></b> or <b><u>SStatus [&lt;entry#&gt;]</u></b>
Number of request handlers currently allocated	<b><u>SHow [&lt;entry#&gt;]</u></b> or <b><u>SStatus [&lt;entry#&gt;]</u></b>
Distribution of work among the request handlers	<b><u>SStatus [&lt;entry#&gt;]</u></b>
Number of requests already processed for a given subsystem	<b><u>SStatus [&lt;entry#&gt;]</u></b>
Number of requests that arrived when all the request handlers were busy. A UNIX platform queues these requests. This command does not apply to a Windows Server system.	<b><u>QPerf [&lt;entry#&gt;]</u></b>

## Frequently Asked Questions

After collecting the required information, review the following FAQs before changing the maximum number of request handlers for any given Image Services subsystem:

- 1** My system consistently allocates only half the maximum number of request handlers. Should I reduce this value to reflect what the system actually uses?

Probably not. Image Services only creates a new request handler when all the existing ones are busy when a new request arrives. If your system is running well, and exhibits no resource shortages, the extra request handlers enable the customer to add additional applications or end users with less risk of exceeding the request handler limit without having to constantly tune the server.

It is unlikely that you would have to reduce the FileNet default settings to avoid resource shortages. Resource shortage usually occurs when users do not adhere to the recommendations provided in this document.

- 2** My system hit the request handler limit for a remote program. Do I need to increase the maximum number of request handlers for this remote program?

It depends on the RPC arrival patterns and how long it takes the server to process those RPCs.

- If the RPC traffic is steady-state, and the server is using the maximum number of configured request handlers, you should probably increase the limit.

- If the RPC traffic is very bursty, the system should continue to perform acceptably without increasing the number of request handlers.

Use the PPMOI **STatus [<entry#>]** command to estimate these RPC traffic patterns by examining the distribution of RPCs to request handlers. The following example shows the distribution of work to request handlers for bursty RPC traffic.

#	Name	Number	Vers	MaxProc	No Proc	No Culs
1	CSMs	134231040	1	8	0	0
Pid	Status	Conns	Avg Time	Min Time	Max Time	
19522	active	64377	1.2403	0.0050	33.0783	
19587	active	15825	0.9688	0.0049	10.3621	
19642	active	16746	1.3207	0.0048	31.0250	
20352	available/idle	5158	1.1394	0.0048	31.0035	
20353	available/idle	962	1.0711	0.0050	30.9743	
26388	available/idle	101	1.1322	0.0052	30.7579	
5787	available/idle	5	0.7566	0.3190	1.2366	
6896	available/idle	1	0.9426	0.9426	0.9426	
	Totals	103175	1.2049	0.0048	33.0783	

Note that, in this case, the first four request handlers have processed about 99% of the requests (102106 out of 103175), while the remaining four request handlers have processed the remaining 1% of the requests. Statistics like this indicate that, although the maximum number of request handlers have been allocated, half of them are barely used. Therefore, you would not need to increase the number of request handlers at this time.

If the RPCs arrive at a steadier rate, PPM would distribute them more evenly to request handlers than shown in the previous example.

- 3 Other than using PPMOI to temporarily allocate the maximum number of request handlers, what other method can I use to help determine when I need to increase the configured value?

**UNIX**

If the Image Services server is running on a UNIX platform, you can use the PPMOI QPerf command. When an RPC arrives on a UNIX-based Image Services server, and all the request handlers are busy, PPM queues the RPC until a request handler finishes its current request and becomes available to process the new one. By examining the queue statistics for a remote program, you can determine by how much the demand for a given remote program has exceeded the configured number of request handlers.

This queuing mechanism can increase the capacity of the Image Services to handle bursts of RPC traffic. However, the process of queuing the request until a request handler is available imposes measurable overhead on the processing of an RPC.

In other words, for a UNIX-based Image Services server:

- If the output of the PPMOI **STatus [<entry#>]** command indicates that RPC traffic for a given remote program is bursty, it is more acceptable to run with a smaller number of request handlers, let them all become busy, and let the queuing mechanism handle the occasional overflow. In this case, there is less motivation to increase the maximum number of request handlers.

- If the RPC counts of the request handlers are more evenly distributed (less bursty), then letting all the request handlers become busy can constantly invoke the queuing mechanism, increasing the system overhead required to process requests. In this case, you should increase the maximum number of request handlers.

As a guideline, for bursty traffic, it can be tolerable to queue 10% of the total number of request handlers (from the SStatus command). For steady state traffic, queuing any more than 1% of the requests can be unacceptable.

- 4 Since Windows-based Image Services servers do not support connection queuing, what happens to requests for service when all configured request handlers for a particular service (subsystem) are busy?

**WIN**

When all server stubs are busy on an Windows-based Image Services server, Courier "sleeps" for a short time, then tries sending the request again. Courier repeats this sequence until it finds an available request handler to process the request. If all request handlers for the particular service remains busy after several attempts, Courier rejects the RPC, returning an error message to the client server.

- 5 The number of clients using the server just doubled. Should I double the number of request handlers configured for each remote program?

No. As the load on a server approaches its capacity, the dividing line between enough request handlers and too many request handlers becomes narrower. The incremental resource consumption for each request handler can be far more than one would expect.

For example, each INX server process can also require one or more new database processes. This is the secondary resource consumption to consider when evaluating such a configuration change.

When a server has limited resources, increasing the number of request handlers often reduces throughput, as the server OS spends more time managing additional processes that cannot contribute to increased output. Other times, the server capacity becomes saturated with request handlers when the server does not have the resources to handle any more, regardless of how high the you increase the limit.

- 6 How do I dynamically change the maximum number of request handlers for a remote program?

The **Modify [<entry#>]** command in PPMOI enables you to increase the maximum number of request handlers. This command has immediate effect, and helps when "rescuing" a poorly configured server.

Note that using the modify command to reduce the maximum number of request handlers cannot prevent PPM from using any pre-existing request handlers. It only prevents PPM from creating more than the new lower maximum. For example, if PPM has allocated eight request handlers, changing its number to zero cannot prevent any RPCs from being processed by the ones that already exist. It will, however, prevent PPM from allocating another new request handler.

- 7 Can I kill request handlers if I think the system has allocated too many?

No. Doing this can corrupt data or shut down the Image Services server.

## Related Topics

**[“ppm\\_log” on page 1028](#)**

**[“whatsup” on page 1309](#)**

## PRI\_tool

### Description

PRI\_tool is a utility that helps you manage printers and print requests in your FileNet system through a series of commands you enter from the PRI\_tool prompt. PRI\_tool has two modes of operation: command line input and interactive command parameter input. Command line input mode allows those familiar with PRI\_tool commands to directly enter commands, options, and parameters at the command line. Interactive input mode prompts you for the command parameters.

### Use

Use PRI\_tool to manage printers and print requests and to obtain statistics about print-related activities. You can suspend, redirect, and cancel print requests, as well as suspend activity on individual printers. You can also use PRI\_tool to display a printer's status and other print-related information.



## Syntax

### PRI\_tool

When PRI\_tool starts, the <PRI\_tool> prompt displays. Enter commands at the prompt or enter ? to display the following:

```
> PRI_tool
Type '?' for help
<PRI_tool>?
Type 'help <command>' for more detailed help on one command.
Type 'help *' for detailed help on all commands.
Type '<command> ?' for interactive command parameter input.
Application commands:
cachestatus      cancel           checkcache
clearrequests   hardcopy        help
modify          printerstatus   quit
requeststatus   resumeprinter   systemstatus
termoff         termon
```

See [“Commands” on page 1058](#) for a description of individual PRI\_tool commands.

## Commands

### cachestatus

The cachestatus command displays the number of objects fetched for each cache and for each printer. Use this command primarily to determine if the number of pages fetched is close to the cache full threshold, a condition that prevents a printer from printing.

```
<PRI_tool> caches
Summary statistics for caches (all printers):
  Cache 0: sys_print_cache1:costa5:FileNet
           min/max_pgs=409/819, #sects=81920, pgs_fetched=2
Statistics on each cache for each printer:
  Printer 0: FES3canon:costa5:FileNet
           Cache 0: sys_print_cache1:costa5:FileNet
                min/max_full=20/50, min/max_pgs=163/409, pgs_fetched=0
...
  Printer 11: fax4:costa5:FileNet
           Cache 0: sys_print_cache1:costa5:FileNet
                min/max_full=20/50, min/max_pgs=163/409, pgs_fetched=2
```

The first section shows summary statistics for each cache, including:

Cache	The name of the cache being displayed
min_pgs	The cache threshold at which the per printer limit changes
max_pgs	The maximum number of pages the cache holds

#sects	The number of sectors in the cache
pgs_fetched	The number of pages for which a migrate has been initiated for the listed cache, regardless of which printer the corresponding request is queued to. This counter increments for each migrate and decrements when a page is deleted. The counter does not reflect the number of pages in the cache because it increments <b>before</b> the migrate completes.

The second section contains the number of pages migrated into each cache for each printer.

Cache	The cache being displayed
Printer	The printer being displayed. Information displays for each printer and cache combination.
min_full	The minimum cache fullness threshold specified as a percentage. The default is 20 percent.
max_full	The maximum cache fullness threshold specified as a percentage. The maximum is 50 percent.
min_pgs	The minimum cache fullness threshold specified as a number of pages
max_pgs	The maximum cache fullness threshold specified as a number of pages
pgs_fetched	The number of pages for which a migrate has been initiated without a delete for the listed cache and printer

The cache fullness affects the number of migrates the system attempts in the following ways:

- If pages fetched is less than minimum pages for a given cache (first section), the system fetches up to the maximum pages limit specified for the cache and printer (second section).
- If pages fetched is greater than minimum pages and less than maximum pages for the cache (first section), the system fetches up to the minimum pages limit for the cache and printer (second section).
- If pages fetched is greater than or equal to maximum pages for the cache (first section), no additional fetching is done for any printer.

## cancel

The cancel command cancels the indicated requests. The syntax of the cancel command is:

```
cancel [<request_ID>] [file=<filename>] [user=<username>]
      [printer=<printername>] [priority=<priority>]
```

If you include multiple options, the program cancels only the requests that satisfy all options. The following table describes each option:

Cancel Option	Description
request ID	The request ID (job number) to cancel
filename	Name of the file that contains a list of request IDs, one request ID per line
username	Cancel only requests from this user
printer name	Cancel only requests from this printer
priority	Cancel only requests with this priority

### checkcache

This option checks the print cache for objects that don't belong there. It suspends all activity by print services while it runs. The time could be anywhere from a few seconds to approximately 15 minutes for systems with 100,000 or more pages waiting to be printed. If necessary, you can abort this command with the Control+c key sequence.

### clearrequests

The clearrequests command deletes all print requests and removes all objects from the print caches.

You might use this command to clear requests that have frozen a Print server. For example, if the Print server violates a protocol, it dumps the trace buffer to a file and logs an appropriate message to sys\_log.

### hardcopy

Use hardcopy to send session output to a magnetic disk file or to turn off hardcopy output. The syntax of the hardcopy command is:

```
hardcopy [<filename>]
```

<filename> is the name of the file that receives the output. If you do not specify a file name, the command turns off hardcopy output.

The following command turns on hardcopy and sends the output to the file named **outfile**:

```
hardcopy outfile
```

**help [<command>]**

This command displays help information for the specified command. If you do not specify a command, help text for **all** commands displays.

To display the complete help text file, enter **help \***. More information displays than can fit in one window so this command is only useful if you are working in a scrollable X-window or MS-DOS window that allows you to see information at the start of the file.

If a command has its own parameters, you can use another form of help, called "interactive help." When you use interactive help, PRI\_tool prompts you for each individual parameter. To use interactive help, enter the command followed by a question mark (?). Each parameter displays in sequence and waits for you to enter a response. For any parameter you do not want to specify, press Enter to accept the default.

Below are two examples using the cancel command. The first example uses the default for each prompt by pressing Enter (indicated by **<cr>**). The second example enters the request ID (5002), making PRI\_tool cancel the print job immediately.

```
<PRI_tool> cancel ?
request_id (CR for other options)? <cr>
name of file with request ids (CR if none)? <cr>
only requests from user (CR=any)? <cr>
only requests from printer (CR=any)? <cr>
only requests with priority (CR=any)? <cr>

<PRI_tool> cancel ?
request_id (CR for other options)? 5002
Request 5002 canceled, status=queued
<PRI_tool>
```

## modify

Use the modify command to change a print request. The command changes either a request ID or a file containing a list of request IDs (one request ID per line). The syntax of the modify command is:

```
modify [<request_id>] [fromfile=<filename>] [fromuser=<user-
name>]
    [fromprinter=<printer>] [frompriority=<priority>]
    [frompapersize=<papersize>]
    [papersize={letter,lgal,b,c,d,e,a0,a1,a2,a3,a4,a5,b4,b5,18x24,top,
        bottom,third,default,half_letter,best_avail,10x14,executive}]
    [scaling={normal,clipboth,exact,approx,original,cener,enhanced_
        exact}]
    [orientation={default,landscape,portrait,no_rotate}]
    [note=<note>] [priority={0..9}]
    [printer=<object:domain:organization>] [copies=<#copies>]
```

```
[overlay={none,firstpage,allpages}] [eject_tray=<eject_tray#>]
[print_time=<yyyy/mm/dd hh:mm:ss>]
[security=<read_name> <write_name> <exe_name>]
[collate] [annotations] [request_header] [doc_headers]
[phone_num=<phonenumstr>] [headline=<headlinestr>]
[fax_mode={coarse,fine}]
```

If you specify more than one of the options that start with **from** (fromprinter, frompriority, or frompapersize), a request must satisfy **all** of those options before PRI\_tool can modify it.

If printing has already started for a particular request ID, you can change only the following options:

- printer
- paper size
- overlay
- reset the priority to 0 (which suspends the request)

---

**Note** Any modification you make with these options causes the system to automatically cancel and resubmit the request, which in turn can change the printing order relative to other requests.

---



The following table briefly describes modify command options. For more detailed definitions, see the "Printing" chapters of the Client-side documentation.

Modify Option	Description
<request_ID>	Identification number (ID) of the request you want to modify
fromfile	Name of a file containing request IDs
fromuser	Modifies only requests from this user
fromprinter	Modifies only requests from this printer
frompriority	Modifies only requests with this priority
frompapersize	Modifies only the requests that are using this paper size
paper_size	Paper size to use. Choose from letter, legal, b, c, d, e, a0, a1, a2, a3, a4, a5, b4, b5, 18x24, top, bottom, third, default, half_letter, best_avail, 10x14, and executive. Available options are based on device configuration. For example, some devices cannot support some paper sizes or fax devices cannot be equipped with multiple paper trays.
scaling	Scaling option to use. Choose from normal, clipboth, exact, approx, original, center, and enhanced_exact
orientation	Specifies which side is up. Choose from default, landscape, portrait, no_rotate.
note	An ASCII string to print on the header page
priority	Priority of job, 0=lowest and suspends printing; 9=highest
printer	Three-part NCH name of a printer. If not specified, PRI picks the best printer device available.
orientation	Specifies which side is up. Choose from default, landscape, portrait, no_rotate.
copies	Number of copies to print
overlay	Overlay no pages, the first page, or all pages. When specifying first page or all pages, the first page contains the overlay data.
eject_tray	Number of the output tray at which pages are to be delivered. Available options are based on how device configuration.

Modify Option	Description
print_time	Time to start printing using the system's default date format. For example, if your system is configured with a date mask of "day mon dd hh:mm:ss yyyy," you would enter the date using a format as shown in this sample: Mon Nov 15 12:30:00 1998)
security	Security names for read, write, and append/execute
collate	Collates pages. (Print entire copy of each document before beginning the second copy.)
annotations	Prints annotations
request_header	Uses a header page for the print request
doc_headers	Uses document header pages
phone_num	For FAX requests, phone number to dial
headline	For FAX requests, text of headline to print at top of each page
fax_mode	For FAX requests, choose coarse or fine resolution. Available options are based on device configuration. For example, some fax devices cannot be configured to support multiple resolutions.

### **printerstatus**

Use this command to get a long or short status report for one or all configured printers and FAX machines.

```
printerstatus [<printer_name> [nonfax] [faxonly] [long]]
```

Specifying the nonfax option displays only printer devices; the faxonly option displays only fax devices.

Entering the command with no parameters displays a report similar to the following:

```
<PRI_tool> printerstatus
#  printer print fetch requests pages idle
   namerequest request queued queued time
0 PRT3canon:costa5:FileNet11 suspended
1 RIC20_4:costa5:FileNet00 9632
2 Xerox:costa5:FileNet00 suspended
3 Versatec:costa5:FileNet00 9632
4 fax1:costa5:FileNet00 down
5 fax2:costa5:FileNet74675050 down
6 fax3:costa5:FileNet752822 down
```

The fields of the printstatus report are described in the following table:

Field Name	Description
#	Printer number
Printer name	Three-part NCH name of the printer
Print request	Request ID of the request that is currently printing
Fetch request	Number of outstanding fetch requests
Requests queued	Total number of print requests queued to this printer
Pages queued	Total number of pages queued to this printer
Idle time	Number of seconds the printer has been idle (not printing) or one of the following status conditions: unknown, down, suspended, needs_attention, needs_service, available, or redirected

Use the long option to obtain detailed information on one of the printers. Specify the printer as shown below in the sample output for a long status for printer PRT3canon:

```
<PRI_tool> printerstatus PRT3canon long
Printer 0 : PRT3canon:costa5:FileNet
Status: suspended
Print error:(n.a. -- suspended)
Request printing:none
Embedded migrate:(n.a. -- suspended)
Next migrate:*
Time since print:3567
Requests queued:1
Pages queued:1
Pages printing:0
Avail papersizes:(n.a. -- suspended)
Config papersizes:letter,legal,a4,b5,top,best_
avail,default,dont_care
```

The following table describes each field of the long status report:

Field Name	Description
Status	Current status of the printer. Valid status includes unknown, down, suspended, needs_attention, needs_service, available, or redirected. If the status of the printer is reported as <b>redirected</b> , the destination printer name also displays.
Print error	Error tuple for an error that is preventing a print operation or 0 if there is no error. Use the fn_msg tool to see the message text for the error.
Request printing	The ID of the request that is currently printing, the number of pages printed for this request, and the total number of pages in the request
Embedded migrate	The ID of the document being migrated due to an embedded document reference and the name of the document service receiving the request
Next migrate	If the printer is waiting for a page, this is the name of the service and the document ID. Otherwise, this field shows <b>done</b> .

---

Field Name	Description
Time since print	The elapsed time, in seconds, since the last page was printed
Requests queued	The number of requests queued to this printer
Pages queued	The number of pages queued to this printer
Pages printing	The number of pages submitted to the Print server hardware but not completed
Avail papersizes	Currently loaded paper sizes
Config papersizes	The complete list of paper sizes configured for the printer

### quit

Exits PRI\_tool. The Control+c key sequence also exits the program.

### redirectprinter <from\_printer> to <to\_printer>

---

**Note** This command does not appear in the help information for PRI\_tool unless you have specified the ALLOWUPDATES command.

---

The redirectprinter command redirects all present and future print requests from one printer to another.

**<from\_printer>** is the name of the printer from which output is being redirected. If you do not specify a <from\_printer> name, PRI\_tool prompts you for the name.

**<to\_printer>** is the name of the printer to which print requests are being transferred. If you do not specify a <to\_printer> name, PRI\_tool prompts you for the name.

Use the resumeprinter command when you want to cancel the redirection.

## requeststatus

Displays the status of one or more print requests.

```
requeststatus [<request_ID>] [user=<username>] [long] [docs]
              [priority=<priority>] [printer=<printername>]
```

The table below describes the requeststatus command options:

requeststatus Option	Description
request ID	The ID of a print request. If you do not specify an ID, the display includes all requests that satisfy other options.
username	The logon name of a user whose print requests you want to view
long	Displays detailed information
docs	Generates a list of the items being printed: document ID for documents and object ID/system serial number for cache objects
priority	Display requests for the specified priority (0–9)
printername	Displays requests queued to the specified printer. Requests appear in their print order. (When you do not specify a printer, requests display in ascending request ID order.)

The following is a partial listing of the information printed when you do not specify a request ID:

```
<PRI_tool> requeststatus
request  total      pages      printer
id       pages      printed    status     name
7466     1           0          queued    PRT3canon
7467     1           0          printing  fax4
7468     1           0          fetching  fax4
7469     1           0          queued    fax4
```

The following is a long report for a single request ID:

```
<PRI_tool> requeststatus 5003 long
request_id=5003, request_status=queued, print_err=00000000, fax_
request=f,
submit_time=93/12/8 16:00:58, print_time=n.a., done_time=n.a., copies=1,
priority=4, paper_size=dont_care, collate=f, two_sided=f, annotate=f,
req_header=t, doc_headers=f, scaling=normal, orientation=default,
printer=FEScanon:costa3:FileNet, user=SysAdmin:costa3:FileNet, form_
name='', note='', eject_tray=0, total_pages=1, pages_printed=0
```

The next sample shows the information displayed by the docs option:

```
<PRI_tool> requeststatus docs
req_id=5000, total_pages=1, status=printing, printer=printflo
objectid=4160000000, ssn=14059, 1stpg=1, lastpg=1, service=app_print_cache_
1:costa3:FileNet
-----
req_id=5003, total_pages=1, status=queued, printer=FEScanon
doc_id=30093216, 1stpg=1, lastpg=1, service=DocServer:costa3:FileNet
-----
req_id=5004, total_pages=1, status=queued, printer=FEScanon
doc_id=30093216, 1stpg=1, lastpg=1, service=DocServer:costa3:FileNet
```

### **resumeprinter <printer name>**

The resumeprinter command starts up printing on a printer that was previously suspended or redirected.

### **systemstatus**

The systemstatus command gives overall system status information similar to the following example:

```
<PRI_tool> systemstatus
Total requests queued:53   Total requests printed:0
Total pages queued:53     Total pages printed:0
```

### **termoff**

The termoff command turns off output to the terminal. Use this command in conjunction with the hardcopy command to get large listings into a file without waiting for terminal I/O.



## termon

The `termon` command turns on output to the terminal. Use `termon` to restart output to the terminal after using the `termoff` command.

## Checklist

Before you use `PRI_tool`, be aware of the following:

- FileNet software and network software must be running.
- You must be logged on to the server as a member of the `fnadmin` group or be logged on to Image Services as the `SysAdmin` user. (Refer to the “Introduction” chapter of the *System Administrator’s Handbook* for logon information. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).)
- Run `PRI_tool` on the server where Print Services is running (usually the Storage Library server). You can remotely login to the server, if necessary.
- To disable or re-enable a printer, you must supply the printer’s object name. If you remotely login on a UNIX system to run `PRI_tool`, you must supply the three-part NCH name of the printer to be deleted or re-enabled.
- If you are running `PRI_tool` from the same domain and organization that owns the printer, you only need to specify the printer’s object name. Otherwise, you must use the three-part NCH name when prompted for a printer name.
- You can display printer names using `fn_edit` (UNIX) or the System Configuration Editor (Windows Server).

- If you enter an incorrect name while disabling or re-enabling a printer, the prompt repeats. If you do not know the name of the printer and want to exit from the prompt, enter the Control+c key sequence.
- Some commands, for example, redirectprinter, do not appear in the help information for PRI\_tool unless you have specified the AL-LOWUPDATEs command and provided a password.
- If a printer encountered problems, you might have to redirect its print jobs to another printer. Re-enabling the original printer does not automatically redirect those print jobs back to the enabled printer.

## Related Topics

[“CSM\\_tool” on page 204](#)

[“MKF\\_tool” on page 801](#)

## QLG\_dump

### Description

QLG\_dump displays the quick logging log data from one of the following log files:

/fnsw/local/logs/qlogs/1/QLG<yyyymmdd> on UNIX platforms

<drive>:\fnsw\_loc\logs\qlogs\1\qlg<yyyymmdd> on Windows platforms

where:

QLG<yyyymmdd> is the name of the log file to which statistics are written.

<yyyymmdd> is the current system date in year, month, and day format.

### Use

Use QLG\_dump to display the quick logging entries in the log file. You can start quick logging either by using the FileNet System Configuration Editor or by using the QLG\_start tool.

### Syntax

```
QLG_dump <log_file_name>
```



**Access type** A number representing how the document was accessed: 1=commit, 2=retrieve, 3=delete, 4=modify attributes.

## Procedure

No specific procedure is required.

## Checklist

Before you use QLG\_dump, be aware of the following:

- QLG\_flush writes any in-memory QLG buffer data to today's QLG log file, so the QLG\_dump report of today's log file contains the most up-to-date information.
- If you start quick logging with the QLG\_start command, QLG\_flush terminates quick logging. If you start quick logging with the FileNet System Configuration editor, QLG\_flush does not terminate quick logging.

## Related Topics

[\*\*“QLG\\_flush” on page 1078\*\*](#)

[\*\*“QLG\\_start” on page 1080\*\*](#)

## QLG\_flush

### Description

QLG\_flush stops the quick logging writer process and writes log data from the quick logging program's in-memory buffers to one of the following log files:

/fnsw/local/logs/qlogs/1/QLG<yyyymmdd> on UNIX platforms

<drive>:\fnsw\_loc\logs\qlogs\1\qlg<yyyymmdd> on Windows platforms

where:

QLG<yyyymmdd> is the name of the log file to which statistics are written.

<yyyymmdd> is the current system date in year, month, and day format.

### Use

Use QLG\_flush to stop the log writer process that you started with the QLG\_start tool, and write its data to the quick logging log file.

---

**Tip** To restart quick logging, run the QLG\_start tool.

---

If you enabled quick logging through the FileNet System Configuration editor, QLG\_flush does not stop quick logging. In this situation you can use QLG\_flush to force data to be written to the QLG log file while quick logging continues.

Use QLG\_dump to examine the quick logging entries in the log file.

## Syntax

QLG\_flush

## Example

The following example stops quick logging:

```
corona(fnsw)/usr/fnsw> QLG_flush  
flushed log entries to file: QLG20051109
```

## Procedure

No specific procedure is required.

## Checklist

Before you use QLG\_flush, be aware of the following:

- If you started quick logging with QLG\_start, you can also stop quick logging by rebooting the server.
- If you enabled quick logging through the FileNet System Configuration editor, QLG\_flush does not disable quick logging over a server reboot. You must disable quick logging from the Configuration Editor.

## Related Topics

[\*\*“QLG\\_dump” on page 1075\*\*](#)

[\*\*“QLG\\_start” on page 1080\*\*](#)

## QLG\_start

### Description

QLG\_start starts up quick logging. When you start quick logging, the program begins logging usage statistics to one of the following log files:

/fnsw/local/logs/qlogs/1/QLG<yyyymmdd> on UNIX platforms

<drive>:\fnsw\_loc\logs\qlogs\1\qlg<yyyymmdd> on Windows platforms

where:

QLG<yyyymmdd> is the name of the log file to which statistics are written.

<yyyymmdd> is the current system date in year, month, and day format.

You can also use the FileNet System Configuration Editor to enable quick logging. You can stop quick logging by running the QLG\_flush tool or by rebooting the server. However, if you enabled quick logging with the Configuration Editor, quick logging stays enabled even if you reboot the server. If you enabled quick logging through the Configuration Editor, you must disable it with the Configuration Editor.

QLG\_start begins logging the number of pages retrieved, documents committed, and documents deleted.

---

**Note** To make quick logging as efficient as possible, documents committed with fast batch committal only appear in the log file if quick logging was started by fn\_edit. They will not appear in the log if quick logging was started by QLG\_start.

---



## Use

Use the QLG\_start program to initiate quick logging of usage statistics.

## Syntax

QLG\_start (UNIX platforms)

start QLG\_start (Windows Server platform)

---

**Note** When using this tool on a Windows Server system use the “start application\_name” syntax. When this is done, a new window will be created for the QLG\_start program while the original window is available for more commands. DO NOT close the newly created window until a recycle is required or you could lock up the Storage Library Control application (error 202,1,2009).

---

## Example

The following example starts the quick logging process with the QLG\_start command and stops quick logging with QLG\_flush:

```
corona(fnsw)/usr/fnsw> QLG_start
Allocated 512 entry (10240 byte) log buffer
QLG: starting log writer (DailyAutoCheckpoint)
corona(fnsw)/usr/fnsw> QLG_flush
flushed log entries to file: QLG20051109
```

Use QLG\_dump to examine the entries in the log file.

## Checklist

Before you use QLG\_start, be aware of the following:

- You can stop quick logging by running QLG\_flush (see **“QLG\_flush” on page 1078**) or by rebooting the server.
- If you enable quick logging through the FileNet System Configuration editor, QLG\_flush does not disable quick logging over a server reboot. You must disable quick logging from the Configuration Editor.

## Procedure

To gather usage information, do the following:

- 1 Enter **QLG\_start** on the command line (**start QLG\_start** on Windows).
- 2 Store the captured records.

When you have recorded enough information, force the QLG entries out to magnetic media by entering the following command:

### **QLG\_flush**

- 3 Examine the data in the log file /fnsw/local/logs/qlogs/1/QLG<yyyymmdd> (UNIX) or <drive>:\fnsw\_loc\logs\qlogs\1\qlg<yyyymmdd> (Windows Server) by entering:

### **QLG\_dump <log\_file\_name>**

## Related Topics

[“QLG\\_dump” on page 1075](#)

[“QLG\\_flush” on page 1078](#)

## remove\_docs\_ref

### Description

This tool removes any and all references to the specified surfaces from the MKF PermDB document locator (doc locator) database table. Documents will be deleted from the doc locator table and index table (DOCTABA) if no other surface references exist in the doc locator table. This tool along with the [remove\\_surf](#) tool make up the surface removal feature that can be run on all Image Services platforms and should always be run in tandem, running the remove\_docs\_ref tool prior to running the remove\_surf tool. This feature works with both optical (5.25" and 12") and MSAR surfaces and it could be used for both primary and tranlog surfaces. For two-sided optical surfaces, both sides must be removed in the same transaction.

### Use

Prior to the development of this tool and the remove\_surf tool, Image Services did not provide a way to remove all references to a surface without erasing the surface, nor was there a way to remove a group of surfaces from the database in a safe and efficient manner.

---

**Note** It is best to run **remove\_docs\_ref** when the system is idle or there is a minimum of activity. DO NOT run this tool during times of high system activity as there could be some negative performance ramifications.

---

An example of when these tools might be used is if you were converting your primary optical surfaces to MSAR surfaces and then you want to remove all references to the tranlog optical surfaces. The remove\_docs\_ref command line tool is run with either the -s option or the -f option. The -s option is designed to remove one surface ID at a

time. The -f option allows you to remove multiple surfaces by including a list of surface IDs in an ASCII file.

This tool should typically be run before running the [remove\\_surf](#) tool.

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3)

The output of the `remove_docs_ref` tool could report some confusing information when being used with the FileNet P8 Content Federation Services functionality. Since the function of `remove_docs_ref` is to delete references to a group of surfaces from the DOCS table and the DOCTABA table, but not the images themselves, the CE catalog entry could reference an image that it believes does not exist when the documents are stored on the Image Services system but indexed on the Content Engine system.

Additionally, in the instances where documents are indexed on the Image Services system and the `remove_docs_ref` tool is run, Index Services will create an `export_log` entry to tell the FileNet P8 CFS Server for Image Services to delete the corresponding index entry on the Content Engine system. Refer to the table below to determine when documents on the IS server will not have the index information associated with them (the **Ns**) because if index information does not exist in DOCTABA for an image, there is no way to notify the Content Engine system to delete its catalog information for that image.

Document Activity	Index on CE	Index on IS
New documents entered via Content Engine	Y	<b>N</b>
New documents entered via Image Services		
<ul style="list-style-type: none"> <li>Indexed on both IS and C</li> </ul>	Y	Y

Document Activity	Index on CE	Index on IS
<ul style="list-style-type: none"> <li>Indexed only on CE</li> </ul>	Y	<b>N</b>
Existing document images transferred to CE		
<ul style="list-style-type: none"> <li>Indexed on both IS and CE</li> </ul>	Y	Y
<ul style="list-style-type: none"> <li>Index deleted from IS</li> </ul>	Y	<b>N</b>

## Syntax

```
remove_docs_ref {-s <surface ID>|-f <surface list file>}
```

**-s <surface ID>** Removes any reference in the document locator table (surface\_id\_1 or surface\_id\_2 field) to the specified surface ID.

**-f <surface list file>** Removes any reference in the document locator table (surface\_id\_1 or surface\_id\_2 field) to a group or list of surfaces (all of the surface IDs entered in the specified ASCII text file). The ASCII file name entry needs to be a full path and file name. For example, /fnsw/local/tmp/docid.txt (UNIX) or C:\fnsw\_loc\tmp\docid.txt

---

**Note** This tool is best run with a complete list of surfaces entered into an ASCII text file (the -f option) instead of running it multiple times with a single surface each time (the -s option), because this tool does a sequential search in the docs table which could take a long time. If the tool is run with no parameters, the usage statement will display.

---

## Sample Output

The following sample is output to the log file from remove\_docs\_ref when run with the -s option specified.

```
corona(root)/> remove_docs_ref -s 3004
Job started at 06/11/2003 15:55:35

Option -s 3004

Begin sequentially scanning the DOCS table. It will take a while ....

#docs table entry scanned = 2501

Successfully process the following surface(s):

Surface  server  surf_info.  surf_info.  surf_info.  surf_info.  total
          num_act_docs  num_del_docs  num_act_docs  num_del_docs  updated
          (before)      (before)      (after)      (after)      (doc count)
=====  =====  =====  =====  =====  =====  =====
          3004      2           501           0           0           501           501

Job completed successfully at 15:15:36
```

The following sample is output to the log file from `remove_docs_ref` when run with the `-f` option specified.

```
corona(root)/> remove_docs_ref -f docid.txt
Job started at 06/11/2003 15:55:35

Option -f docid.txt

Begin sequentially scanning the DOCS table. It will take a while ....

#docs table entry scanned = 2501

Successfully process the following surface(s):

Surface   server  surf_info.      surf_info.      surf_info.      surf_info.      total
          num_act_docs  num_del_docs  num_act_docs  num_del_docs  updated
          (before)      (before)      (after)       (after)       (doc count)
=====  =====
          3004         2             501            0              0              501            501
          3006         3            1000           0              0             1000           1000

Job completed successfully at 15:15:36
```

## Checklist

Before you use `remove_docs_ref`, be aware of the following:

- The user running the tool must be a member of the `fnadmin` group.
- This tool must be run on the storage library server containing the document locator table that has the references that are being removed.
- This tool cannot be run while Image Services is down. The tool requires Document Services to be running.
- For multiple storage library server configurations, remote surface must be read and write disabled. In other words, the user must



manually read/write disable the surface(s) not on the server with the doc locator table. However, the tool will automatically read/write disable the surface(s) on the server with the doc locator table.

- For two-sided optical disks, only one side is removed when using the -s option.
- For two-sided optical disks, when one side is specified, the other side is not removed unless it is explicitly specified.
- The -s option is used to remove a single surface and the -f option is used to remove multiple surface for best performance.
- There is no password required to run this tool.
- This tool should typically be run before running the **remove\_surf** tool.
- The tool performs a sequential scan of the doc locator table and this scan could take a while to complete if the table is large.
- If the tool is interrupted, statistics in the surf\_info table might not be correct. The user must rerun the tool to fix any inconsistencies in the database.
- This tool does not create a checkpoint file. If the tool is interrupted, it must be restarted from the beginning.
- Upon completion, a log file is generated.

## Procedures

### Phases of remove\_docs\_ref

The remove\_docs\_ref tool's program goes through a few distinct phases as it runs:

- 1** In the **initiation phase**, the tool does the following: calls `fnc_register_program`, links to the necessary abstracts, checks to see if the user is in the `fnadmin` group via `SEC_validate_logon()`, logs on to services (SEC, DOC, INX) so that `remove_docs_ref` runs as a service process, logs on to databases (document locator and Permanent MKF DB), verifies that the tool is run on the server with the document locator table, process the input surface ID(s), verifies whether the input surface ID(s) are read and write disabled. If there is any error, the program will terminate.
- 2** In the **search the document locator table phase**, the tool does the following: logs that the `remove_docs_ref` tool was invoked, searches the docs table sequentially by starting with the lowest document ID, removes any references that point to a surface in the removal list found in the document ID list just searched, removes both the Index Services Doctaba entry and the table entry if both `surface_id_1` and `surface_id_2` in the docs table do not have any associated surfaces, reduces the total of the `num_act_docs` field in the `surf_info` table to reflect the removal of active documents for a particular surface (calls the `ODT_update_remote` entry point inside the RMK transaction), displays dots (. . .) to indicate the progress of documents processed (a dot = every 1000 documents).
- 3** In the **final phase**, the tool does the following: sets the `num_act_docs` field in all of the associated `surf_info` entries to zero (0) upon the successful completion of `remove_docs_ref` (calls the `ODT_update_remote` entry point inside the RMK transaction), generates a system log of surface ID(s) to indicate the surfaces that have been processed, logs off from the various databases and services, unlinks from the abstracts.

## How to form a list of optical tranlog surfaces

The following steps can be used to form a list of optical tranlog surfaces on a storage library server. This optical tranlog list can then be used as input to the remove\_docs\_ref program.

**Note** If you have a multiple storage library server configuration, you will need to perform these steps on each storage library server. For more information on using MKF\_tool, see [“MKF\\_tool” on page 801](#).

- 1 Invoke the MKF\_tool and run the following MKF\_tool command to create a join between the surf\_stat\_info table and the family\_disk table with the family\_id field.

```
<MKF_tool>join surf_stat_info to family_disk as surf_family_join by family_id
```

- 2 In the same MKF\_tool session, run the MKF\_tool hardcopy command to generate a file that will contain the output from the query command in the next step.

```
<MKF_tool>hardcopy tranlog_surf
```

- 3 Next, generate a complete list of tranlog surfaces (optical and MSAR tranlog surfaces):

```
<MKF_tool>select surf_family_join * where is_primary=0 showing surface_id
```

- 4 Exit from the MKF\_tool program. This will close the tranlog\_surf file opened in Step 2, which now contains a concatenated list of all both MSAR and optical tranlog surfaces.

```
<MKF_tool>quit
```

- 5 Using your preferred editor, cleanup the tranlog\_surf file so that only surface\_ids are in the file.

**Note** Steps 6 - 9 must be done to remove MSAR tranlog surfaces from the inclusive list of tranlog surfaces generated in the previous steps.

- 6 Invoke the MKF\_tool and run the MKF\_tool hardcopy command to generate a file for all the MSAR surfaces.

```
<MKF_tool>hardcopy msar_surf
```

- 7 Next, run a query to list all of the MSAR surface IDs on the storage library server.

```
<MKF_tool>select lib_surfaces * showing surface_id
```

- 8 Exit from the MKF\_tool program. This will close the msar\_surf file opened in Step 6, which now contains a concatenated list of all both MSAR and optical tranlog surfaces.

```
<MKF_tool>quit
```

- 9 Using your preferred editor, cleanup the msar\_surface file so that only surface\_ids are in the file.

---

**Note** If you are running Image Services on a UNIX server, complete Steps 10 - 12 to form a list of optical-only tranlog surfaces. If you are running Image Services on a Windows Server system, you will need to use Windows native tools or third-party utility programs (e.g., Hamilton CSH) to do what is described here.

---

**10** Using the UNIX sort utility, sort the msar\_surf file.

```
sort msar_surf>sorted_msar_surf
```

**11** Then sort the tranlog\_surf file.

```
sort tranlog_surf>sorted_tranlog_surf
```

**12** Finally, using the diff command, remove any MSAR surfaces from your tranlog surfaces list.

```
diff sorted_msar_surf sorted_tranlog_surf | grep ">" | sed -e "s/>/">optical_
tranlog_surf
```

You now have a complete list of optical tranlog surfaces in the optical\_tranlog\_surf file. You can now use this file as an input file in the remove\_docs\_ref program when used with the **-f** option.

## How to remove all optical tranlogs

The following steps can be used to remove all of the optical tranlogs in a system.

- 1 Identify all of the optical tranlogs on all of the servers on the system. To do this, refer to [“How to form a list of optical tranlog surfaces” on page 1091](#).
  - 2 Next, on each server that is NOT the doc locator server, use the Storage Library Control program to read/write disable every surface on the list. To do this, see the “Storage Library Control” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).
  - 3 On every server, use the Storage Library Control program to eject every surface in the list. (Be sure to remove each surface from the I/O station.) To do this, see the “Storage Library Control” chapter of your *System Administrator’s Handbook*.
  - 4 Run `remove_docs_ref -f` using the input file from Step 1.
- `remove_docs_ref -f docid.txt` (example)
- 5 Run `remove_surf` on each surface in the list on each server. For more information, refer to [“remove\\_surf” on page 1095](#).

`remove_surf 3002` (example)

## Related Topics

[“remove\\_surf” on page 1095](#)

[“MKF\\_tool” on page 801](#)

## remove\_surf

### Description

This tool removes the specified surface from the surf\_info Permanent MKF DB tables, family\_disk write surfaces list (current and future write surfaces), Permanent MKF DB table, surf\_locator Permanent MKF DB table, lib\_surfaces Permanent MKF DB table, OSA shared memory, SRF shared memory and OSA checkpoint file (the tables other than the document locator table). For two-sided optical disks, both sides will automatically be removed.

### Use

Prior to the development of this tool and the [remove\\_docs\\_ref](#) tool, Image Services did not provide a way to remove all references to a surface without erasing the surface, nor was there a way to remove a group of surfaces from the database in a safe and efficient manner.

An example of when these tools might be used is if you were converting your primary optical surfaces to MSAR surfaces and then you want to remove all references to the tranlog optical surfaces. This tool can be used to change an assigned blank surface (unlabeled surface - it has an associated surface ID but it has not been swapped into a drive and no volume label has been written.) to an unassigned blank surface.

The remove\_surf command line tool is run with a surface ID (local surface ID) as its only parameter and it is a required parameter.

This tool is typically run after the remove\_docs\_ref tool, but can be invoked without running remove\_docs\_ref for a surface ID that has no associated references in the document locator table. This would be

the case if the surface is unlabeled and has no documents or if the surface was copied or consolidated.

## Syntax

remove\_docs\_ref <surface ID>

**<surface ID>** Removes any reference in the surf\_docs table to the specified surface ID on this system.

---

**Note** If the tool is run with no parameters, the usage statement will display.

---

## Sample Output

The following sample is output to the log file when remove\_surf is run.

```
corona(root)/> remove_surf 3002
Successfully removed surface id=3002 from table(s): surf_info surf_locator
Successfully removed surface id=3003 from table(s): surf_info

Program successfully terminated
```

## Checklist

Before you use remove\_surf, be aware of the following:

- The user running the tool must be a member of the fnadmin group.
- This tool must be run on the storage library server containing the surf\_info (ODT) entry for the specified surface.
- This tool cannot be run while Image Services is down. The tool requires Document Services to be running.



- The specified written surface must be out of the library.

---

**Note** An unlabeled, blank surface does not need to be removed from the library.

---

- The num\_act\_docs field of surf\_info must be set to zero (0). For two-sided surfaces, both sides must have no active documents (num\_act\_docs field must equal zero for both sides). If there are no references, (surface\_id\_1 or surface\_id\_2 fields) in the docs table for a given surface, the num\_act\_docs field will be set to zero. A surface that does not have any documents will also have a zero (0) value in the num\_acts\_docs field. The **remove\_docs\_ref** tool will set the num\_act\_docs field to zero (0) for a surface that has successfully processed.
  - In addition, the **deldocs** tool will set the num\_acts\_docs field to zero (0) after all the documents of a surface have been deleted from the docs and doctaba tables.
  - The **doccnt** tool can also be used in the event that the num\_act\_docs field does not accurately represent the number of active documents. The **doccnt** tool does a sequential search through the entire docs table to make the num\_act\_docs field accurate. This could take some time depending upon the size of your docs table. **doccnt** takes a single surface or a list of surfaces as input parameters.

---

**Note** Running **doccnt** will take the same amount of time as remove\_docs\_ref, so if you are not sure there of documents in the docs table to remove or if the statistics are correct, running remove\_docs\_ref before running remove\_surf will be sufficient to fix the statistics so remove\_surf can be run successfully.

---

- The specified surface must not have any associated pending read/write/background job requests.
- For two-sided optical disks, the remove\_surf tool will remove both sides (Side A and B). Either side can be used as an input parameter.
- There is no password required to run this tool.
- This tool should typically be run after running the **remove\_docs\_ref** tool.

## Procedure

The remove\_surf tool's program goes through a few distinct phases as it runs:

- 1** In the **initiation phase**, the tool does the following: calls fnc\_register\_program, links to the necessary abstracts, checks to see if the user is in the fnadmin group via SEC\_validate\_logon(), does a service logon (SEC\_service\_logon) so that remove\_surf runs as a service process.
- 2** In the **verification phase**, where the surface needs to meet the removal criteria (If any criteria is not met, the tool will terminate.), the tool does the following: checks to see if the surf\_info tables know about this surface (call ODT), verifies that the surface is not currently in the box or is unlabeled, verifies that num\_act\_docs is equal to zero (0) (If num\_act\_docs is not set to zero, the tool will indicate to the user that remove\_docs\_ref needs to be run before remove\_surf.), disables the surface for reads and writes (call HLT), verify there are no pending requests (call SRF to check the total\_count and odu\_total fields).
- 3** In the **remove DB references phase**, the tool does the following: removes references in the surf\_info, surf\_locator, and lib\_surfaces tables

from the permanent DB entries. (Normally, removal of all database entries can be done with a single, permanent DB MKF transaction. There is one exception, and that is when removing a surface on a storage library server that does not have the document locator table local on the server. The surf\_locator table, used for multiple storage library server configurations, is always located on the same server as the document locator table. In this case, two transactions are needed, one remote and one local.)

- 4 In the **final phase**, the tool does the following: generates a system log to indicate that the surface has been successfully removed from the surf\_info tables, lib\_surfaces table, etc., logs off from SEC services, unlinks from the abstracts.

## Related Topics

[“remove\\_docs\\_ref” on page 1084](#)

[“deldocs” on page 282](#)

[“doccnt” on page 310](#)

---

## rollcall

**AIX**

The rollcall tool is supported in Image Services for AIX/6000 systems only.

### Description

The rollcall tool broadcasts a request over a network for a specific system or all active systems to identify themselves. For a specific system request, you must specify the network number of the system. The rollcall tool stores the specified network number in the net number field of the net address it uses for broadcasting.

Output from rollcall lists the names of systems that respond, sorted alphabetically. If the output device is a tty, rollcall prints the list in columns. You can specify that the addresses of each responding system also be included in the output.

### Use

Use rollcall to list the systems that are running on the network.

### Syntax

```
rollcall [ -vn# ]
```

- v** Prints the network address of responding systems
- n#** Stores the specified network number # in the net number field of the rollcall net address

---

**Note** You can specify only one rollcall option. If you want to use both **-v** and **-n** options, specify them together in the format **-vn#**.

---

## Sample Output

The sample below is output from rollcall when run with no options specified.

```
corona (root) /> rollcall
BOSTON_DEMO      alpha          costa2         hawk           smsdev
CSO_FAX          amber         costa2APP      hydra          snads
CTnet5           anza          costa3         int66          sqArabic
```

The following sample is output from rollcall -v and displays the network addresses of each responding system.

```
corona (root) /fns/bin> rollcall -v
00000001.0800340010ce tyche
00000001.080034002515 astro
00000001.080034001b59 int66
00000001.080034002530 sql
000002fa.080034002494 wimble
00002001.080034002494 wimble
00000001.0800340013b2 snads
00000001.080034001f5d magic
00000001.080034002478 amber
```

The last sample is output from `rollcall -v` and specifying a network number:

```
corona (root) /fnsw/bin> rollcall -vn1
00000001.080034002250 sqArabic
00000001.080034002468 alpha
00000001.080034001b56 sq4
00000001.08003400237c venice
00000001.02608c2f605e sunset
00000001.02608c2f605b atlas
00000001.02608c2fbdd2 verde
00000001.02608c2f6717 swat
00000001.02608c2f5ccc elvis
00000001.02608c2cb07f costaNLS
00000001.08005a4d09f9 sparta
```

## Checklist

Before you use `rollcall`, be aware of the following:

- Since `rollcall` only takes one option, you must use both `-v` and `-n` together in the format `-vn#`.
- The `rollcall` tool is supported in Image Services for AIX/6000 systems only.

## Procedure

No specific procedure is required.

## Related Topics

[\*\*“gcp” on page 498\*\*](#)

## RSEC\_imp

### Description

Image Services provides a single user logon from a trusted Windows Server (NT) domain. This functionality uses FileNet-provided tools to first export the users names and group names from the Windows Server domain and then import them into the Image Services (IS) Security Service.

The step you need to complete before you use the RSEC\_imp tool is exporting the Windows Server domain security information to an XML formatted file. For more information on this file, go to [“XML-Formatted File Information” on page 1110](#). The tool used to perform this function is ntdm\_exp. To review the ntdm\_exp tool go to [“ntdm\\_exp” on page 881](#).

### Use

It is the use of the RSEC\_imp tool in conjunction with the ntdm\_exp tool that makes the single logon from a trusted Windows Server domain feature possible.

Run RSEC\_imp to import the Windows Server domain security information into the IS Security Service. This requires the use of an XML formatted file generated from using the export tool mentioned above and must be run on a Windows Server machine with no Image Services software present, but that has either Web Service or IDM Desktop installed. This tool can import the XML file information to any other IS server (Windows Server or UNIX) on the network. There are two separate versions of the RSEC\_imp command that can possibly be run. The first is importing the XML input file, which is mandatory.

The second has to do with the various renaming options which could need to be done depending on your IS system configuration.

---

**Note** You must be logged in as the SysAdmin user to run the RSEC\_imp tool.

---

## Syntax

The command for importing the XML Input file:

```
RSEC_imp /H<host> /I<file>
```

The command for renaming domain name:

```
RSEC_imp /H<host> /R:[<old_name>]:<new_name> [/E<file>] [/F] [/O]
```

**/?** Help

**/H<is\_domain>** Image Services (IS) domain name

**/R:[<old\_name>]:<new\_name>** Rename existing IS user/group to the <new\_name> domain name. If <old\_name> is omitted, (e.g., RSEC\_imp/R::<new\_name>), all the users, except systems accounts, are prepended with "<new\_name>\" (/E, /F, /O).

EXAMPLE 1:

If you have the following IS objects:

```
MOOREA\USER1
MOOREA\USER2
MOOREA\USER3
MOOREA\USER4
```



MOOREA\USER5  
MOOREA\USER6

Then you run /R:MOOREA:GOOFY as part of the RSEC\_imp command, the above object are renamed to the following (with new hashed passwords):

GOOFY\USER1  
GOOFY\USER2  
GOOFY\USER3  
GOOFY\USER4  
GOOFY\USER5  
GOOFY\USER6

#### EXAMPLE 2:

If you have the following IS objects:

USER1  
USER2  
USER3  
USER4  
USER5  
USER6

Then you run /R::GOOFY as part of the RSEC\_imp command (no old\_name), the above objects are renamed to the following (with new hashed passwords):

GOOFY\USER1  
GOOFY\USER2  
GOOFY\USER3  
GOOFY\USER4

---

```
GOOFY\USER5
GOOFY\USER6
```

**/E<file>** (use w/ /R) Specify an exclude users file. This is a comma delimited ASCII file where you place one user per line, followed by a comma (For example., MOOREA\USER3,). This one user will not be renamed after the command is executed.

If you have the following IS objects:

```
MOOREA\USER1
MOOREA\USER2
MOOREA\USER3
MOOREA\USER4
MOOREA\USER5
MOOREA\USER6
```

And you have an exclude file name exclude.lst with one line,

```
MOOREA\USER3,
```

Then you run /R:MOOREA:GOOFY /Eexclude.lst as part of the RSEC\_imp command, the above object are renamed to the following (with new hashed passwords):

```
GOOFY\USER1
GOOFY\USER2
MOOREA\USER3
GOOFY\USER4
GOOFY\USER5
GOOFY\USER6
```

Notice that USER3 didn't change. Also, user exclude files are case sensitive since Image Services is case sensitive.

- 
- /F<file> (use w/ /R)** Force removing of users during rename operation if the user already exists as 'new\_name\user'.
- /O<file> (use w/ /R)** Override system accounts. During a rename operation, by default, system accounts are skipped. If the user wants to also rename system accounts, they use the /O option. This option should be used with care.

During an Image Service installation, the following seven accounts are created:

3 system user accounts

FieldService  
Operator  
SysAdmin

4 system group accounts

SysAdminG  
AuditG  
FieldServiceG  
OperatorG

- /I<file>** XML input file (XML formatted file created in import step).

## Procedure

- 1 Log on to the Windows Server client to which you plan to import the user and group information. You need to log on as SysAdmin.
- 2 Download the RSEC\_imp tool from the Web to the \fns\CLIENT\bin directory. The tool is located on the IBM Information Support Web site ([www.ibm.com/software/data/support](http://www.ibm.com/software/data/support)).
- 3 Go to the directory where your XML formatted file was created, for example, \fns\CLIENT\tmp.
- 4 Import the Windows Server domain security information into the IS Security Service. This step take the XML formatted input file and update the Image Services security tables. Review the RSEC\_imp options **before** running the import command.

When ready, use the following command to import the groups and users to the IS system using the XML formatted file created during the ntdm\_exp:

```
RSEC_imp /H<is_domain> /I<file>
```

Example:

```
RSEC_imp /Hidm1 /Ixml.dat
```

When you enter the command, you are prompted for user name and password with the necessary privileges to complete the import (e.g., SysAdmin). You then see messages indicating you are importing an Windows Server domain input file and security information.

- 5 Use the following command to rename (if desired):

```
RSEC_imp /H<is_domain> /R:[<old_name>]:<new_name> [/E<file>] [/F] [/O]
```

Example:

```
RSEC_imp /Hidm1 /R:oldname:newname /Eexclude.lst /F
```

```
RSEC_imp /Hidm1 /R::newname /Eexclude.lst /O
```

---

**Note** When the old domain name is missing as in the second example above, existing users are converted to unified logon users.

The /R option should never be used in the same command as the /I option. They are mutually exclusive.

---

**Note** All error messages are logged to a file called ntdm\_exp.log in the current directory.

To check if the import was successful, run Xapex.exe, log in as Administrator, and check to see if the new groups and users are present. For example, if you imported a group called **support** from domain **nt55** and you know user **johnf** belongs to the group, then after a successful import a new group (**NT55/SUPPORT**) and a new user (**NT55/JOHNF**) will have been created.

---

---

## XML-Formatted File Information

This section describes the XML specification for RSEC\_imp. RSEC\_imp is a IS Toolkit-based import utility which takes an XML-formatted input file and imports it into Image Services. RSEC\_imp support the following XML elements:

**<domains>** Global document element.

This is the beginning of the document indicator.

Subelements:

- <domain>: Required. See <domain> below.
- <group>: See <group> below.

**<domain>** Domain element.

This is the exported Windows Server group name.

Subelements:

- <dtype>: Required. Windows Server domain name.

**<group>** Group element.

this is the exported Windows Server group indicator.

Subelements:

- <gname>: Required. Exported Windows Server group Name.
- <gcomment>: Optional. Group comments.
- <op>: Required. This is the group operator. The following operations are supported:

**add** - Add group

**expire** - expire group

**sync\_kill** - sync group/kill users if applicable

**sync\_expire** - sync group/expire uses if applicable

**Sync\_noop** - sync group/no\_op on users

**delete\_kill** - delete group/kill users if applicable

**delete\_expire** -delete group/expire uses if applicable

**delete\_noop** - delete group/no\_op on users

**noop** - no\_op on users

- **<user>**: User indicator. The **<user>** element should have the following subelements:

**<uname>**: Required. User name

**<ucomment>**: Optional. User comments.

**<fullname>**: Optional. User full name.

## XML File General Concepts

- `<domains>` tag should be the first element after the XML declaration tag and `</domains>` tag should be the last element of the XML document.
- This is followed by one or more `<domain>` tag - `</domain>` tag pairings.
- Within each `<domain>` tag, groups are listed in linear order and the nested groups are not supported.
- Within each group tag `<group>`, there is a single group name element tag `<gname>`, a single group operator tag `<op>`, a single group comment tag `<gcomment>`, and 0 or more user element tags `<user>`.
- Within each user tag `<user>`, there is a single user name tag `<uname>`, a single user comment tag `<ucomment>`, and a single user full name tag `<fullname>`. Depending on the group operator you choose, some group operators to not require user tags.



---

## XML File General Form

---

<code>&lt;?xml version="1.0"?</code>	XML declaration
<code>&lt;domains&gt;</code>	document begins
<code>&lt;domain&gt;&lt;dname&gt;...&lt;/dname&gt;</code>	domain name
<code>&lt;group&gt;</code>	group begins
<code>&lt;gname&gt;...&lt;/gname&gt;</code>	group name
<code>&lt;op&gt;...&lt;/op&gt;</code>	group operator
<code>&lt;gcomment&gt;...&lt;/gcomment&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user1
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user2
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user3
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user4
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user5
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>.</code>	
<code>.</code>	add more users here
<code>.</code>	
<code>&lt;/group&gt;</code>	group ends
<code>.</code>	
<code>.</code>	add more groups here
<code>.</code>	
<code>&lt;/domain&gt;</code>	domain ends
<code>&lt;domain&gt;&lt;dname&gt;...&lt;/dname&gt;</code>	
<code>.</code>	
<code>.</code>	
<code>&lt;/domain&gt;</code>	
<code>.</code>	add more domains
<code>.</code>	
<code>&lt;/domains&gt;</code>	document ends

---

---

## XML File Example Form

Here is an example that contains group **HR** with **add** operator and group **ENG** with **expire** operator of domain **Windows**.

---

```
<?xml version="1.0"?  
  
<domains>  
<domain><dtype>Windows</dtype>  
  
<group><gname>HR</gname>  
<op>add</op>  
<gcomment>Human Resource Group</gcomment>  
<user><uname>bob</uname><fullname>Bob Jones</fullname>  
<ucomment>Developer</ucomment></user>  
<user><uname>steve</uname><fullname>Steve Jobs</fullname>  
<ucomment>CEO</ucomment></user>  
<user><uname>lucy</uname><fullname>Lucy Ball</fullname>  
<ucomment>Secretary</ucomment></user>  
</group>  
  
<group><gname>ENG</gname>  
<op>expire</op>  
<gcomment>Engineering</gcomment>  
</group>  
  
</domain>  
</domains>
```

---

## Related Topics

[“ntdm\\_exp” on page 881](#) and [“SEC\\_imp” on page 1129](#).

---

**Note** Portions of this software were developed using the “expat XML parser library”. The contents of this file are subject to the **Mozilla Public License Version 1.1** (the "License"); you cannot use this file except in compliance with the License. You can obtain a copy of the License at <http://www.mozilla.org/MPL/>.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either expressed or implied. See the License for the specific language governing rights and limitations under the License.

---

# SC\_convert

## Description

This program converts a bi-level image from one standard format to another by calling the SC data abstract. This program produces an ASCII output file to report completion status named './SC\_convert.out'.

## Use

Keywords and values can be abbreviated by typing only the first few initial characters of the keyword or value. In many cases, you only need to type the first letter. However, if new parameters are implemented in the future, the minimum abbreviations could change. Therefore, shell scripts running this program should not abbreviate.

The keywords and values are usually case insensitive. Alternatives bracketed by curly braces and separated by commas are syntactic notation for 'choose exactly one alternative.' Items enclosed in angle brackets are meta-variables with mnemonic names to indicate what they stand for.

## Syntax

```
SC_convert [<options>]
```

All command line options are optional, except as follows:

'if' and 'of' are required for DOS.

## Options

The command line parameters are mostly of the form:

---

<keyword>=<value>

---

**Note** There must be NO SPACE before or after the equal sign (=).

---

**{?,help}** Creates a file named SC\_convert.out containing the menu of options. Entering invalid parameters also produces this help file.

---

**Tip** To display the list of options for this tool directly to your computer screen, enter SC\_convert -h. To save the list of options on your hard drive, enter the SEC\_debug command without any parameters. SC\_convert creates a file named SC\_convert.out in the current directory that you can view using the **more** or **less** commands.

---

**{if,inputfile}** Input image filename. Default = stdin. Required option on DOS. For example:

if=<filename>

**{of,outputfile}** Output image filename. Default = stdout. Required option on DOS. For example:

of=<filename>

**format** Output image format could be 'FileNet', 'TIFF', or 'CALs'. For example:

f={FileNet,TIFF,CALS}

Default is TIFF. Input image format is recognized automatically, and so it can not be specified. The formats can be abbreviated f=FileNet, t=TIFF, and c=CALS.

---

**compression** Compression type of output image: either CCITT Recommendation T.4 (Group 3 FAX) or CCITT Recommendation T.6 (Group 4 FAX). Default is group4. For example:

```
c={g3,group3,g4,group4}
```

**tiled** Default: When this parameter is omitted, the output image will be tiled if the uncompressed image size exceeds a certain empirically determined threshold. Otherwise, the output image will not be tiled. For example:

```
t={{TRUE,FALSE},{1,0},{yes,no}}
```

TRUE, 1, yes → output image will be tiled.

FALSE, 0, no → output image will not be tiled.

This is recommended in order to make the best overall performance-space tradeoff.

**rotate** Put a tag in the output image that says to rotate the image counter-clockwise on display by the specified number of degrees. For example:

```
r={90,180,270}
```

(Do NOT rotate it now as part of the conversion.)

## Examples

**Example 1 (UNIX-like and 'Windows+DOS'-like platforms):**

```
SC_convert if=imagein.fn of=imageout.tiff f=t c=g4
```

This example converts an image in a file called 'imagein.fn' into a group 4 TIFF image in a new file called 'imageout.tiff'. The output image will be untitled unless it exceeds a reasonable upper size limit in which case the output image will be tiled.

**Example 2 (UNIX-like platforms):**

```
SC_convert <imagein.fn>imageout.tiff
```

This example has the same effect as Example 1. This is possible because SC reads the input file sequentially and writes the output file sequentially.

**Example 3 (UNIX-like platforms):**

```
cat imagein.fn | SC_convert | cat >imageout.tiff
```

This example has the same effect as Example 1 and Example 2.

## SEC\_debug

### Description

The SEC\_debug tool enables support personnel to trace execution within the Security Services subsystem of the Image Services server. This tool provides different trace levels based on specified command options.

FileNet software defines classes of functions within shared libraries (also known as “abstracts”). Each function, called an “entrypoint,” can be a member of exactly zero or one of the following classes: entry, ventry, reentry, sentry, and ientry. If it is not a member of any of these classes, the function is referred to within the SEC\_debug program as a “uentry” for “unexposed entry.”

You can use the SEC\_debug tool to enable or disable trace debugging for each entrypoint class within a given shared library or set of shared libraries. The debugging system for security contains three shared libraries (SEC, SECI, and SECr) and one executable program (SECs). For example, you can turn on or off debugging for entries in SEC or sentries and reentries in SECI. This enables you to minimize the number of debugging statements logged to solve a specific problem, thus maximizing system performance, while maintaining maximum possible debugging coverage.

Security Service modules contain built-in debugging messages, sent to the standard system log.



## Use

Use SEC\_debug from the command line while the Image Services server is running to activate or deactivate debugging for a set of entry-point classes for the SEC, SECI, SECr shared libraries and the SECs executable. You can also use it to view the debugging status at any time.

---

**Note** Do not use the SEC\_debug tool unless explicitly instructed to do so by your service representative.

---

The SEC\_debug tool terminates immediately after changing the debugging levels. The Image Services software maintains debugging status. When enabled through SEC\_debug, debugging remains enabled until disabled using this tool or by shutting down and restarting the Image Services server.

## Syntax

```
SEC_debug [+<type>] [-<type>] [STATUS]
```

---

**Tip** To display a help screen for this tool, enter the SEC\_debug command without any parameters.

---

**+<type>** Turns on debugging for the specified shared library or executable and entrypoint type.

<type> is one of the following:

SEC_ENTRY	A shared library (abstract) and an entrypoint classification.
SEC_SENTRY	
SEC_VENTRY	
SEC_RENTRY	
SEC_IENTRY	
SEC_UENTRY	
SECI_ENTRY	
SECI_SENTRY	
SECI_VENTRY	
SECI_RENTRY	
SECI_IENTRY	
SECI_UENTRY	
SECr_ENTRY	
SECr_SENTRY	
SECr_VENTRY	
SECr_RENTRY	
SECr_IENTRY	
SECr_UENTRY	
SECs_ENTRY	
SECs_SENTRY	
SECs_VENTRY	
SECs_RENTRY	
SECs_IENTRY	
SECs_UENTRY	

SEC_ALL	Shortcut to a group of entrypoint classifications.
SECI_ALL	
SECr_ALL	
SECs_ALL	
ALL	Shortcut to all shared libraries and entrypoint classifications.

For details, see [“Types” on page 1124](#).

**-<type>** Turns off debugging for the specified shared library or executable and entrypoint type.

---

<type> is one of the following:

SEC_ENTRY	A shared library (abstract) and an entrypoint classification.
SEC_SENTRY	
SEC_VENTRY	
SEC_RENTRY	
SEC_IENTRY	
SEC_UENTRY	
SECI_ENTRY	
SECI_SENTRY	
SECI_VENTRY	
SECI_RENTRY	
SECI_IENTRY	
SECI_UENTRY	
SECr_ENTRY	
SECr_SENTRY	
SECr_VENTRY	
SECr_RENTRY	
SECr_IENTRY	
SECr_UENTRY	
SECs_ENTRY	
SECs_SENTRY	
SECs_VENTRY	
SECs_RENTRY	
SECs_IENTRY	
SECs_UENTRY	
SEC_ALL	Shortcut to a group of entrypoint classifications.
SECI_ALL	
SECr_ALL	
SECs_ALL	
ALL	Shortcut to all shared libraries and entrypoint classifications.

For details, see [“Types” on page 1124](#).

---

**STATUS** Prints a table to standard output of each debugging type and its status. It also displays a status table if a command changed any debug settings whether you added the STATUS parameter or not.

## Types

Types let you enable or disable debugging for all entrypoint types within a particular shared library at the same time. The final type, "ALL," affects debugging for all shared libraries and all entrypoints at once.

You can combine types (using a plus or minus sign) on the command line by separating them with a space to produce the desired effects.

For example, to enable debugging for sentry and reentry functions within the SEC shared library and disable debugging for ventry functions in the SECr shared library with one command, type the following:

```
SEC_debug +SEC_SENTRY +SEC_RENTRY -SECr_VENTRY
```

The order of parameters on the command line is not significant.

If the command enables a debugging type that is already enabled or disables a type that is already disabled, the command has no effect on that debugging type. This is not an error.

However, if you enable and disable a single debugging type with the same command, the command generates an error. For example:

```
SEC_debug +SEC_SENTRY -SEC_SENTRY
```

This combination is invalid because it is ambiguous whether the program should enable or disable SEC\_SENTRY debugging.

## SEC\_debug +ALL -SEC\_RENTRY

This combination is invalid because it is ambiguous whether the program should enable or disable SEC\_RENTRY debugging.

The SEC\_debug tool rejects and reports an invalid command due to ambiguity.

## Sample Output

The following examples show the output that displays after running the SEC\_debug command using various options.

The following sample shows output after entering the following command when all debugging is turned off.

### SEC\_debug STATUS

```
SEC_debug:  Manages SEC Enhanced Runtime Debugging
```

```
-----  
All debugging is currently off.
```

The following sample shows output after entering the following command when SEC\_sentry debugging is turned on.

## SEC\_debug STATUS

```

SEC_debug:  Manages SEC Enhanced Runtime Debugging
-----

Debugging Status:
-----
| Abst | Entry | Ventry | Rentry | Sentry | Ientry | Uentry |
-----
| SEC  | OFF  | OFF   | OFF   | ON    | OFF   | OFF   |
-----
| SECs | OFF  | OFF   | OFF   | OFF   | OFF   | OFF   |
-----
| SEC1 | OFF  | OFF   | OFF   | OFF   | OFF   | OFF   |
-----
| SECr | OFF  | OFF   | OFF   | OFF   | OFF   | OFF   |
-----

```

The following sample shows output after entering the following command when no debugging is turned on.

```
SEC_debug +SEC_IENTRY +SECs_ALL
```

```
SEC_debug:  Manages SEC Enhanced Runtime Debugging
```

```
-----  
Settings changed.
```

```
Debugging Status:
```

```
-----  
| Abst | Entry | Ventry | Rentry | Sentry | Ientry | Uentry |  
-----  
| SEC  | OFF  | OFF   | OFF   | OFF   | ON    | OFF   |  
-----  
| SECs | ON   | ON    | ON    | ON    | ON    | ON    |  
-----  
| SEC1 | OFF  | OFF   | OFF   | OFF   | OFF   | OFF   |  
-----  
| SECr | OFF  | OFF   | OFF   | OFF   | OFF   | OFF   |  
-----
```

## Checklist

Before you use SEC\_debug, be aware of the following:

- You should use this tool only under the explicit direction of your service representative.
- Since SEC\_debug runs while the Image Services server is running, it must share memory resources to make debugging calls to the software. Furthermore, since each security operation can make tens or hundreds of function calls, the system log fills quickly, causing severe performance degradation. Use debugging sparingly and with extreme caution.

## Procedure

Run this program from the command line. See the [“Syntax” on page 1121](#) for details on command line options.

## Related Topics

[“WALSEC\\_debug” on page 1300](#)

See the “Security Administration” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).



## SEC\_imp

### Description

Image Services provides a single user logon from a trusted Windows Server domain. This functionality uses FileNet-provided tools to first export the users names and group names from the Windows Server domain and then import them into the Image Services (IS) Security Service.

The step you need to complete before you use the SEC\_imp tool is exporting the Windows Server domain security information to an XML formatted file. For more information on this file, go to [“XML-Formatted File Information” on page 1136](#). The tool used to perform this function is ntdm\_exp. To review the ntdm\_exp tool go to [“ntdm\\_exp” on page 881](#).

### Use

It is the use of the SEC\_imp tool in conjunction with the ntdm\_exp tool that makes the single logon from a trusted Windows Server domain feature possible.

Run Sec\_imp to import the Windows Server domain security information into the IS Security Service. This step requires the use of an XML formatted file generated from using the export tool mentioned above and must be run on an IS server (Windows Server or UNIX). This tool can import the XML file information to any local IS server (Windows Server or UNIX) and it can import to an IS server in its own domain (Windows Server or UNIX). There are two separate versions of the SEC\_imp command that can possibly be run. The first is importing the XML input file, which is mandatory. The second has to do with the var-

ious renaming options which might need to be done depending on your IS system configuration.

---

**Note** You must be logged in as the SysAdmin user to run the SEC\_imp tool.

---

## Syntax

The command for importing the XML Input file:

```
SEC_imp /H<host> /I<file>
```

The command for renaming domain name:

```
SEC_imp /H<host> /R:[<old_name>]:<new_name> [/E<file>] [/F] [/O]
```

**/?** Help

**/H<is\_domain>** Image Services (IS) domain name

**/R:[<old\_name>]:<new\_name>** Rename existing IS user/group to the <new\_name> domain name. If <old\_name> is omitted, (e.g., SEC\_imp/R::<new\_name>), all the users, except systems accounts, are prepended with "<new\_name>\\" (/E, /F, /O).

EXAMPLE 1:

If you have the following IS objects:

```
MOOREA\USER1  
MOOREA\USER2  
MOOREA\USER3  
MOOREA\USER4
```

MOOREA\USER5  
MOOREA\USER6

Then you run /R:MOOREA:GOOFY as part of the SEC\_imp command, the above object are renamed to the following (with new hashed passwords):

GOOFY\USER1  
GOOFY\USER2  
GOOFY\USER3  
GOOFY\USER4  
GOOFY\USER5  
GOOFY\USER6

#### EXAMPLE 2:

If you have the following IS objects:

USER1  
USER2  
USER3  
USER4  
USER5  
USER6

Then you run /R::GOOFY as part of the SEC\_imp command (no old\_name), the above objects are renamed to the following (with new hashed passwords):

GOOFY\USER1  
GOOFY\USER2  
GOOFY\USER3  
GOOFY\USER4

```
GOOFY\USER5
GOOFY\USER6
```

**/E<file> (use w/ /R)** Specify an exclude users file. This is a comma delimited ASCII file where you place one user per line, followed by a comma (For example., MOOREA\USER3,). This one user will not be renamed after the command is executed.

If you have the following IS objects:

```
MOOREA\USER1
MOOREA\USER2
MOOREA\USER3
MOOREA\USER4
MOOREA\USER5
MOOREA\USER6
```

And you have an exclude file name exclude.lst with one line,

```
MOOREA\USER3,
```

Then you run /R:MOOREA:GOOFY /Eexclude.lst as part of the SEC\_imp command, the above object are renamed to the following (with new hashed passwords):

```
GOOFY\USER1
GOOFY\USER2
MOOREA\USER3
GOOFY\USER4
GOOFY\USER5
GOOFY\USER6
```

Notice that USER3 didn't change. Also, user exclude files are case sensitive since Image Services is case sensitive.

- /F<file> (use w/ /R)** Force removing of users during rename operation if the user already exists as 'new\_name\user'.
- /O<file> (use w/ /R)** Override system accounts. During a rename operation, by default, system accounts are skipped. If the user wants to also rename system accounts, they use the /O option. This option should be used with care.

During an Image Service installation, the following seven accounts are created:

3 system user accounts

- FieldService
- Operator
- SysAdmin

4 system group accounts

- SysAdminG
- AuditG
- FieldServiceG
- OperatorG

- /I<file>** XML input file (XML formatted file created in import step).

## Procedure

- 1 Transfer the output from Step 4 of the export procedure ([“ntdm\\_exp” on page 881](#)) to the IS server, for example using FTP. Place the output file in a directory, for example /fnsw/local/tmp (\fnsw\_loc\tmp on Windows Server).
- 2 Log on to the IS server as the SysAdmin user.
- 3 Download the SEC\_imp tool from the Web to the /fnsw/bin directory. The tool is located on the IBM FileNet Web site ([www.ibm.com/software/data/support](http://www.ibm.com/software/data/support)) in the appropriate directory for your operating system:
- 4 Go to the directory on the IS server where you transferred the XML formatted file in Step 1 of this procedure (for example, /fnsw/local/tmp on UNIX, \fnsw\_loc\tmp on Windows Server).
- 5 Import the Windows Server domain security information into the IS Security Service. This step take the XML formatted input file and update the Image Services security tables. Review the SEC\_imp options **before** running the import command.

When ready, use the following command to import the groups and users to the IS system using the XML formatted file created during the ntdm\_exp:

```
SEC_imp /H<is_domain> /l<file>
```

Example:

```
SEC_imp /Hidm1 /lxml.dat
```

When you enter the command, you are prompted for user name and password with the necessary privileges to complete the import (e.g., SysAdmin). You then see messages indicating you are importing an Windows Server domain input file and security information.

- 6 Use the following command to rename (if desired):

```
SEC_imp /H<is_domain> /R:[<old_name>]:<new_name> [/E<file>] [/F] [/O]
```

Examples:

```
SEC_imp /Hidm1 /R:oldname:newname /Eexclude.lst /F
```

```
SEC_imp /Hidm1 /R::newname /Eexclude.lst /O
```

---

**Note** When the old domain name is missing as in the second example above, existing users are converted to unified logon users.

The /R option should never be used in the same command as the /I option. They are mutually exclusive.

---

---

**Note** All error messages are logged to a file called ntdm\_exp.log in the current directory.

To check if the import was successful, run Xapex.exe, log in as Administrator, and check to see if the new groups and users are present. For example, if you imported a group called **support** from domain **nt55** and you know user **johnf** belongs to the group, then after a successful import a new group (**NT55/SUPPORT**) and a new user (**NT55/JOHNF**) will have been created.

---

---

## XML-Formatted File Information

This section describes the XML specification for SEC\_imp. SEC\_imp is a IS Toolkit-based import utility which takes an XML-formatted input file and imports it into Image Services. SEC\_imp support the following XML elements:

**<domains>** Global document element.

This is the beginning of the document indicator.

Subelements:

- <domain>: Required. See <domain> below.
- <group>: See <group> below.

**<domain>** Domain element.

This is the exported Windows Server group name.

Subelements:

- <dtype>: Required. Windows Server domain name.

**<group>** Group element.

this is the exported Windows Server group indicator.

Subelements:

- <gname>: Required. Exported Windows Server group Name.
- <gcomment>: Optional. Group comments.
- <op>: Required. This is the group operator. The following operations are supported:



**add** - Add group

**expire** - expire group

**sync\_kill** - sync group/kill users if applicable

**sync\_expire** - sync group/expire uses if applicable

**Sync\_noop** - sync group/no\_op on users

**delete\_kill** - delete group/kill users if applicable

**delete\_expire** -delete group/expire uses if applicable

**delete\_noop** - delete group/no\_op on users

**noop** - no\_op on users

- **<user>**: User indicator. The **<user>** element should have the following subelements:
  - <uname>**: Required. User name
  - <ucomment>**: Optional. User comments.
  - <fullname>**: Optional. User full name.

## XML File General Concepts

- <domains> tag should be the first element after the XML declaration tag and </domains> tag should be the last element of the XML document.
- This is followed by one or more <domain> tag - </domain> tag pairings.
- Within each <domain> tag, groups are listed in linear order and the nested groups are not supported.
- Within each group tag <group>, there is a single group name element tag <gname>, a single group operator tag <op>, a single group comment tag <gcomment>, and 0 or more user element tags <user>.
- Within each user tag <user>, there is a single user name tag <uname>, a single user comment tag <ucomment>, and a single user full name tag <fullname>. Depending on the group operator you choose, some group operators to not require user tags.

---

## XML File General Form

---

<code>&lt;?xml version="1.0"?</code>	XML declaration
<code>&lt;domains&gt;</code>	document begins
<code>&lt;domain&gt;&lt;dname&gt;...&lt;/dname&gt;</code>	domain name
<code>&lt;group&gt;</code>	group begins
<code>&lt;gname&gt;...&lt;/gname&gt;</code>	group name
<code>&lt;op&gt;...&lt;/op&gt;</code>	group operator
<code>&lt;gcomment&gt;...&lt;/gcomment&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user1
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user2
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user3
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user4
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>&lt;user&gt;&lt;uname&gt;...&lt;/uname&gt;&lt;fullname&gt;...</code>	user5
<code>&lt;/fullname&gt;&lt;ucomment&gt;...&lt;/ucomment&gt;&lt;/user&gt;</code>	
<code>.</code>	
<code>.</code>	add more users here
<code>.</code>	
<code>&lt;/group&gt;</code>	group ends
<code>.</code>	
<code>.</code>	add more groups here
<code>.</code>	
<code>&lt;/domain&gt;</code>	domain ends
<code>&lt;domain&gt;&lt;dname&gt;...&lt;/dname&gt;</code>	
<code>.</code>	
<code>.</code>	
<code>&lt;/domain&gt;</code>	
<code>.</code>	add more domains
<code>.</code>	
<code>&lt;/domains&gt;</code>	document ends

---

---

## XML File Example Form

Here is an example that contains group **HR** with **add** operator and group **ENG** with **expire** operator of domain **Windows**.

---

```
<?xml version="1.0"?  
  
<domains>  
<domain><dname>Windows</dname>  
  
<group><gname>HR</gname>  
<op>add</op>  
<gcomment>Human Resource Group</gcomment>  
<user><uname>bob</uname><fullname>Bob Jones</fullname>  
<ucomment>Developer</ucomment></user>  
<user><uname>steve</uname><fullname>Steve Jobs</fullname>  
<ucomment>CEO</ucomment></user>  
<user><uname>lucy</uname><fullname>Lucy Ball</fullname>  
<ucomment>Secretary</ucomment></user>  
</group>  
  
<group><gname>ENG</gname>  
<op>expire</op>  
<gcomment>Engineering</gcomment>  
</group>  
  
</domain>  
</domains>
```

---

## Related Topics

[“ntdm\\_exp” on page 881](#) and [“RSEC\\_imp” on page 1103](#).

---

**Note** Portions of this software were developed using the “expat XML parser library”. The contents of this file are subject to the **Mozilla Public License Version 1.1** (the "License"); you cannot use this file except in compliance with the License. You can obtain a copy of the License at <http://www.mozilla.org/MPL/>.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either expressed or implied. See the License for the specific language governing rights and limitations under the License.

---

---

## SEC\_init

### Description

The SEC\_init program sets or initializes the reserved objects in the security database.

SEC\_init sets the security system in the following manner:

- Rebuilds the reserved object users and groups (SysAdmin, Operator, and FieldService) and groups (SysAdminG, OperatorG, AuditG, and FieldServiceG) in the security database.
- Resets the system-wide security defaults for users, groups, and devices to their initial settings.

SEC\_init first checks for the existence of the reserved objects. If they do not exist, SEC\_init adds them to the security database. If they exist, SEC\_init overwrites them using initial default settings. In addition, it overwrites the SysAdmin password with the initial system default password for SysAdmin, and resets the Operator, and FieldService passwords. The System Administrator must restore the passwords for Operator and FieldService. See the “Security Administration” chapter of the *System Administrator’s Handbook* for more information. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

As SEC\_init takes each action, the system displays informational messages at the standard output device (see [\*\*“Sample Output” on page 1144.\*\*](#)). SEC\_init affects only reserved objects. It does not change other objects.

## Use

Use SEC\_init to rebuild the security database. If users have forgotten or lost the System Administrator's password, or the System Administrator has changed it, you can reset the password to the default. Since this also resets the other reserved object passwords, the System Administrator must change them again through the Security Administration application after SEC\_init -r completes.

---

**Note** If you need to initialize the security database by zeroing out its contents, refer to the fn\_util initsec command. Contact your service representative for assistance before initializing a database.

---

## Syntax

```
SEC_init -r | -i | -u
```

- r Resets the security table by overwriting all reserved objects in the database with the initial default values

The -r option affects **only** the set of reserved objects (users SysAdmin, Operator, and FieldService and groups SysAdminG, OperatorG, AuditG, and FieldServiceG), resetting their attributes to the default settings. It does not modify other user password information. You must be logged in as a member of fnadmin group or be logged on to the FileNet software as SysAdmin to use this option.

- i Initializes basic database fields, if not present
- u Updates the F\_OPEN database name

---

## Sample Output

---

```
corona(root)/> SEC_init -r
Overwriting...

Adding user defaults record: UserDefaults...
    (replacing...)
Adding group defaults record: GroupDefaults...
    (replacing...)
Adding device defaults record: DeviceDefaults...
    (replacing...)
Adding SysAdminG object...
    (replacing...)
Adding SysAdminG to the groups table...
    (exists)
Adding AuditG object...
    (replacing...)
Adding AuditG to the groups table...
    (exists)
Adding SysAdmin object...
    (replacing...)
Adding SysAdmin to the groups table...
    (exists)
Adding OperatorG object...
    (replacing...)
Adding OperatorG to the groups table...
    (exists)
Adding Operator object...
    (replacing...)
Adding Operator to the groups table...
    (exists)
(exists)
Adding system control table information...
    (replacing...)
corona(root)/>
.....
```

---



## Checklist

Before you use SEC\_init, be aware of the following:

- If you changed the passwords for the standard system-defined (reserved) groups and users from their defaults, the System Administrator must change them again through the security application after running SEC\_init -r.
- SEC\_init -r resets system-wide security defaults to their original default settings.

## Procedure

No specific procedure is required.

## Related Topics

See the “Security Administration” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

# SEC\_map

## Description

SEC\_map is a command line tool for mapping one Image Services user or group name to one or more Content Engine distinguished names (dn) in a Content Federation Services for Image Services (CFS-IS) environment. The mapped CE users and groups are stored in the sec\_map\_prin\_to\_dn table of the MKF Security database. Mapping IS security objects to CE security objects is required for exporting IS annotations to CE.

The SEC\_map tool does not have a graphical user interface (GUI).

## Use

SEC\_map walks you through a series of prompts and asks you to enter the appropriate values. Except for the help command, none of the other commands needs any parameters.

### Distinguished Names

There are two types of distinguished names:

- LDAP (Lightweight Directory Access Protocol)
- UPN (User Principal Name)

A distinguished name based on the **LDAP protocol** consists of key and value pairs:

key=value,key=value,key=value

The first key and value pair relates to the user identification. The prefix **cn** (common name) or **ui** (user id) are some of the preferred prefixes.

```
cn=smith,co=USA  
ui=smith,co=USA
```

A distinguished name based on the **UPN protocol** has a format similar to an email address:

```
smith@filenet.com
```

The maximum size of a distinguished name is **1023 characters**.

### User and Group Files

Since a distinguished name can be long, SEC\_map reads records from, and writes records to, a flat file. Each line in the file represents a record.

The two flat files are:

```
user.txt  
group.txt
```

The first line in the file has the format:

```
CE Domain Name<tab>CE Domain GUID<new line>
```

Each subsequent line represents a record with the format:

```
id<tab>name<tab>distinguished name<new line>
```

## Syntax

To launch the SEC\_map tool, enter the following at the command line:

### SEC\_map

The SEC\_map menu displays:

```
?
help
quit
logon
logout
set_dn_suffix
set_dn_prefix
set_ldap_type
export_all
export_blank
import
help
```

Enter commands at the <SEC\_map> command line prompt.

From the <SEC\_map> prompt, you can display online help for each command by entering **help** followed by the command name. For example:

```
<SEC_map>help set_ldap_type
```

## Commands

- ?** Displays the menu of SEC\_map commands.
- quit** SEC\_map terminates gracefully.

**logon** SEC\_map prompts you for a user name and password. It attempts to logon to the Image Services server using the entered user name and password.

If you are already logged onto the Image Services server, SEC\_map logs you out and then tries to log you on again.

**logout** Logs you out of the Image Services server. If you were not logged on, SEC\_map ignores the command.

**set\_dn\_suffix** Prompts you to enter a suffix for the distinguished name.

- The maximum length for a suffix is **922 ASCII characters**.
- The maximum length for a user name is **40 ASCII characters**.

Using the **LDAP protocol**, anything that appears after the 'cn=user-name', is considered to be a suffix. In the example, "co=USA" is a suffix.

```
cn=smith,co=USA  
ui=smith,co=USA
```

A distinguished name based on the **UPN protocol** has a format similar to an email address:

```
smith@filenet.com
```

Anything that appears after the '@' sign is considered to be a suffix.

**set\_dn\_prefix** Prompts you to enter a prefix to precede the user name.

- The maximum length for a prefix is **16 ASCII characters**.
- The maximum length for a user name is **40 ASCII characters**.

---

Using the **LDAP protocol**, the prefix **cn** (common name) or **ui** (user id) are some of the preferred prefixes.

**cn**=reza,co=USA

**ui**=reza,co=USA

---

**Note** The **UPN protocol** does not use a prefix for the distinguished name.

---

**set\_ldap\_type** SEC\_map prompts you to select either LDAP protocol or UPN protocol.

**export\_all** Generates two flat files: user.txt and group.txt. Each line in the file represents a record. The first line in each file has the format:

CE Domain Name<tab>CE Domain GUID<new line>

Each subsequent line represents a record with the format:

id<tab>name<tab>distinguished name<new line>

- If you have not logged on, SEC\_map walks you through the logon procedure.
- If you have not set the LDAP type, SEC\_map prompts you to select either LDAP protocol or UPN protocol.
- If you selected LDAP protocol, but have not set the prefix for the distinguished name, then SEC\_map prompts you to enter a prefix.
- If you have not set the suffix for the distinguished name, SEC\_map prompts you to enter a suffix.

SEC\_map also prompts you to enter a CE domain name. If it finds the CE domain GUID for the entered CE domain name, then it records this information to the flat file as the first line:

---

CE Domain Name<tab>CE Domain GUID<new line>

SEC\_map goes through the security database and generates a record for each user name and saves the record in the flat file. Devices are not considered to be users.

**export\_blank**

Generates two flat files named user.txt and group.txt. Each line in the file represents a record. The first line in the file has the format:

CE Domain Name<tab>CE Domain GUID<new line>

Each subsequent line represents a record with the format:

id<tab>name<tab>distinguished name<new line>

- If you have not logged on, SEC\_map walks you through the logon procedure.
- If you have not set the LDAP type, SEC\_map prompts you to select either LDAP protocol or UPN protocol.
- If you selected LDAP protocol, but have not set the prefix for the distinguished name, then SEC\_map prompts you to enter a prefix.
- If you have not set the suffix for the distinguished name, SEC\_map prompts you to enter a suffix.

SEC\_map prompts you to enter a CE domain name. If it finds the CE domain GUID for the entered CE domain name, then it records this information in the flat file as the first line: CE Domain Name<tab>CE Domain GUID<new line>

SEC\_map goes through the security database and for each user name that does not have a distinguished name, it generates a record and saves the record in the flat file. Devices are not considered to be users.

**import** SEC\_map prompts you to enter a file name to import, either user.txt or group.txt. SEC\_map reads the flat file and parses the records in the file. For each user name, it validates the user id. If the validation passes, the distinguished name is inserted into the security database and the record is deleted from the file. Otherwise, the record is kept in the file.

The first line in the file must have the format:

```
CE Domain Name<tab>CE Domain GUID<new line>
```

Each subsequent line represents a record with the format:

```
id<tab>name<tab>distinguished name<new line>
```

- If you have not logged on, SEC\_map walks you through the logon procedure.

## Procedure

### Mapping User and Group Security for Annotations

After you have created the document class mapping in Enterprise manager on the Content Engine server, you can use the **SEC\_map** tool on the Image Services system to map the user and group security for annotations.

- 1 Launch the SEC\_map tool by entering at a command prompt:

```
SEC_map
```

- 2 At the SEC\_map prompt, enter export\_all to create the user.txt and group.txt files.



SEC\_map> **export\_all**

- 3 Answer the prompts to logon, select the LDAP type, and select a suffix for the distinguished name.

SEC\_map reads the MKF security database and creates the user.txt and group.txt files in the current directory. The IS user and group names are converted to CE distinguished names based on your answers to the previous prompts.

- 4 When the export is finished, exit from SEC\_map:

SEC\_map> **quit**

- 5 Now you can use your preferred text editor, such as **vi**, to view these two files and verify that all the IS users and groups are mapped correctly to the appropriate CE distinguished names. Make any changes you wish, and save the files.

---

**Tip** If necessary, you can delete the two files, and run **export\_all** again.

---

- 6 When you're sure the user.txt and the group.txt are correct, you can launch SEC\_map again to import them into the MKF security database:

**SEC\_map**  
SEC\_map> **import**

- 7 Again, follow the prompts to import the user.txt file, then the group.txt file. SEC\_map validates each distinguished name before storing it in the MKF Security database on the IS Root server.

After an entry is stored in the MKF security database, the corresponding entry is deleted from the user.txt or group.txt file. If an entry cannot be validated, it is not removed from the file.

- If all the distinguished names are valid and the user.txt and group.txt files are empty, SEC\_map automatically deletes them.
- If any invalid entries remain in either file, SEC\_map does not delete the file. You can then re-edit the file, correct the invalid entries, and re-run the import.

## Related Topics

See the “Annotations” section in Chapter 3 of the *FileNet P8 Content Federation Services for Image Services Guidelines*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## SEC\_rm\_config

### Description

SEC\_rm\_config invokes the Image Services Record Management Configuration Utility. This utility is used in a Content Federation Services for Image Services (CFS-IS) environment and provides you the ability to display and edit current Record Management configuration security settings on the Image Services server.

### Use

Use the SEC\_rm\_config utility to set the three system level settings: Record Security level, the Record Management Group and the Record Management Log level. You must run this tool on the Root server as a member of the FNADMIN group.

#### Record Security Level

The Record Security Level (also referred to as Lockdown Mode) can be set to READ\_ONLY, APPEND\_ONLY, or NO\_CHANGE.

---

**Note** These security level settings apply only to normal users and do not impact the SysAdminG user.

---

- READ\_ONLY - Any user that is not a member of the SysAdminG group and already had at least Read privileges to the documents before the lockdown can only view the documents. This is the default setting.
- APPEND\_ONLY - Any user that is not a member of the SysAdminG group and already had at least Read and Append/Execute

privileges to the documents before the lockdown can only view the documents and add new annotations.

- NO\_CHANGE - The same security settings as currently exist in the Index Database of the IS system.

---

**Note** Once the Record Level Security has been set, and documents have been locked down at that level, changing the Record Level Security to a less restrictive setting does not change the security level of these documents. This is true even if you lock the same documents again. You can only make the security level more strict. For example, you cannot change the Record Level Security from READ\_ONLY to APPEND\_ONLY.

If you change the Record Level Security to a less restrictive setting, only future documents will be locked at the new level. If you change the RM Group, you must re-declare all of the old records for the changes to take place.

---

## RM Group

The RM Group names the Record Management group and a single RM group is used at the system level. When the tool is first launched, the value for the RM Group is UNDEFINED and any attempts to lock-down records will fail until the RM Group is named. It is recommended that a new IS security group be created as the RM Group using the Security Administration tool in the Application Executive (Xapex) utility as opposed to using an existing security group. Be aware that you cannot assign an IS user for the RM group, it must be an IS security group.

Once documents are declared as records (locked down), they inherit the security attribute (the Group ID) of the configured RM Group. If the

RM Group must be changed, the CE configured IS user (the IS user associated with the IS Fixed Content Device) must always be a member of the new RM Group. For example, if a group named **Records** was initially assigned to the RM Group, and then a new group named **Records\_grp** is assigned to the RM Group, the CE Configured user must be made a member of both Records and Records\_grp.

---

**Note** To provide the maximum protection, no other IS user should ever be made a member of the RM Group.

---

### RM Log Level

The RM Log Level sets the level of detail the system writes to the system's log file in the Edit Activity Log Level section. This can be set to MINIMAL or VERBOSE.

- MINIMAL - Logs errors only associated with any Records Management operations. This is the default setting.
- VERBOSE - Logs errors and detailed information on the document IDs being locked and the updated security.

---

**Note** The Edit Activity Log Level section is referring to logging to the system's log file and has nothing to do with the Activity Logging feature.

---

## Syntax

To launch the SEC\_rm\_config utility, on the systems Root server, logon as a member of the FNADMIN (FNSW) group and enter the following at the command line:

### **SEC\_rm\_config**

The IS Record Management Configuration Utility screen displays.

```
c:\WINNT\system32>SEC_rm_config

Image Services Record Management Configuration Utility

Command line options:

d      Display current configuration settings
e      Edit configuration settings
s      Save configuration settings
p      Print menu

q      Quit

Enter command =>
```

From this screen, you have four options in addition to quitting the utility: displaying the current IS configuration, editing the IS configuration, saving the current settings, or printing the menu.

### d option

The **d** option displays the current configuration settings for the IS system and the selected user (the first time, the current user configuration displays the same as the IS configuration, the default settings):

```
Enter command =d

Current Image Services Configuration:
Record Security Level      : READ_ONLY
RM Group                   : UNDEFINED
RM Log Level               : MINIMAL

Current User Selected Configuration:
Record Security Level      : READ_ONLY
RM Group                   : UNDEFINED
RM Log Level               : MINIMAL
```

### e option

The **e** option allows you to edit the current settings:

```
Enter command =e

Edit Record Security Level
s  READ_ONLY
w  APPEND_ONLY
n  NO_CHANGE
Enter Record Security Level [READ_ONLY] =>

Edit Record Manager Group
Enter Record Manager Group name [UNDEFINED] => records

Edit Activity Log Level
m  MINIMAL
v  VERBOSE
Enter Activity Log Level [MINIMAL] =>
```

In the example above, the Record Security Level setting and the Activity Log Level setting were left as the default by pressing the Enter key, and the Record Manager Group was named **records**.

### s option

The **s** option must be selected to save your changes. However, you are prompted to save your changes if you go into the edit menu and select q to exit.

### p option

The **p** option prints the main menu.

---

**Note** See the [“Sample Output” on page 1160](#) to see a sample of a typical SEC\_rm\_config session

---

## Sample Output

The following is a sample of a typical SEC\_rm\_config session.

```
c:\WINNT\system32>SEC_rm_config

Image Services Record Management Configuration Utility

Command line options:

d      Display current configuration settings
e      Edit configuration settings
s      Save configuration settings
p      Print menu

q      Quit

(contintued on next page)
```



```
(continue from previous page)

Enter command =>e

Edit Record Security Level
s  READ_ONLY
w  APPEND_ONLY
n  NO_CHANGE
Enter Record Security Level [READ_ONLY] => s

Edit Record Manager Group
Enter Record Manager Group name [records] =>

Edit Activity Log Level
m  MINIMAL
v  VERBOSE
Enter Activity Log Level [VERBOSE] =>m

Enter command => d

Current Image Services Configuration:
Record Security Level      : READ_ONLY
RM Group                   : records
RM Log Level               : VERBOSE

Current User Selected Configuration:
Record Security Level      : READ_ONLY
RM Group                   : records
RM Log Level               : MINIMAL

Enter Command => s

Saving Record Management control table information...
SEC_rm_config: Record Management settings have been successfully updated

(continued on next page)
```

```
(continue from previous page)

Enter Command => d

Current Image Services Configuration:
Record Security Level      : READ_ONLY
RM Group                   : records
RM Log Level               : MINIMAL

Current User Selected Configuration:
Record Security Level      : READ_ONLY
RM Group                   : records
RM Log Level               : MINIMAL

Enter Command => q

Exiting...
```

It is important to note that once you edit the security settings and then display them before saving them, you will see what the system settings currently are in the Current Image Services Configuration area of the display and what they will become in the Current User Selected Configuration area of the display once you save the settings using the `s` option. After you display the setting after saving them, you will see the two areas are the same again.

# SEC\_tool

## Description

System administrators and your service representative can view internal details of the Security Services and can perform certain administrative actions with SEC\_tool.

## Use

Use SEC\_tool to view and update various security aspects of the Image Services system. Some of the SEC\_tool commands require SysAdmin privileges. To obtain SysAdmin privileges, you should log onto the Image Services server as SysAdmin before running SEC\_tool.

You can also log onto the Image Services server for all platforms using Xapex. On a UNIX-based Image Services server, you can also use the fnlogon command.

## Syntax

SEC\_tool

Enter commands at the <SEC\_tool> command line prompt.

## Commands

In the command descriptions that follow, replace items in angle brackets (< >) with the appropriate value; items in square brackets ([ ]) are optional. All commands are case insensitive; therefore, you can enter any keyword or database item name in either uppercase or lowercase.

Although you can enter all letters of a command name, SEC\_tool only requires that you enter the letters shown in this section with UPPER-CASE letters. For example, to initiate the command shown as “DECOde” with its command syntax, you must enter at least the following characters:

### **deco**

SEC\_tool provides a variety of commands. From the <SEC\_tool> prompt, you can display online help for each command by entering a question mark followed by the command. For example:

<SEC\_tool>? **decode**

The following table lists all commands with a brief description of each:

SEC\_tool Application Commands

<b>Application Command</b>	<b>Description</b>
DECOde	Converts encoded value to decoded value
DUMP	Copies security database information to the specified file
ENCOde	Converts access description to encoded value
EXPort	Dumps security tables
HARDcopy	Sends output to a file
HELP	Displays help information
IDTOName	Converts a security ID to a name
IMPort	Copies tables from an exported file to the security database
LICense	Displays license information
LOGOFF	Kills security sessions

## SEC\_tool Application Commands, Continued

Application Command	Description
NAMETOid	Converts a string name to a security ID
PASswd	Changes your logon password without going through Security Administration application
Quit	Exits SEC_tool
TERMOFF	Turns off terminal output
TERMON	Turns on terminal output
STATs	Displays internal security statistics
WHO	Displays all logged-on users
WHOAMI	Displays your current logon information
?	Displays help information

The following commands contain detailed explanations of all options, parameters, and arguments you need to use the application commands.

### **DECOde [<number> | <number> | <...>] | [ 0x<hexstring>]**

Converts an encoded access restriction value to its decoded value

**<number>** A decimal number less than or equal to 255; you can specify up to 12 decimal numbers separated by spaces. Alternatively, you can specify a hexadecimal value.

**<0x<hexstring>** A hexadecimal string where <hexstring> is 1 to 12 hexadecimal bytes preceded by the identifier 0x. Alternatively, you can specify a decimal value.

**DUMP [ dump\_type ]**

Displays information from the security database based upon the specified dump\_type parameter. The dump types are:

## Security Dump Types

Dump Type	Description
user	Displays all users on the system
group	Displays all groups on the system
device	Displays all devices on the system
allobj	Displays all users, groups, and devices on the system
userid <numlogons> <numconcur>	Displays specific users on the system with logon information. If no parameters are provided, all users are returned.
usermbr	Displays the extended membership list for each user
groupmbr	Displays the extended membership list for each group
devicembr	Displays the extended membership list for each device
allmbr	Displays the extended membership list for each user, group and device
adminusr	Displays all administrative users on the system
admingrp	Displays all administrative groups on the system
adminmbr	Displays all objects which can be affected through the administrative group's extended membership
alladmin	Displays information for all of the above listed administrative types (adminusr, admingrp, adminmbr)
all	Displays information for all of the above listed types

For example, to display a list of users whose logons are greater than one, and/or whose concurrent logons are greater than 1, you would enter at the SEC\_tool prompt:

<SEC\_tool>dump userid 1 1

The resulting display would look similar to this:

```
-----  
User Object Listing          Thu Mar 16 17:01:47 2006  
  
Total logons greater than:      1  
Concurrent logons greater than:  1  
-----  
id          name                # logons # concur  
                                     logons  
-----  
00011 SysAdmin:vermeer:FileNet      28        2  
10021 bryan1:vermeer:FileNet        2        1  
10022 blambert:vermeer:FileNet     5        1  
10023 bkl:vermeer:FileNet          4        1  
10024 bryan1:vermeer:FileNet       3        1  
-----  
userids with > 1 logons and/or > 1 concurrent logons:      5  
Number of users with more than 1 logons:                    5  
Number of users with concurrent logons set greater than 1:  1  
-----
```

If you enter the **dump userid** option with no parameters:

<SEC\_tool>dump userid

The resulting display would look like this:

```

-----
User Object Listing           Thu Mar 16 17:03:16 2006
-----
  id          name                # logons # concur # max
                                logons   sessions
-----
00011 SysAdmin:vermeer:FileNet      28         2         3
00012 FieldService:vermeer:FileNet 0         0         1
00013 Operator:vermeer:FileNet     0         0         1
00021 user_test100:vermeer:FileNet  0         0         1
00022 user_test101:vermeer:FileNet  0         0         1
00023 user_test102:vermeer:FileNet  0         0         1
00024 user_test103:vermeer:FileNet  0         0         1
00025 user_test104:vermeer:FileNet  0         0         1
00026 user_test105:vermeer:FileNet  0         0         1
-----
Total userids stored:                9
Number of users with one or more logons: 1
Number of users with concurrent logons > 1: 1
-----

```

### **ENCOde** <read\_id> | <write\_id> | <ax\_id>

Converts an access description to an encoded value

### **EXP**ort [<filename>]

Dumps (exports) the security tables, in ASCII format, to the specified file. This command requires SysAdmin privileges.

**Note** Do not use this option to export the security database with the intention of importing the security tables on a new server. Doing so can prevent you from retrieving documents with certain document classes on the



new server. When performing a server-to-server migration, we recommend using the Backup and Restore Utility.

---

### **HARDcopy [<filename>]**

Sends program output to a file. If you specify a <filename>, SEC\_tool creates a file of that name, and sends all output to both your terminal screen and to this file. If you do not specify a <filename>, SEC\_tool closes the previous output file and turns off hard copy output.

### **HELP [<command> | \* ]**

Displays help information. If you specify <command>, it prints help for the indicated command. To get help for all commands, enter:

**HELP \***

### **IDTOName <id>**

Converts a specified security ID to a name. If it can't find the ID in the object table, the system searches the deleted object table. If the ID corresponds to a deleted object, a message informs you the system has deleted the object, displaying its name.

### **IMPort [<filename>] [overwrite | overwrite\_all] [from=<from\_domain>] [to=<to\_domain>]**

Imports tables from an exported file to the local security database. This command requires SysAdmin privileges.

**<filename>** File name of the exported security tables. This parameter is required.

**overwrite** A literal string that, when specified, overwrites the **reserved** FileNet objects in the target database with the related import objects data. A

reserved object is an object, created by the SEC\_init command (as it is issued from SysAdmin, SysAdminG, or AuditG, for example).

### **overwrite\_all**

A literal string that, when specified, updates the **all** FileNet security objects in the target database with the information contained in the specified <import\_filename>. All security objects, both reserved and custom, are updated with the same id contained in the file. If the id on the target system is already taken, the object associated with that id is removed during the import process.

This option is used primarily when importing the security database in preparation for an upgrade to IS 4.0 SP5 as described in the document, *Updating the MKF Security Database (Prerequisite for IS 4.0 SP5)*, which is available from your service representative.

---

### **Important**

After you export the security database, **you should not make any changes to Image Services security** before you import it with the overwrite\_all option. If you do, unexpected results can occur.

For example, if you export the security database and then **delete** a session group, you would expect the session group to be restored with all its members intact when you import the security base with the overwrite\_all option. However, the session group is not restored to its previous state. The group is restored, but previous members of the group are **not** reinstated.

Also, if you export the security database and then **rename** a session group, you would expect the group to be restored to its previous name and membership when you import the security database with the overwrite\_all option. However, although the group is returned to its original name, the previous members are **not** reinstated.

---

**from=<from\_domain>** When specified, replaces all domain names in the <from\_domain> to with the domain names specified in the to=<to\_domain> parameter.

**to=<to\_domain>** When specified, replaces all domain names in the domain specified in the from=<from\_domain> parameter with the name specified in <to\_domain>.

## LICense

Displays licensing information for the local domain, as shown in the example below. (For SLU column descriptions, see [“SEC tool License Report Column Descriptions” on page 1172.](#))

```
<SEC_tool>
SLU = Simultaneously Logged on Users
```

License Type	Current Users	Soft SLU Limit	Hard SLU Limit	Soft SLU Violations	Rejections	Maximum Usage
1	3	99999	99999	0	0	12
2	0	99999	99999	0	0	0
3	0	999	999	0	0	0
4	0	999	999	0	0	0
5	0	999	999	0	0	0
6	0	999	999	0	0	0
7	0	999	999	0	0	0
8	0	999	999	0	0	0
9	0	999	999	0	0	0
10	0	999	999	0	0	0

## SEC\_tool License Report Column Descriptions

Heading	Description
License Type	<p>This number identifies the type of license the customer has purchased.</p> <p>1 = Image Services only SLUs 2 = eProcess only SLUs</p>
Current Users	<p>Displays the number of users logged onto the Image Services system at the end of the report sample interval</p>
Soft SLU Limit	<p>License management policy enables users to obtain Universal SLAC keys that set this value to 99,999. Although other system constraints would prevent this many users from logging onto an Image Services server at the same time, this number indicates that up to 99,999 users can log on simultaneously.</p> <p>Customers are still bound not to exceed the license quantity purchased; however, the Universal SLAC keys do not lockout connections. Comparison of report results with customer license configurations is a sales team activity. For more details, refer to the Marketing Bulletins regarding SLU License Management.</p> <p>This field is called the "SOFT CUTOFF" in the Customer Support and Service Report.</p>
Hard SLU Limit	<p>License management policy enables users to obtain Universal SLAC keys that set this value to 99,999. Although other system constraints would prevent this many users from logging onto an Image Services server at the same time, this number indicates that up to 99,999 users can log on simultaneously.</p>

## SEC\_tool License Report Column Descriptions, Continued

Heading	Description
Hard SLU Limit, cont.	<p>Customers are still bound not to exceed the license quantity purchased; however, the Universal SLAC keys do not lockout connections. Comparison of report results with customer license configurations is a sales team activity. For more details, refer to the Marketing Bulletins regarding SLU License Management.</p> <p>This field is called "MAX ALLOWED" in the Customer Support and Service Report.</p>
Soft SLU Violations	This field does not apply to the current software release.
Rejections	The Image Services system rejects all users after the number specified as Hard SLU Limit has been reached.
Maximum Usage	Displays the highest number of users logged on since the Image Services system was last recycled. This parameter is called the "PEAK" in the customer service report generated with getreports.

**LOGOFF [ all | (user=<user\_name> | endpt=<endpt\_name>) ]**

Terminates security sessions

**all** Logs off all users. To execute LOGOFF with the **all** parameter, you must be SysAdmin or a user possessing the Supervisor or Principal attribute. You cannot use **all** in conjunction with any other parameter of the LOGOFF command.

**user=<user\_name>** Identifies the name of a logged on user. You can determine the logged on users by using the who command. If you specify the user's name, the program terminates **every** logon instance of that user. If used in

conjunction with the `endpt=<endpt_name>` parameter, it terminates only that user's **specific** instance. A user can terminate his or her own logon instances but only the SysAdmin user or a user with the Supervisor or Principal attribute can terminate another user's logon instance.

If you use this option, but do not specify a `<user_name>`, SEC\_tool issues a syntax error message.

**endpt=<endpt\_name>**

Identifies the name of the logged-on endpoint. You can use the `who` command to determine logged-on endpoint names. You must be a non-system administrator user to use the `user=<user_name>` option in conjunction with this option.

If you use this option, but do not specify an `<endpt_name>`, SEC\_tool issues a syntax error message.

**NAMEToid <name\_string>**

Converts the specified string name to an ID

**PASswd**

Use this command to change your logon password without going through the Security Administration application windows. SEC\_tool displays your current user ID in the field "User:" and prompts you for your old password, new password, and confirmation of your new password. These passwords do not appear on the screen as you type them. This command requires SysAdmin privileges.

The example below shows the prompt sequence when user “tmkppr” enters the PAS command:

```
<SEC_tool>pas
User: SysAdmin:allaha:FileNet
Old password:
New password:
Retype new password:
Password has been updated.
```

### Quit

Exits the SEC\_tool program

### TERMOFF

Turns off terminal output. You might want to do this when writing large amounts of data to an output file using the HARDCOPY command.

### TERMON

Turns on terminal output. Use this command to restore terminal output after it was shut off with TERMOFF.

### STAts

Displays the internal security statistics as shown below:

---

```
SEC_tool>stats
Events
-----
FileNet software was started at: Tue Dec 23 10:54:24 1997
deserialize calls:      total=0 as of Wed Dec 31 16:00:00 1969
service_logon calls:    total=25 as of Mon Jan 05 11:52:27 1998
```

---

find\_object calls:           total=201 as of Mon Jan 05 16:21:42 1998

MKF db hits

-----

obj finds:           1988  
obj adds:            0  
obj updates:         159  
obj deletes:         0

-----

total:               2147  
grp finds:           1110  
grp adds:            0  
grp deletes:         0

-----

total:               1110  
sys finds:           201  
sys updates:         3

-----

total:               204

del finds:           0  
del adds:            0  
del deletes:         0

-----

total:               0

func finds:          11  
func adds:           0  
func deletes:        0

-----

total:               11

fmbr finds:          3  
fmbr adds:           0  
fmbr deletes:        0

-----

total:               3



---

grand total: 3475

---

### **WHO [ ( v | user=<user\_name> | endpt=<endpt\_name> ) | <minutes> ]**

Displays a list of logged-on users.

- v** Verbose mode. This option displays detailed information related to a logon session. The detailed information includes the user\_name, the endpt\_name, the logon\_time, the calculated expiration\_time and the number of sessions. If you do not specify verbose mode, then just the user and endpoint names are displayed in a columnar format.

**user=<user\_name>** Identifies the name of a logged-on user. The <user\_name> is a string by which this command attempts to filter through the list of logged-on users. This string does not need to be the full-length user name. All users partially matching this string will be displayed. For example, a search for all logged-on users whose names begin with the letter 'S' can be executed by specifying 'user=S'.

**endpt=<endpt\_name>** Identifies the name of the logged-on endpoint. Like the user name parameter, the endpoint name parameter allows you to specify a partial string. If both the endpoint name and the user name parameters are specified, then the displayed logged-on user must meet **both** search criteria. This parameter can be useful in identifying all logged-on users on a certain type of device. Since all FileNet Personal Computer (PC) endpoint names begin with 'PC', it would be possible to list these logon sessions by specifying 'endpt=PC'.

**<minutes>** This option reports all users that logged on the system within the specified number of minutes. This parameter must be an integer and is the number of minutes back from the current time. It is meant to be used

alone. If it's used with the v, user, or endpt parameters, this option will override (ignore) them.

**Note** The number of users that have logged on during the specified number of minutes is not necessarily the same as the number of currently logged on users.

For example, if you specified a time period of 30 minutes, some users can have logged on for a few minutes and then logged off within that time span. They would be included in the list.

There can also be some currently logged on users who first logged on, say, 45 minutes ago that would not be included in the list.

Here are some examples:

- Display a standard list of all the currently logged on users:

```
<SEC_tool>who
=====
2 sessions found on Tue Apr 11 16:21:19 2006
=====

user_name                               endpoint_name
-----
ServiceProcess:System:System            ServProcTerm@0.0.0.0
SysAdmin:senna:FileNet                  SV001.0000@10.14.100.22
```

- Display a list of all logged on users in verbose mode:

```

<SEC_tool>who v
=====
2 sessions found on Tue Apr 11 16:23:33 2006
=====

----- 1 -----
who:          ServiceProcess:System:System
endpoint:     ServProcTerm@0.0.0.0
logon_time:   Mon Apr 03 09:32:25 2006
exp_time:    no expiration
sessions:     56

----- 2 -----
who:          SysAdmin:senna:FileNet
endpoint:     SV001.0000@10.14.100.22
logon_time:   Fri Apr 07 15:13:22 2006
exp_time:    no expiration
sessions:     1

```

- Display a list of logged on users whose user names begin with Se:

```

<SEC_tool>who user=Se
user_name                endpoint_name
-----
ServiceProcess:System:System  ServProcTerm@0.0.0.0

```

- Display a list of logged on users whose endpoint names begin with SV:

```
<SEC_tool>who endpt=SV
user_name                               endpoint_name
-----                               -
SysAdmin:senna:FileNet                  SV001.0000@10.14.100.22
```

- Display a list of users who logged on in the last 15 minutes:

```
<SEC_tool>who 15
-----
Unique Logons      Wed Apr 19 14:26:33 2006
-----

id      name                               last logon
-----
00011 SysAdmin:senna:FileNet          4/19/2006 2:12:41 PM
00022 blambert:senna:FileNet         4/19/2006 2:13:50 PM
00023 bryan1:senna:FileNet           4/19/2006 2:18:13 PM
00024 bryan2:senna:FileNet           4/19/2006 2:23:07 PM
=====
4 unique user logons in last 15 minutes
=====
```

## WHOAMI

- Displays the FileNet user's current logon information:

```
<SEC_tool>whoami

FileNet name:      SysAdmin:allaha:FileNet
FileNet id:        11
prim name:         SysAdminG:allaha:FileNet
prim id:           2
Admin classes:     supervisor; principal; group; password;
```

?

Displays all help information

## Checklist

Before you use SEC\_tool, be aware of the following:

- When you use the SEC\_tool **logoff** command to terminate a user, you must specify the user name. If you do not specify a name, SEC\_tool issues a syntax error message.
- When you use the SEC\_tool **logoff** command to terminate a logged-on end point, you must specify an end point name. If you do not specify a name, SEC\_tool issues a syntax error message.

## Procedure

Logon to SEC\_tool.

## Related Topics

[\*\*“fnlogon” on page 489\*\*](#)

[\*\*“getreports” on page 523\*\*](#)

See the “Security Administration” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## sgs

**UNIX**

The sgs tool is available only on a UNIX-based Image Services server.

### Description

The sgs (system get status) tool lists the system configuration settings on the Image Services server. This tool gathers specified information from the latest conf\_db file and displays it in an easy-to-read format.

You can use this tool on any server in which it is installed. When running this command, you must specify the type of information you want to view using the appropriate option.

If you run the sgs tool on a server that does not provide the function specified by the command option, the list does not present complete details.

For example, if you run sgs using the -o option on a server that does not provide an OSAR or storage library, it lists the OSAR command option format without any values describing an installed device.

### Use

Use sgs when you want to view the configuration settings on a UNIX-based Image Services server.

#### **CAUTION**

---

Once you issue an sgs command, do not abort its display before the system redisplay its prompt. Doing so leaves database calls open, hanging up the process.

---

## Syntax

```
sgs [-h | -d | -r | -n | -l | -t | -o | -e | -c | -p | -w]
```

---

**Note** This command requires a space between the command and its option.

---

- h** Displays the sgs command help screen, a list of command options
- d** Identifies the server details, including its domain name, system serial number, operating system version, and FileNet software version
- r** Displays information about any remote servers configured on the system, including the system type and domain name
- n** Displays the system configuration, including its description, serial number, Image Services release number, domain name, organization name, networking protocol, and type of system verification method used
- l** Displays server details, including the server's station number, type of server, IP address, and server name
- t** Displays details about the server's tape drive, including its network clearing house name, type, and capacity
- o** Displays details on the server's storage library, including its type, number of slots, current mode, drive capacity, and device address
- e** Lists all subsystems configured on the system
- c** Identifies the server's cache details, including the configured thresholds for each type of cache
- p** Identifies the server partitions allocated to system cache, as well as the various databases



- w Identifies printers and fax machines connected to the server, including the device IP address, name, and type

## Sample Output

After you run the sgs command, it collects the information for the specified configuration parameter and displays a warning banner, followed by the requested information.

### sgs -h Sample Display

The following example shows the command help list displayed using the **sgs -h** command option.

```
Help command for 'sgs'           Program Date: 03/26/97

syntax: sgs
      where -h lists this help screen           {default}
            -d lists domain main server identity
            -r lists all domain / remote servers
            -n lists system configuration
            -l lists server names and IP addresses
            -t lists tape information
            -o lists osar/optical information
              ( shows device links on OSAR server only )
            -e lists services on each server
            -c lists cache percentage information
            -p lists partition information
            -w lists printer/fax information
              ( shows device type on ROOT server only )
```

### sgs -d Sample Display

The following example shows a list displayed using the **sgs -d** command option. This option identifies the server details, including its domain name, system serial number, operating system version, and FileNet software version.

```
H(root)/> sgs -d

***** WARNING *****

Please DO NOT abort this program while running.

Please WAIT.. for a prompt.. or processes may be left hanging.

***** Program Rev: 97-06.1 *****

The number of Servers configured on this system.. 1

Collecting -->> DOMAIN/MAIN SERVER information. <<-- Please wait..

Domain_Name:   neelix:FileNet   Domain_No: 1
SSN: 1100201788
HP9000: B.11.0
IS: 4.0.0
```

### sgs -r Sample Display

The following example shows a list displayed using the **sgs -r** command option. This option displays information about any remote

servers configured on the system, including the system type and domain name.

```
(root)/> sgs -r

***** WARNING *****

Please DO NOT abort this program while running.

Please WAIT.. for a prompt.. or processes may be left hanging.

***** Program Rev: 97-06.1 *****

The number of Servers configured on this system.. 1

Collecting -->>  REMOTE_Systems information.  <<-- Please wait..

Domain / Remote Servers configured on this system.. 1

      Domain:      System_Type:      Name:
      -----      -
      1             IS                neelix
```

### sgs -n Sample Display

The following example shows a list displayed using the **sgs -n** command option. This option displays the system configuration, including its description, serial number, Image Services release number, domain name, organization name, networking protocol, and type of system verification method used.

(root)/> sgs -n

\*\*\*\*\* WARNING \*\*\*\*\*

Please DO NOT abort this program while running.

Please WAIT.. for a prompt.. or processes may be left hanging.

\*\*\*\*\* Program Rev: 97-06.1 \*\*\*\*\*

The number of Servers configured on this system.. 1

Collecting -->> SYSTEM CONFIGURATION information. <<-- Please wait..

System Configuration:

-----  
System Description: [ HP9000: B.11.0 ]  
System Serial Number: [ 1100201788 ]  
System IS Release: [ 4.0.0 ]

Domain Name: [ neelix ]  
Organization Name: [ FileNet ]

Protocols in use: [ TCP ]  
Preferred Protocol: [ TCP ]

System Checksumming: [ NONE ]

---

## sgs -l Sample Display

The following example shows a list displayed using the **sgs -l** command option. This option displays server details, including the server's station number, type of server, IP address, and server name.

```

sas1(root) > sgs -l

***** WARNING *****

Please DO NOT abort this program while running.

Please WAIT.. for a prompt.. or processes may be left hanging.

*****

The number of Servers configured on this system.. 1

Collecting -->>  SERVER LISTING information.  <<-- Please wait..

Sta_Num      Station_Type                IP Address      Server_Name
-----
   1   Root/Index/Document/Osar   [ 135.0.2.24 ]   sas1

```

### sgs -t Sample Display

The following example shows a list displayed using the **sgs -t** command option. This option displays details about the server's tape drive, including its network clearing house name, type, and capacity.

```
sasl(root)/> sgs -t

***** WARNING *****

Please DO NOT abort this program while running.

Please WAIT.. for a prompt.. or processes may be left hanging.

*****

The number of Servers configured on this system.. 1

Collecting -->> TAPE information. <<-- Please wait..

Tape Configuration for server... sasl
-----
NCH_name: rmt0 Device_driver: /dev/rmt0 Tape_type: 8mm 5GB/10GB

sasl(root)
```

### sgs -o Sample Display

The following example shows a list displayed using the **sgs -o** command option. This option displays details on the server's optical

---

storage library, including its type, number of slots, current mode, drive capacity, and device address.

---

```
sasl(root)/> sgs -o
```

```
***** WARNING *****
```

```
Please DO NOT abort this program while running.
```

```
Please WAIT.. for a prompt.. or processes may be left hanging.
```

```
*****
```

```
The number of Servers configured on this system.. 1
```

```
Collecting -->> OSAR/OPTICAL information. <<-- Please wait..
```

```
Number of Optical Storage Libraries on this Server= 1
```

```
OSAR Configuration for... sasl
```

```
-----
```

```
Library Unit: 0 (a): HP Autochanger, Model 20XT Slots: 16 Mode: Normal
```

```
Device Addr: 0 7 5 0
```

```
osara is linked to.. /dev/fnsod.0,7,5,0
```

```
Drive 1: HP C1716T (1.3 GB)
```

```
Device Addr: 0 7 2 0
```

```
oddal is linked to.. /dev/fnsod.0,7,2,0
```

```
-----
```

```
sasl(root)/>
```

---

## sgs -e Sample Display

The following example shows a list displayed using the **sgs -e** command option. This option lists all subsystems configured on the system.

```
(root)/> sgs -e

***** WARNING *****

Please DO NOT abort this program while running.

Please WAIT.. for a prompt.. or processes may be left hanging.

***** Program Rev: 97-06.1 *****

The number of Servers configured on this system.. 1

Collecting -->> SERVICES information. <<-- Please wait..

Services Configured for server... neelix
-----
index doc batch cache print osar skf wqs nch sql rmt_file sec
```



## sgs -c Sample Display

The following example shows a list displayed using the **sgs -c** command option. This option identifies the server's cache details, including the configured thresholds for each type of cache.

```
(root)/> sgs -c
```

```
***** WARNING *****
```

```
Please DO NOT abort this program while running.
```

```
Please WAIT.. for a prompt.. or processes may be left hanging.
```

```
***** Program Rev: 97-06.1 *****
```

```
The number of Servers configured on this system.. 1
```

```
Collecting -->> CACHE information. <<-- Please wait..
```

```
Cache information for server... neelix
```

```
-----Thresholds-----
```

Cache_Type	Min_sec	Max_sec	Daemon	Locked	Write
Retrieval	20 %	20 %	80 %	85 %	90 %
Batch	10 %	60 %	80 %	85 %	90 %
System Print	10 %	20 %	80 %	85 %	90 %
Application Print	10 %	30 %	80 %	85 %	90 %
Fill-in	0 %	0 %	80 %	85 %	90 %
Revise	0 %	0 %	80 %	85 %	90 %
Folder View	0 %	0 %	80 %	85 %	90 %

## sgs -p Sample Display

The following example shows a list displayed using the **sgs -p** command option. This option identifies the server partitions allocated to system cache, as well as the various databases.

```
(root)/> sgs -p
```

```
***** WARNING *****
Please DO NOT abort this program while running.
Please WAIT.. for a prompt.. or processes may be left hanging.
***** Program Rev: 97-06.1 *****
```

```
The number of Servers configured on this system.. 1
```

```
Collecting -->> PARTITION information. <<-- Please wait..
```

```
Partitions Configured for server... neelix
```

```
-----
Disk:  PPs:    Mbytes:      Partition Name
      20      /fnsw/dev/1/oracle_rl2
      200     /fnsw/dev/1/oracle_tr0
      200     /fnsw/dev/1/oracle_udb0
      20      /fnsw/dev/1/oracle_rl1
      20      /fnsw/dev/1/oracle_rl0
      210     /fnsw/dev/1/oracle_db0
      100     /fnsw/dev/1/cache0
      100     /fnsw/dev/1/permanent_db0
      40      /fnsw/dev/1/permanent_rl0
      20      /fnsw/dev/1/transient_db0
      40      /fnsw/dev/1/transient_rl0
      12      /fnsw/dev/1/sec_db0
      4       /fnsw/dev/1/sec_rl0
```

## sgs -w Sample Display

The following example shows a list displayed using the **sgs -w** command option. This option identifies printers and fax machines connected to the server, including the device IP address, name, and type.

```
(root)/> sgs -w

***** WARNING *****

Please DO NOT abort this program while running.

Please WAIT.. for a prompt.. or processes may be left hanging.

***** Program Rev: 97-06.1 *****

The number of Servers configured on this system.. 1

Collecting -->> PRINT/FAX information. <<-- Please wait..

Prt/Fax   IP Address   Printer_Name   Printer_Type
-----
  1      [135.0.75.230]      p1
```

## Procedure

Enter the **sgs** command, followed by a space and the option for the type of information you want to obtain.

## SNT\_update

### Description

The SNT\_update tool updates the scalar numbers table (SNT) after a restore operation of the MKF permanent database.

### Preventing Duplicate Document Numbers

The scalar numbers table (SNT) of the MKF permanent database keeps a record of the next available numbers for several Image Services and Content Services number spaces. One of these numbers is the next available document ID. Whenever a system operator enters a new document into the system, the Image Services system assigns it a document ID using an increasing number sequence. It obtains the next available number from SNT when creating a new document or batch of documents.

If a magnetic disk crash occurs that results in the loss of the permanent database, it also loses the SNT. This requires restoring the permanent database from the last available backup tape. (A full restore of the permanent database overwrites the current values in the SNT with old values from the SNT of the restored system.) It uses any available transaction logs to roll the database forward to the most current transaction. However, it might not always be possible to roll forward to the exact moment of the crash so the permanent database might become unsynchronized with the index database and documents on storage media. If this unsynchronized condition occurs, the next available document ID in the SNT could be well below what it should be. After the operator restarts the FileNet software, the system might duplicate document IDs when scanning in new documents.

To prevent assigning duplicate document IDs, the system refers to the scalar numbers table checkpoint file, `snt.chkpt`, located in the `/fnsw/local/sd` directory for UNIX platforms or `\fnsw_loc\sd` directory for Windows Server platforms. As shown in the example below, this checkpoint file contains a backup copy of the critical data from the SNT (next available document ID, next surface ID, next background job ID, and encoded date/time stamp):

```
5018595 3344 98 856828856
```

The system updates the `snt.chkpt` file any time it allocates a new surface ID or creates a background job, once for every 1,000 new documents allocated. The system time stamps the file to help you determine how old the checkpoint file is compared to the last SNT restore. It decodes the encoded timestamp for display when you start `SNT_update`. (See the example on [page 1200](#).)

### Automatic Checkpoint Verification

Each time Document Services starts up during a start or restart of FileNet software, the system compares the values in the `snt.chkpt` file against the values in the SNT of the permanent database. If any of the `snt.chkpt` file values is higher than those in the SNT, Document Services terminates and logs the following message:

```
Severe Error condition: The Scalar Numbers Table is behind the snt.chkpt
file. This should only happen after a Permanent DB restore has been done.
Continuing with this condition may cause multiple documents to be committed
with the same doc ID. To resolve this problem, you must either remove the
snt.chkpt file (if its contents are invalid) or update the Scalar Numbers
Table with the SNT_update program. Doc Services will not function until
this problem is resolved.
```

SNT\_update reads the contents of the snt.chkpt file and adds 1,000 to the next available document ID value to ensure that the scalar numbers table is updated to the highest possible value. (Remember that the snt.chkpt file is only updated for every 1,000 newly allocated documents.) For example, if the value of the next document ID in the snt.chkpt file is 5018595, SNT\_update modifies the value in the checkpoint file to 5019595 during the update.

## Use

The SNT\_update program updates the scalar numbers table of the MKF permanent database from information obtained from the SNT checkpoint file. You could use SNT\_update to prevent creation of duplicate documents, surfaces, and background jobs under one of the following conditions:

- After restoring the MKF permanent database from tape: You might restore data if you lose the database due to a magnetic media crash and the /fnsw/local (\fnsw\_loc) partition is still available and undamaged.
- After initializing the software: You might use an initialization tool (such as ds\_init, fn\_util init, or fn\_util initperm) to set software parameters back to their original values.

## Syntax

SNT\_update

The program prompts you to quit or continue before it begins updating the scalar numbers table (SNT).

## Checklist

Before you use SNT\_update, be aware of the following:

- You must not run SNT\_update if the /fnsw/local (\fnsw\_loc) partition has been lost or damaged. If the partition is lost or damaged, call your service representative for assistance in restoring a valid SNT from a backup copy of the snt.chkpt file.
- SNT\_update could waste up to 1,000 document IDs but does not waste any surface IDs or background job numbers.

## Procedure

The following procedure updates the scalar numbers table with a valid checkpoint file.

---

**Tip** In the following procedure, references to the directory path for the snt.chkpt file depend on your platform, as follows:

For UNIX platforms: /fnsw/local/sd/snt.chkpt

For Windows Server platforms: \fnsw\_loc\sd\snt.chkpt

---

- 1 Before you start the restore or initialization operation, make a copy of the magnetic disk-resident scalar numbers table checkpoint file, snt.chkpt.

You can copy the checkpoint file to tape or to a directory such as /fnsw/local/tmp (in UNIX) or \fnsw\_loc\tmp (in Windows Server).

---

**Note** If you cannot save the checkpoint file prior to performing a restore or initialization or if a magnetic disk crash has corrupted the checkpoint file, call your service representative immediately.

---

- 2 Perform the restore or initialization operation.
- 3 Verify that the restore or initialization operation did not overwrite the disk-resident snt.chkpt file.

If snt.chkpt was overwritten, copy the **saved** version of the snt.chkpt file (created in step 1 above) back into /fnsw/local/sd or \fnsw\_loc\sd, depending on your platform.

- 4 Restart the FileNet software.

The Document Services initialization routine automatically compares the snt.chkpt file with the SNT table and, if necessary, issues messages directing you to run SNT\_update.

- 5 Enter **SNT\_update** at the command line.



An example of the confirmation display is shown below. The values for surface, job, and document IDs are for illustration only. The values in your display will be different:

```
corona>SNT_update
"The time stamp on your checkpoint is Tue Feb 26 13:45:52 1997
"The SCALAR_NUMBERS table values in the Permanent Database are:
      Next Surface  ID (MKF) = 3344
      Next Job      ID (MKF) = 98
      Next Document ID (MKF) = 5018595
The check point file values are:
      Next Surface  ID (checkpoint) = 3344
      Next Job      ID (checkpoint) = 98
      Next Document ID (checkpoint) = 5018598

      ===> New Next Document ID will be = 5019598

Do you wish to update the Permanent Database Scalar numbers with the
checkpoint
values? (y/n):
```

## 6 Respond to the update prompt.

If you reply **n**, SNT\_update terminates.

If you reply **y**, one of the following messages displays indicating successful completion or errors that prevent the successful update of the scalar numbers table:

- “Scalar numbers table updated”
- “Update was not necessary  
Scalar numbers table already up to date”
- “Scalar numbers table update failed, err=<err\_code>”

where <err\_code> is a value indicating the type of error encountered. SNT\_update terminates after displaying this message.

If SNT\_update does not complete successfully, call your service representative for assistance.

## spacerpt

### Description

The spacerpt tool lists space information for a whole database and all or specific tables of one Oracle user database.

The spacerpt tool displays information in table format to standard output (normally the terminal used to invoke spacerpt). Since the report can be lengthy, you might want to work in a scrollable window or redirect the output from spacerpt to a file, from which you can then view the information using the less or more commands.

---

**Note** The output report formats for Microsoft SQL Server are slightly different than those for Oracle.

For IBM DB2, this tool is not supported.

For information on running spacerpt on a remote server, see [\*\*“Running on a Remote Oracle Server” on page 1214.\*\*](#)

---

### Use

The spacerpt program provides information about the space used by the FileNet tables in the index or WorkFlo databases. Run spacerpt weekly to establish a baseline measurement of space use. Then, depending on the rate of growth observed, run it more frequently (or less frequently) to monitor the database. You can run spacerpt on an Index server for the index database or a WorkFlo Queue server for a WorkFlo queue database.

## Syntax

```
spacerpt [-x] [-u <username> ] [-d <dbname> ... ]  
[<table name list>]
```

- x** Includes extended detailed reports on Oracle extents  
Due to the volume of output created, this option is recommended for use only on selected tables.

This option is valid only for Oracle databases.

- u <username>** Specifies the user that owns the tables to be reported  
User names commonly specified are f\_sw, f\_sqi (for Folder View tables), f\_open (for user tables). If you do not specify a user name, spacerpt produces a report on tables owned by the f\_sw and f\_sqi users.

- d <dbname>** Identifies one or more database names

---

**Note** You must specify each database name with the **-d** option, for example:

**- db1 -d db2**

If you specify **-d db1 db2**, the db2 is interpreted as a table name.

---

- <table name>** Specifies one or more table names for which space information displays

By default, spacerpt displays information for all index and WorkFlo tables.

## Sample Output

This section shows sample output reports from both Oracle and Microsoft SQL Server.

### Sample Output from Oracle Environment

The following is a partial listing of spacerpt output from an Oracle environment. The sample is followed by explanations of the report fields. The "... (listing continues)" text designates where entries have been truncated to reduce the size of the sample output listing for easier viewing.

```

                                SPACE REPORT
                                =====
SYSTEM NAME:  costa10                               Mon May 19 11:29:54 1997

Tablespace   File#   Size(KB)  NAME
-----
SYSTEM      1       5000  /usr/ora/816/dbs/dbs1IDB.dbf
            2     204794 /fnsw/dev/1/oracle_db0
TEST_TS     3     18432  /fnsw/sd/1/test1
            4       8192  /fnsw/sd/1/test2
            -----
                                236418
                                Free      Largest free
Tablespace   space(KB)  extent(KB)
-----
SYSTEM      107980    104878
TEST_TS     25540    17350

                                ROLLBACK SEGMENT SPACE

Rollback Segment                                Exts Max   Pct   Largest   Last   Total
                                                used exts  incr  ext (KB)  ext (KB)  Space (KB)
-----

```

RBS_1	8	121	0	10	10	80
RS0	6	121	0	100	100	600
SYSTEM	5	121	0	50	50	250

TABLE SPACE

Table	Exts used	Max exts	Pct incr	Largest ext (KB)	Last ext (KB)	Total Space (KB)
DOCTABA	1	#####	0	10000	10000	10000
DOCUMENT_CLASS	1	#####	0	50	50	50
DOC_CLASS_INDEX	1	#####	0	30	30	30
FOLDER	1	#####	10	10	10	10
FOLDER_CONTENTS	1	#####	10	10	10	10
FOLDER_TABS	1	#####	10	10	10	10
INDEX_CLUSTER	1	#####	0	10	10	10
... (listing continues)						

INDEX SPACE

Table	Index	Exts used	Max exts	Pct incr	Largest ext (KB)	Last ext (KB)	Total Space (KB)
DOCTABA	DA_A140	1	#####	0	50	50	50
	DA_A141	2	#####	0	10000	10000	10050
DA_ARCHIVEDATE		1	#####	0	50	50	50
	DA_DELETEDATE	1	#####	0	50	50	50
	DA_DOCNUMBER	1	#####	0	50	50	50
							81800

Each section of the output is described below.

**Part 1 Description**

The first part of the Space Report provides information about the entire index database, including a rollback segment space section.

**Tablespace**

Identifies the name of the tablespace

**File #**

Shows the sequential number of each file within the database

**Size (KB)**

Shows the file size in kilobytes. The total size of all files displays at the end of the column.

**Name**

Shows the full path name of each index database data file

**Free Space (KB)**

Shows the number of kilobytes not allocated to a table, rollback segment, or index

**Largest Free Ext (KB)**

Shows the number of kilobytes of the largest free (unused) extent for the table space. A database table is divided into regions called *extents*. Extents contain the actual indexing information that users enter.

**Rollback Segment**

Shows the name of the rollback segment

**Note**

---

Rollback segment field descriptions are the same as Part 2 and Part 3 field descriptions.

---

## Part 2 Description

Individual tables are described in part two, "Table Space," of the report.

### **Table**

Shows the name of the table

### **Exts Used**

Shows the total number of extents used by the table so far. It can display up to 9999 extents. If the database table has more than 9999 extents, this field displays the character string #####.

### **Max Exts**

Shows the maximum number of extents that can be allocated for the table. It can display up to 9999 extents. If the database table has more than 9999 extents, this field displays the character string #####.

### **Pct Incr**

Shows the percentage by which the size of a newly allocated extent is increased over its predecessor

### **Largest Ext (KB)**

Shows the size in kilobytes of the largest extent allocated for this table

### **Last Ext (KB)**

Shows the size in kilobytes of the last extent allocated for this table



**Total Space (KB)**

Shows the number of kilobytes allocated to the table

**Part 3 Description**

The third part, "Index Space," displays information on the index space for each table. Field descriptions are the same as those in Part 2 with the exception of Index.

**Index**

Shows the internal name of the individual index

## Sample Output from Microsoft SQL Server Environment

Sample output from an Microsoft SQL Server environment is shown below. The output differs slightly from spacerpt output on an Oracle system.

```

MS Command Prompt
SPACE REPORT
=====
SYSTEM NAME:  yamato                               Fri May 16 11:24:12 1997

Database Name          Size <KB>
-----
fnsys                  256000

Reserved KB    Data KB      Index KB      Unused KB
-----
1536           172           106           1258

TABLE SPACE

name            Rowtotal  Reserved KB  Data KB  Index KB  Unused KB
-----
doc_class_index 0          32           2        2         28
doctaba         0          64           2        6         56
document_class  0          48           2        4         42
folder          0          80           2        8         70
folder_contents 0          64           2        6         56
folder_tabs     0          48           2        4         42
index_cluster   0          32           2        2         28
menu            0          48           2        4         42
menu_items      0          32           2        2         28
no_cat_audit    0          16           2        0         14
sys_numbers     9          32           2        2         28
user_index      16         48           2        4         42
validation_tab  0          48           2        4         42
validation_tab_items 0 32 2 2 28
wqm001q000001  29         80           2        8         70
wqs_fields      3          32           2        2         28
wqs_idseed      1          32           2        2         28
wqs_queues      1          48           2        4         42
wqs_release     1          16           2        0         14
wqs_workspaces  1          32           2        2         28

INDEX SPACE

Table           Index                Reserved KB  Used KB  Unused KB
-----
doc_class_index di_docclassnumber    16          2        14
doctaba         da_archivedate       16          2        14
                da_deletedsdate     16          2        14
                da_docnumber         64          8        56
document_class  dc_docclassname      16          2        14
                dc_docclassnumber    16          2        14
folder          fl_archivedate       16          2        14
                fl_deletedsdate     16          2        14
                fl_foldername        16          2        14

```

Each section of the output is described below.

### **Part 1 Description**

Part 1 of the Space Report provides information about the entire index database.

#### **Database Name**

Shows the name of the database

#### **Size (KB)**

Shows the total size of the database in kilobytes

#### **Reserved KB**

Shows the number of kilobytes of reserved space in the database

#### **Data KB**

Shows the number of kilobytes used for storage of data

#### **Index KB**

Shows the number of kilobytes used for storage of indexes

#### **Unused KB**

Shows the number of kilobytes not currently used

### **Part 2 Description**

The second part of the Space Report, titled “Table Space,” provides information on individual tables.

**Name**

Shows the table name

**Rowtotal**

Shows the total number of rows in the table

**Reserved KB**

Shows the number of kilobytes of reserved space in the table

**Data KB**

Shows the number of kilobytes used by data in the table

**Index KB**

Shows the number of kilobytes used by index entries in the table

**Unused KB**

Shows the number of kilobytes not currently used in the table

**Part 3 Description**

The third part of the Space Report, titled "Index Space," displays information about index space use for each table.

**Table**

Shows the table name

**Index**

Shows the internal name of the individual index

**Reserved KB**

Shows the number of kilobytes of reserved space for the index

**Used KB**

Shows the number of kilobytes used to store the indexes

**Unused KB**

Shows the number of kilobytes not currently used

**Checklist**

Before you use spacerpt, be aware of the following:

- The spacerpt tool executes in read-only mode on the database so you can run it any time on any system.
- You must start the database.
- Output report for MS SQL Server is slightly different than that for an Oracle database.

**Procedure**

Enter **spacerpt** with appropriate parameters.

---

## Running on a Remote Oracle Server

If the Oracle databases are located on a remote Oracle server, you cannot run **spacerpt** directly because Oracle OS authentication prevents it. Instead, you need to modify two script files and login to sqlplus to get space information.

- 1 Copy the following two files from the `/fnsw/oracle` directory of the Image Services server to the `/fnsw/oracle` directory on the remote Oracle server:

**`/fnsw/oracle/spacerpt_summary.sql`**  
**`/fnsw/oracle/spacerpt_extended.sql`**

- 2 On the remote Oracle server, use your preferred text editor, such as **vi**, to modify these two files. The first line of each file is:

`/ as sysdba`

- 3 Remove this line from each file. Exit and save your changes.

Now you can run the **spacerpt** scripts successfully on the remote Oracle databases.

- 4 On the remote Oracle server, login to sqlplus to run the scripts:

**sqlplus**

- 5 When you're prompted, enter the user name `f_maint` and its password.
- 6 To run **spacerpt**, enter the following command at the sqlplus prompt:

**`@/fnsw/oracle/spacerpt_summary.sql`**

- 7 If you want to get a more detailed report, enter:

**`@/fnsw/oracle/spacerpt_extended.sql`**

---

# sqlplus

## Description

The sqlplus program provides commands to view tables in the Oracle-based FileNet index and WorkFlo databases. You can choose the columns and rows you want to view, from such tables as document\_class and user\_index. You can also view information on WorkFlo queues. In addition, sqlplus provides several administration commands that allow you to edit tables. However, use these editing commands only after contacting your service representative.

## Use

Use sqlplus when you need to examine tables in the index and WorkFlo databases. See the individual command descriptions for specific uses.

## Syntax

sqlplus

After entering the sqlplus command, the following prompts display.

```
corona(fnsw)/usr/fnsw> sqlplus

SQL*Plus: Release 3.3.2.0.0 - Production on Fri Mar 7
15:31:49 1997

Copyright (c) Oracle Corporation 1979, 1994. All rights
reserved.

Enter user-name:
Enter password:
```

Supply the appropriate user name and password to start sqlplus. When sqlplus starts, the prompt changes to **SQL>**. Enter sqlplus commands at this prompt.

To terminate the sqlplus program, enter any one of the following at the SQL prompt: **exit**, **quit**, or the Control+d key sequence.

## Commands

The following commands are a small subset of available sqlplus commands. The commands and their options listed are those most useful when you want to examine tables in FileNet databases. For more information, refer to the *Oracle SQL Language Reference* manual for your RDBMS system.

---

**Note** UPPERCASE letters indicate the minimum characters you must enter to run the command.

---

**Select <column names> FROM <table name> WHERE <expression>;**

Displays database tables. The following is a sample of information displayed from the Select command:

```
SQL> select f_docclassname, f_docclassnumber from
document_class;
```

F_DOCCLASSNAME	F_DOCCLASSNUMBER
-----	-----
CreditApps	1
LoanForm	2
Letter	3
Bills	4



You can limit the output from the Select command using the “where” statement. In the example below, the “where” statement lists only those document classes that begin with the letter L. For example, entering the command:

```
SQL> select f_docclassname from document_class where
      f_docclassname like 'L%';
```

displays the following output:

```
F_DOCCLASSNAME
-----
LoanForm
Letter
```

When using the “where” statement, verify that each column name you use is listed after the Select clause.

The “where” statement has the following arithmetic operators:

Operator	Definition
=	Equal to
!=	Not equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to

The “where” statement also has relational operators, as described below:

## And

Requires meeting both conditions joined by And. Be sure to select all columns in the Where clause.

**Where f\_batchsize > 10 And f\_pages > 3;**

## Or

Uses any condition joined by "or." Be sure to select all columns in the Where clause.

**Where f\_docclassname = 'CreditApps' Or f\_docclassnumber = 3;**

## Between

Displays rows with attributes in the range specified by "between"

**Where f\_docclassnumber Between 1 and 3;**

## In

Displays all specified columns, much like the = operator. Be sure to enclose words or numbers in parentheses. If you specify a word, enclose it in single quotes.

**Where f\_docclassname In ('LoanForm', 'Letter');**

## Not In

Does not display the specified columns, much like the != operator. Be sure to enclose words or numbers in parentheses. If you specify a word, enclose it in single quotes.

**Where f\_docclassname Not In ('CreditApps', 'Bills');**

---

## Like

Lists all columns with text strings that match those specified. The example below specifies displaying all columns beginning with the letter “L”. The % indicates that it is acceptable to display any text string occurring after the L.

**Where f\_docclassname Like ‘L%’;**

## Not Like

Lists columns that do *not* have the characteristics specified

**Where f\_docclassname Not Like ‘B%’ and f\_docclassname  
Not Like ‘C%’;**

## Null

Lists rows in which the specified column has no value (that is, “null”). (Use the Select clause to choose columns.)

**Where f\_pages Is Null;**

## Not Null

Lists rows in which the specified column has a value

**Where f\_pages Is Not Null;**

## DESCRIBE <table name>

Lists the names of all columns in the table. In addition, it indicates whether the column requires values (NULL or NOT NULL) and the column type.

For example, the following output displays for the `document_class` table:

```
SQL> describe document_class;
```

Name	Null?	Type
F_DOCCLASSNUMBER	NOT NULL	NUMBER (5)
F_DOCCLASSNAME	NOT NULL	CHAR (18)
F_DESCR		CHAR (30)
F_DOCTABLEID		CHAR (2)
F_PAGES		NUMBER (5)
F_BATCHSIZE		NUMBER (5)
F_PRIMARYPATH		NUMBER (5)
F_OPTIONALDE		CHAR (16)
F_BYPASSINDX		CHAR (1)
F_TABOUT		CHAR (1)
F_INDEXINGFORM		CHAR (14)
F_QUERYFORM		CHAR (14)

In the example above, the column names are either numeric (NUMBER) or character (CHAR) type. The number in parentheses indicates the maximum allowable digits (for a NUMBER type) or characters (for a CHAR type).

#### **ORDER BY <column name> Desc**

Specifies how to order rows based on the values of a given column name. This command orders numeric columns from smallest to largest values. If you specify Desc at the end of the command, then it orders values from largest to smallest. It orders strings alphabetically; if you specify Desc, it orders strings in inverse alphabetical order.

The following command displays the output in alphabetical order by document class name:

```
SQL> Select f_docclassname, f_docclassnumber from
document_class Order by f_docclassname;
F_DOCCLASSNAME          F_DOCCLASSNUMBER
-----
Bills                    4
CreditApps              1
Letter                  3
LoanForm                2
```

### TTITLE '<Title Name>'

Places a title at the top of computer screen output. This title appears above all column headings and is re-used for each table you display. You must change this title by issuing another TTITLE command or by clearing it using the TTITLE OFF command. If you type TTITLE without a title name, then the current title displays.

The following example uses two commands to display the output with the table heading, "Document Classes":

```
SQL> TTITLE 'Document Classes'
SQL> Select f_docclassname, f_docclassnumber from
document_class;
Document Classes

F_DOCCLASSNAME          F_DOCCLASSNUMBER
-----
CreditApps              1
LoanForm                2
Letter                  3
Bills                    4
```

**BTITLE 'Title Name'**

Places a title at the bottom of a computer screen. This title appears below all column data and is re-used for each table that you display. You must change this title by issuing another BTITLE command or by turning it off (BTITLE OFF). If you type BTITLE without a title name, the current title displays.

The following two commands used together display the output with the table heading at the bottom of the output:

```
SQL> BTITLE 'Listing of all document classes on Index
server'
SQL> Select f_docclassname, f_docclassnumber from
document_class;
F_DOCCLASSNAME          F_DOCCLASSNUMBER
-----
CreditApps                1
LoanForm                  2
Letter                    3
Bills                     4

Listing of all document classes on Index server
```

**SET PAGESIZE <number of lines per page>**

Sets the number of lines on a given page. For example, the following command limits the lines per page to 40.

```
SQL> set pagesize 40
```

This command is helpful for viewing large tables with many columns. If the page size is small (for example, 20 lines), the column headings reprint at the top of each page. To avoid this redundancy, set a larger page size.

**COLUMN <column name> HEADING '<Heading Name>' FORMAT A<number of spaces>**

Specifies a heading to use for a given column. You provide the name of the column to be replaced and the new heading name. You can also indicate the number of spaces to reserve for the column width of character headings; indicate this width after the 'A' option. The following example changes the column name for f\_docclassname:

```
SQL> Column f_docclassname heading 'Document Class  
Name';
```

As a result of the above command, the output of a Select command is modified to include the new column heading, as shown below:

```
Document Class Name  
-----  
CreditApps  
LoanForm  
Letter  
Bills
```

The program reuses this column name each time you select the f\_docclassname column.

**Change/<old text>/<new text>**

Edits the current command. <old text> is only the text that you want to change. <new text> is text that will replace the first occurrence of the <old text>. Remember to separate the command, <old text>, and <new text> with forward slashes (/).

Use the List command to view the current command.

If you type "f\_docclassname" incorrectly as in the following command

---

SQL> **Select docclass from document\_class;**

you can edit the line by typing:

SQL> **Change/docclass/f\_docclassname**

The updated command then displays:

**1\* select f\_docclassname from document\_class**

Then use the Run command (see below) to execute the updated command.

### List [<line number>]

Displays the current command, or optionally, one or more lines of the current command. The current command is that which executes if you use the Run command. For example, the following displays all lines of the command that would execute with Run:

SQL> **list**

**1 select f\_docclassname from document\_class**

**2\* where f\_docclassname like '1%'**

An asterisk after the line number indicates the current line. The current line is the line you would edit with the Change command.

To display one or more specific lines of the current command, use List with the <line number> option.



As an example, your command could contain the following 3 lines:

```
1 select f_docclassname,f_docclassnumber from document_class
2 where f_docclassname like 'L%'
3 order by f_docclassnumber;
```

Type “L1” at the SQL prompt and press return to display line number 1 of your command, as the following example and its output shows:

```
SQL> L1
1 select f_docclassname,f_docclassnumber from document_class
```

A typical use of the List command is to set the line number in preparation for editing with the Change command.

## RUN

Executes the current command. Use RUN to execute a command after you edit it with the Change or Input commands. Use the List command to view the current command.

## INPUT

Adds a new line to the current command. The next available line of the command displays after you enter **Inp**.

For example, if you want to enter a select command (such as **select f\_docclassname from document\_class**) and add a “where” statement (such as **where f\_docclassname like 'I%'**), you would follow these steps:

- 1 Enter Inp. The value 2 then displays.

```
SQL> select f_docclassname from document_class
SQL> Inp
2
```

The “2” indicates that you are about to enter the second line of the command.

- 2 Enter your new text after the 2:

```
2 where f_docclassname like 'I%';
```

- 3 Use the List command to display the full command:

```
SQL> list
select f_docclassname from document_class where f_
docclassname like 'I%'
```

### **SPOOL [<filename> | OFF]**

Indicates the file to which you want to save sqlplus output. When you have finished saving output to the file, type SPOOL OFF. To display the current spooling status, type SPOOL with no options.

## **Administrative Commands**

### **CAUTION**

Administrative commands alter tables within the index or WorkFlo databases. Do not use these commands unless authorized by your service representative.

---

**CREATE TABLE <table name> (<column name> <data type>);**

Adds a new table to the database. Specify the table name, column names, and the data type for each column. You must list column names and corresponding data types.

The following command creates a table called TestTable. The table has two columns. The first column is called Column\_1 and is a number. The second column is called Column\_2 and has text that can be up to 10 characters long.

```
SQL> Create Table TestTable (Column_1 number,Column_2
char(10));
```

**DROP TABLE <table name>;**

Deletes the specified table from the database

**CAUTION**

---

This command deletes all columns and rows of the specified table, as well as the table definition.

---

**INSERT INTO <table name> Values (<list>);**

Inserts values into a row of the specified table. You must include a separate Insert command for each row of a table. The list of row values should appear in the order of columns in the table. The following command inserts a row into TestTable:

```
SQL> Insert Into TestTable Values (1, first_row);
```

The value of "1" becomes data in the column named Column\_1, and the text string "first\_row" becomes data in Column\_2.

**UPDATE <table name> SET <expression> WHERE <condition>;**

---

Changes the specified table name. Use the SET and WHERE commands to indicate the values to change. The SET command indicates a column value to alter. The WHERE command indicates the condition that must be met for the column name to be altered.

For example, the following command updates all values in Column\_1 to “2” when the variable in Column\_2 is “second\_row”.

```
Update TestTable Set Column_1 = 2 Where Column_2 =  
'second_row';
```

### **DELETE FROM <table name> Where <expression>;**

Deletes specified rows in the specified table. The Where expression indicates the row to delete. For example, the following command deletes rows that have a Column\_1 value that is greater than 1.

```
SQL> Delete From TestTable Where Column_1 > 1;
```

Refer to the Select command for more information on the Where expression.

## **Checklist**

Before you use sqlplus, be aware of the following:

- You must not shut down the database (index or WorkFlo).
- You must specify a password to enter sqlplus. Contact your service representative for this password.
- You must log on to sqlplus as f\_cso, f\_maint, or f\_operator.
- You must log in as f\_maint to view the main Oracle table, sys.dba\_tables.
- Before you use any command that modifies the index or WorkFlo databases, contact your service representative.

## Procedure

- 1 Contact your service representative for a password.
- 2 Enter **sqlplus** to start the program and enter the correct user name (f\_maint, f\_operation, or f\_cso) and password at the prompts.
- 3 Enter commands at the SQL prompt (SQL>).

To view FileNet table names, go to the next step.

- 4 List the FileNet tables.
  - Enter the following command to list the names of all FileNet tables:

```
select owner, table_name from sys.dba_tables
       where owner = 'F_SW';
```

F\_SW must be in uppercase and enclosed in single quotes. All other parts of the command can be in lowercase. Output similar to the following displays:

OWNER	TABLE_NAME
-----	-----
F_SW	DOCTABA
F_SW	DOCUMENT_CLASS
F_SW	DOC_CLASS_INDEX
F_SW	FOLDER
F_SW	FOLDER_CONTENTS
F_SW	FOLDER_TABS
F_SW	INDEX_CLUSTER
F_SW	SYS_NUMBERS
F_SW	USER_INDEX

- To list tables for FolderView, specify the owner as F\_SQL:

---

```
select owner, table_name from sys.dba_tables where  
owner = 'F_SQI';
```

---

**Note** The table, sys.dba\_tables, contains information on all Oracle tables on the system.

---

Use other commands to display index or WorkFlo database information. Use the Describe command to list all columns for a particular table.

The format of WorkFlo tables is WQMsssQttttt, where sss is a 3-digit decimal number corresponding to the server ID, and ttttt is a 6-digit decimal number corresponding to the table ID. If the table ID is less than 6 digits, the value is padded with leading zeroes. As an example, a table might have a name of "WQM001Q001234."

---

**Note** A queue table name, once created, does not change by upgrading to a new release. Only queues created in IDMIS Release 3.1.0 or later have the table name format as described above. You can see both formats in your output if your tables were created in an earlier release.

---

If you need to alter database tables, contact your service representative for instructions.

- 5 Enter **exit** at the SQL prompt to terminate sqlplus.

# ssn

## Description

The ssn tool displays the system serial number.

## Use

Use the ssn tool to display the system serial number.

You can also use the System Attributes tab of the FileNet System Configuration Editor to display the system serial number and other system-related information.

## Syntax

```
ssn
```

## Sample Output

```
> ssn  
149718
```

## Procedure

No specific procedure is required.

## Related Topics

See the online help for FileNet configuration tools (fn\_edit for UNIX platforms or System Configuration Editor for Windows Server platforms) for a description of System Attributes.

## st\_msar\_convert

### Description

The st\_msar\_convert program starts an MSAR convert background job, which converts optical surfaces to MSAR surfaces. This tool maintains the natural collection of documents as they were on the optical surface. MSAR surfaces that have been converted from optical become read-only surfaces. You will not be able to write to the newly-created MSAR surface.

### Use

Use st\_msar\_convert to run the conversion of an optical surface to an MSAR surface in the background. This tool is for local surfaces only. Foreign surfaces must use the **Consolidate Media** option in Background Job Control.

### Syntax

```
st_msar_convert [-noupdatedocs] [-bothsides] [-trace] [-msar_lib  
<MSAR Library>] <surface_id>
```

- noupdatedocs** An optional parameter used to NOT update the docs table during the conversion.
- bothsides** An optional parameter used to indicate that both sides of the optical surface being converted should be copied.
- trace** An optional parameter that will log the process of the background job in the system log. The log lists each phase as it starts.



- msar\_lib <MSAR Library>** An optional parameter that allows the user to specify to which MSAR library to convert the surface (for example, a, b, c, d, ...).
- <surface\_id>** Identifies the optical surface to convert.
- The default settings of st\_msar\_convert are: updatedocs, not both-sides, trace turned off, and no preferred library. The optical surface ID is a required parameter.

## Sample Output

After the st\_msar\_convert program is started, you should get the following output:

```
MSAR convert started. Job # = <number>
```

## Checklist

None.

## Procedure

No specific procedure is required.

## Related Topics

None.

---

# stamp

## Description

Stamp sets or displays the system and developer release numbers, subsystem, release type, and SCR number for a set of files. You can use it on virtually all kinds of files: VAX a.out files, AIX a.out files, shell scripts, include files, libraries, and amorphous binary large objects (“blobs”) such as fonts and forms. If invoked with no options, stamp displays the system and developer release numbers, subsystem, SCR number, mode, and size for the named files. If you do not specify a file, it displays stamp information for a.out.

---

**Note** The stamp tool options that modify the stamp of a file require special permission levels and are not available to be run manually.

---

When displaying the stamp for a library or an ASCII file, stamp beeps if the file has changed since it was stamped.

## Use

Use stamp to display file characteristics (such as release number, developer, and Software Change Report (SCR) number).

## Syntax

The format for entering both system and development stamps is the same: “a.b.c.d.e” (e.g., 1.8.0.10.3). The unspecified components are set to zero.

```
stamp [<filename>] ... [<filename>]
```

**<filename>** Identifies one or more file names for which you are requesting information. If you do not specify a file name, stamp information for a.out displays.

## Sample Output

```
corona(fnsw)/usr/fnsw> stamp /fnsw/bin/TAPs
system 3.4.1.19(0) (userid 8, Thu Aug 29 06:46:32 1996)
developer 3.4.1.0.0 (userid 8, Thu Aug 29 06:46:06 1996)
SubSys: br, Rel_type: rel_aix, SCR#: 31528, mode: 100755, size: 67449
```

## Procedure

No specific procedure is required.

## Related Topics

[“stampro” on page 1236](#)

# stampro

## Description

The stampro program is a read-only version of the stamp program. It displays file characteristics (for example, release number, date of last update, and subsystem).

## Use

Use stampro to display file characteristics.

## Syntax

```
stampro [<filename>] ... [<filename>]
```

**<filename>** Identifies one or more file names for which you are requesting information. If you do not specify a file, the information for a.out, if present, displays.

## Sample Output

In the example below, stamp information is requested for TAPs.

```
corona(fnsw)/usr/fnsw> stamp /fnsw/bin/TAPs
system 3.4.1.19(0) (userid 8, Thu Aug 29 06:46:32 1996)
developer 3.4.1.0.0 (userid 8, Thu Aug 29 06:46:06 1996)
SubSys: br, Rel_type: rel_aix, SCR#: 31528, mode: 100755, size: 67449
```

## Checklist

Be aware that stampro is a read-only version of the stamp tool.

## Procedure

No specific procedure is required.

## Related Topics

**[“stamp” on page 1234](#)**

# stdoccpy

## Description

The stdoccpy tool copies information from one storage medium to another without first importing the storage medium. Because you do not have to first import the medium, using stdoccpy can save up to several hours when performing copy operations between storage media. Although Background Job Control (BJC) provides a document copy function, stdoccpy offers more options than BJC. You can use BJC functions to see the results of a copy performed with stdoccpy.

The phase of stdoccpy are:.

- **Phase 1:** Generate a document ID list from the source surface.
- **Phase 2:** Sort the document IDs from Phase 1 and eliminate duplicates.
- **Phase 3:** Copy the document image from the source surface to the target surface. New index values for documents copied to the target surface are generated using the update short descriptor or index database.
- **Phase 4:** Copy annotations from the source surface to the target surface. (This phase is optional.)
- **Phase 5:** Erase the source media. (This phase is optional.)

## Use

Use stdocopy to start a document copy process between storage media.

You can also use stdocopy to convert images from FileNet Banded Group 3 image format to TIFF Group 4 or CALS formats.

---

**Note** The stdocopy image conversion option is not available through the Background Job Control graphical user interface menu. You must use stdocopy from the command line to convert images.

---

If an error occurs during the conversion operation, stdocopy logs the error, copies the unconverted image to the target media surface, and continues. You must check the system error log for indications of conversion errors.

---

**Warning** The family entered **MUST** be the same type (primary/tranlog) and preferred to the same server as its alternate. If this rule is not followed, will result in primary and tranlog documents being separated across servers or residing on the wrong disk type. These conditions are not supported and will result in the loss of active use of the tranlog document.

---

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3)

The output of the **newindexes** and **erase** options of the stdoccpy tool can report some confusing information when being used with the FileNet P8 Content Federation Services functionality.

For the newindexes option, the index values are retrieved from DOCTABA rather than from the Content Engine (CE) catalog. DOCTABA does not always contain the latest index values, so older or null values could be used to populate the new media. For the erase option, the CE catalog entry could also be erased.

In the instances where index values on the new media will contain values from DOCTABA, these values could be old since updates on the Content Engine do not propagate back to Image Services

In the instances where documents are indexed on the Image Services system and the erase option is run, Index Services could create an export\_log entry to tell the FileNet P8 CFS Server for Image Services to delete the corresponding index entry on the Content Engine system.

Refer to the table below to determine when documents on the IS server will not have the index information associated with them (the **Ns**) because if index information does not exist in DOCTABA for an image, there is no way to notify the Content Engine system to delete its catalog information for that image.



<b>Document Activity</b>	<b>Index on CE</b>	<b>Index on IS</b>
New documents entered via Content Engine	Y	<b>N</b>
New documents entered via Image Services		
• Indexed on both IS and C	Y	Y
• Indexed only on CE	Y	<b>N</b>
Existing document images transferred to CE		
• Indexed on both IS and CE	Y	Y
• Index deleted from IS	Y	<b>N</b>

## Syntax

```
stdoccopy {--family <familyname> | --usecluster}
{--docidfile <docidfile> | --surface <surfaceid>}
[--findby {surface | db}] [--bothsides] [--nodatabase]
[--optdiskfile <optdiskfile>]
[--altsurf] [--oncopy] [--updatedb] [--newindexes]
[--writelist] [--copyannot] [--erase]
[--cformat {TIFF | CALS} --compression {G3 | G4}
--ctiled {TILED | BANDED | AUTOTILED} ]
```

You must specify either `--family <familyname>` or `--usecluster` to specify the location to which documents are to be copied. You must specify either `--docidfile <docidfile>` or `--surface <surfaceid>` to indicate which documents to copy. All other parameters are optional.

You must specify `--surface <surfaceid>` to use any of the following options:

- `--bothsides`
- `--findby`
- `--nodatabase`
- `--optdiskfile <optdiskfile>`

For a detailed description of all parameters and options, see [“Parameters and Options” on page 1242](#).

### Parameters and Options

**`--family <familyname>`**

Indicates the family to which documents are to be written

**–docidfile <docidfile>**

Identifies the file specified by <docidfile> that contains a list of the documents to copy. Each line in the file contains one document ID. The stdoccopy program uses the database to locate and copy each specified document.

**–surface <surfaceid>**

Copies documents on the specified <surfaceid>

**–findby {surface | db}**

Specifies the method by which the documents to copy are found. The –findby surface option uses the media surface directory to locate the documents; –findby db uses the database to locate the documents.

**Note**

---

If you do not specify –findby and the media is more recent than IMS Release 2.3 (which had no directory), the system uses the surface directory. Otherwise, it uses the database.

---

**–bothsides**

Copies both sides of <surfaceid>.

This option is mandatory for an optical disk.

This option does not apply to an MSAR surface and will be ignored.

**–nodatabase**

Indicates that no database exists. Also copies documents without checking for their presence in the DOCS table.

**Note** Using this option with `–altsurf` can result in an error because the program cannot use the database to find the alternate location of a document. This option must be used with the `–surface` and `–findby` options.

If you do not include the `–nodatabase` option on the command line, it cannot copy a document on the disk if it is not in the database.

---

### **–optdiskfile <optdiskfile>**

Indicates the name of the file on media that contains the documents. You must also specify the `–surface` and the `–findby` options.

### **–usecluster**

Uses cluster information to determine where to write out the documents. Writes out a document to the surface specified by the cluster map table (if the specified surface is not disabled) or to the current write surface of the family (if the cluster surface is disabled).

### **–altsurf**

Indicates which copy of the document to read on the first attempt. If you specify `–surface <surface id>`, then `–altsurf` reads the document from the surface not equal to `<surfaceid>`. If you do not specify `–surface <surfaceid>`, then `–altsurf` reads the document from `<surfaceid>`. If you specify `–docidfile <docidfile>`, then `–altsurf` reads the document from the transaction log surface. If you do not specify `–docidfile <docidfile>`, then `–altsurf` reads the document from the primary surface.

**–onecopy**

Indicates that there is only one copy of the document available to be read, and failure to read this one copy generates an error in the copy. If you do not specify **–onecopy** and the program fails to read the primary copy of the document, then the program reads the transaction log copy of the document. Likewise, if you do not specify **–onecopy** and the program fails to read the transaction log copy of the document, then the program reads the primary copy of the document.

**–updatedb**

Updates the database with the new document location.

**–newindexes**

Retrieves index information from the Index database (DOCTABA) rather than copying index information from the source media.

---

**Note** The Image Services 4.0 SP3 has some direct impact with this option of stdoccopy. For more information, see **[“Use with FileNet P8 Content Federation Services \(IS 4.0 SP3\)” on page 1240.](#)**

---

**–writelist**

Creates a list of the documents that were copied and the destination surface. The list name is the one of the following:

`/fnsw/local/logs/bkglogs/CpyDoc.nnn` for UNIX platforms

`<drive>:\fnsw_loc\logs\bkglogs\CpyDoc.nnn` for Windows Server

where *nnn* is the job number

**–copyannot**

Copies annotations. If you specify `–nodatabase` and `–findby surface`, the program copies annotations written on the input surface to the output surface. If you do specify either `–nodatabase` or `–findby`, for each document copied, the program writes annotations from the document's magnetic disk database to the destination media.

**–erase**

Erases the media after the copy of the document and removes the entries from the DOCTABA and DOCS tables. You can only use this option if **all** of the following are true:

- the medium is erasable.
- the `–surface`, `–bothsides`, and `–updatedb` options are specified
  - The `bothsides` option is ignored for MSAR surfaces.
- `–optdiskfile` option is **not** specified.

---

**Note** The Image Services 4.0 SP3 has some direct impact with this option of `stdoccopy`. For more information, see [\*\*“Use with FileNet P8 Content Federation Services \(IS 4.0 SP3\)” on page 1240.\*\*](#)

---

**–cformat {TIFF | CALS}**

Triggers a conversion from FileNet Banded Group 3 image format to either TIFF or CALS format.

**–ccompression {G3 | G4}**

Specifies image compression type. G4 is the default. G3 compression is invalid for CALS format.

**-ctiled {tiled | banded | autotiled}**

Selects the tiled image format. You can specify one of the following formats:

**tiled** specifies tiled images and is valid for TIFF images only.

**banded** specifies banded images.

**autotiled** chooses either tiled or banded automatically based on image size and density (number of pixels in a bitmap).

## Checklist

Before you use stdoccpy, be aware of the following:

- The program is directory independent. You can run stdoccpy from any working directory.
- Using stdoccpy can save you several hours when copying media because the media does not have to be imported first.
- You should carefully read the option descriptions to ensure that the resulting document copy meets your requirements.
- The stdoccpy tool offers more options than the document copy invoked in the Background Job Control (BJC). Use BJC to see the results of the document copy.
- Image file conversion is available only by using stdoccpy from the command line. You cannot use Background Job Control menu selections to convert image files.

## Procedure

Enter **stdoccpy** at the command line and specify the options, file names, and media surface IDs for the document to be copied.

## Related Topics

**[“stdocimp” on page 1249](#)**

See the “Storage Library Control” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see **[“Accessing IBM FileNet Documentation” on page 21](#)**.



# stdocimp

## Description

The stdocimp tool starts a document import job. Using this tool imports documents from optical media by updating the DOCS and DOCTABA tables with document descriptor information on the optical media.

The import document function enables you to update existing document location information. For example, stdocimp updates `surface_id_1` and `surface_id_2` with new surface id information from the document descriptor on the optical media to reflect the new primary and tranlog surfaces. However, the update cannot take place if the new surface is NULL.

If stdocimp encounters an error (such as mismatched document class) while importing a document into DOCTABA, stdocimp still corrects the inconsistency in the databases based on DOCTABA in the following manner:

- If the document exists in DOCTABA but not in DOCS prior to the import, stdocimp imports the document to the DOCS table.
- If the document exists in DOCS but not in DOCTABA prior to the import, stdocimp deletes the document from the DOCS table.

Although Background Job Control (BJC) also provides a document import function, stdocimp offers more options.

## Use

Use stdocimp to import only the documents that did not commit successfully using the standard import function in BJC.

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3)

The output of the stdocimp tool can report some confusing information when being used with the FileNet P8 Content Federation Services functionality. Since the function of stdocimp is to import documents from optical media by updating the DOCS and DOCTABA tables with document descriptor information on the optical media, a catalog entry is inserted into DOCTABA for each document. If the affected document class and indexes are mapped to Content Engine, then the new CE catalog entry is inserted into the CE database and contains the mapped index values from the media, this can cause a problem when the documents are stored on the Image Services system but indexed on the Content Engine system. In this scenario, the images could be erroneously catalogued twice in the CE catalog.

Additionally, in the instances where documents are indexed on the Image Services system and the stdocimp tool is run, existing DOCTABA indexes are overwritten with data from the media. This can overwrite the existing document properties in the CE catalog, if the properties are mapped. Refer to the table below to determine when documents on the IS server will not have the index information associated with them (the **Ns**).

<b>Document Activity</b>	<b>Index on CE</b>	<b>Index on IS</b>
New documents entered via Content Engine	Y	<b>N</b>
New documents entered via Image Services		
• Indexed on both IS and C	Y	Y
• Indexed only on CE	Y	<b>N</b>
Existing document images transferred to CE		
• Indexed on both IS and CE	Y	Y
• Index deleted from IS	Y	<b>N</b>

---

## Syntax

```
stdocimp [-nonexactclass] [-noinsertdoctaba]
[-security {none | doc} ] [-updatesnt]
[-noworkfloqueue] [-nodeleteddocs]
[-bothsides] [-highpriority]
[-optdiskfile <optdiskfile>]
[-skipcount <skipcount>]
[-docidfile <docidfile>] [-redo <redojobnum>]
[-ssn <ssn>] [ignoredeleteupdate] [overwritedoctaba] <surface_id>
```

The only required entry is <surface\_id>.

For a detailed description of all parameters and options, see [“Parameters and Options” on page 1252](#).

### Parameters and Options

#### **-nonexactclass**

Allows document imports even if document classes do not match exactly

#### **-noinsertdoctaba**

Does not insert documents into DOCTABA

---

**Note** You must also specify **-security none** when importing documents from non-existent classes.

---

This object is mutually exclusive with the -overwritedoctaba option.

**–security {none | doc}**

If you specify **none**, import document security is set to ANYONE; otherwise, security is set to the same value as the specified document on optical media. If you omit this option, it uses the security defined for the document class by default.

**–updatesnt**

Updates the scalar\_numbers table if the import document ID to be imported is greater than the next available document ID. This option permits you to reinsert documents into a database if the database on magnetic disk has been back-dated by a restore.

**–noworkfloqueue**

Does not insert the document into a WorkFlo queue if a queue is defined for the document class

**–nodeleteddocs**

Does not import deleted documents (that is, those documents that are not in either the DOCS nor DOCTABA tables). Using this option, you can insert records that are missing from either the MKF permanent database or the index database.

**–bothsides**

Imports both sides of media. This option is ignored if you specify –opt-diskfile.

**–highpriority**

Runs at high priority

**-optdiskfile <optdiskfile>**

Specifies the name of the document headers file on the media to be imported

**-skipcount <skipcount>**

Specifies the number of documents to skip before starting to import. For example, if <skipcount> is 100, the first 100 documents are not imported unless they are explicitly specified in the docidfile option.

**-docidfile <docidfile>**

Specifies the name of a magnetic disk file where each line consists of [SSN] doc\_ID. If a system serial number (SSN) is present, you must separate the SSN and document ID by spaces. If an SSN is not present, it imports a document for any SSN with the given document ID. If the import reassigns document IDs, the document IDs in the <docidfile> are IDs on optical media, not IDs on the importing system. If you specify both a <docidfile> and a skip count, this option imports a document that satisfies either condition.

**-redo <redjobnum>**

Imports the documents not imported by background job number <redjobnum>.

**-ssn <ssn>**

Indicates the ssn of a surface to be imported if ssn is from an incompatible system.

**-ignoredeleteupdate**

Allows short descriptors that have been marked Deleted by the stsur-fupdate tool to be imported to a system's Index database. Normally a short descriptor that's marked Deleted on an optical disk cannot be imported into a system's Index database.

However, if a value already exists in doctaba and the corresponding field for the same document on the optical disk is a null value, the doctaba value for that field will remain the same after the import. The doctaba value will NOT be set to null.

**-overwritedoctaba**

Allows stdocimp to replace index information in doctaba with matching index information from the short descriptors on the optical media.

However, if a value exists in the Index database and the corresponding value for the same document on the optical disk is null, the Index dataset value will NOT be overwritten.

This object is mutually exclusive with the -noinsertdoctaba option.

**<surface\_id>**

Identifies the ID of surface to be imported. Do not use this option if you use the -redo <redojobnum> option.

## Checklist

Before you use stdocimp, be aware of the following:

- The command is directory independent. Run it under any working directory.
- The stdocimp tool has more options than the normal document import invoked in Background Job Control (BJC).
- Entering the command with no options displays help text.
- After stdocimp starts, it reports the background job number assigned to the document import job. You can then use BJC to monitor the import process like any other document import job.

## Procedure

Enter **stdocimp** at the system command line using the options necessary to import the document.

## Related Topics

[\*\*“odrecover” on page 932\*\*](#)

[\*\*“stdocopy” on page 1238\*\*](#)

See “Importing Documents” in the “Background Job Control” chapter of the *System Administrator's Handbook*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)



# stmigrate

## Description

The stmigrate tool migrates documents from magnetic disk cache to storage media. You can migrate all unmigrated documents in cache or selectively migrate documents by document ID or by date. The actual migration consists of queuing write requests to perform asynchronously.

If you choose to migrate documents by document ID, you must create a file that contains the document IDs. You must list each document ID on a separate line in the file. The stmigrate tool reads each document ID in the file and uses the transient database information to locate the document to migrate.

If the transient database runs out of space due to a large number of write requests, stmigrate sleeps for ten minutes before retrying the remaining migration requests.

Before stmigrate generates a write request for the document to be migrated, it performs the following actions:

- Updates the document descriptors (page 0) with the latest index key from the index database
- Updates the document descriptors with the security names information from the index database
- Checks the family ID information in the index database to ensure document migration to the designated family media

You can start multiple stmigrate jobs, without errors, for the same unmigrated document. In addition, an stmigrate job can include one or more documents that other stmigrate jobs have specified for migration. In

either case, stmigrate generates only one write request per document. If a write requests exists, it doesn't generate additional write requests. However, each migration job's log entries reflect that the document was successfully migrated.

A document eligible for migration must exist in the unmigrated document list in the permanent database DOCS table with a status of unmigrated.

The program reports critical errors such as program failures that occur during processing in the system event log. It reports less serious errors, such as missing document pages or no matching document against the specified document file, only in the background job log, MiLog.<jobnum>, where <jobnum> is the background job number. The log file is located in the following directory:

`/fnsw/local/logs/bkglog/MiLog.<jobnum>` for UNIX platforms

`\fnsw_loc\logs\bkglog\MiLog.<jobnum>` for Windows platforms

The following example shows the entries when one document was successfully queued for migration and one (document ID 148657) was skipped due to error (message tuple 80,1,56):

```
Information for docmigrate job number 146 started at Fri Mar 28 20:08:07 1997
Job parameters:
Migrate: All Documents from File '/usr/fnsw/testmigr'.
Errors encountered:
Document id: 148657, ssn: 1717, error: <80,1,56>
End of job information:
Number of errors encountered: 1
Number of documents successfully queued for migrate: 1
Job completed at Fri Mar 28 20:08:09 1997
```

## Use

A cache-only system stores committed documents in magnetic disk cache. When you add a storage library or ODU to a cache-only system, you can use stmigrate to move cache-resident documents to your new storage media and free up magnetic disk space.

---

**Note** If you prefer to use a graphical user interface over a command line tool, use the Migrate Docs task from the New menu in Background Job Control.

---

## Syntax

```
stmigrate { -all | -file <filename> | -date <date> }
```

**-all** Migrates all documents in local page cache to storage media; mutually exclusive with **-file** and **-date** options

**-file <filename>** Migrates only those committed documents listed in the user-generated file specified by <filename>; mutually exclusive with **-all** and **-date** options. Each line in the file contains a single document ID. For each committed document specified, the program uses the database to locate and migrate the document.

**-date <date>** Migrates all unmigrated documents that have a create date older than the date specified by <date>, where <date> is in mm/dd/yyyy format; mutually exclusive with **-all** and **-file** options

## Checklist

Before you use stmigrate, be aware of the following:

- The stmigrate tool is directory independent. You can run it from any working directory.
- The stmigrate tool is the command line equivalent of running Migrate Docs from Background Job Control.
- Entering the stmigrate command alone displays command syntax.
- The document migrate background job is interruptible. For example, if your system crashes while stmigrate is running, the job restarts automatically when the FileNet software restarts.

## Procedure

- 1 Create the media family.

Before you use **stmigrate** to move documents from magnetic disk cache to optical media, you must create the media family. You can also change the media size if the optical media surfaces are not assigned. Refer to “Define/Update Media Families” online help in Database Maintenance for more information.

- 2 Start **stmigrate**.

Enter **stmigrate** at the system command line using the options necessary to migrate the documents you desire.

## Related Topics

See your *System Administrator's Handbook* for information about document migration.

Refer to Database Maintenance online help for Define/Update Media Families.

Refer to Background Job Control (BJC) online help for BJC tasks.

To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

# stsurfupdate

## Description

The stsurfupdate tool starts up a surface update background job that supports the updating of the short descriptors. It updates short descriptors (or user indexes) stored on optical surfaces by writing to a new location with index information taken from the Index Database.

## Use

Three log files are always generated in the background job directory, **/fnsw/local/logs/bkglog/** at the end of the surface update. They are:

- UpdLog.job#
- UpdProg.job#
- UpdSummary.job#

An additional log file, **UpdExcpt.job#**, could also be found in the same background job directory. However, this file only exists to record exception conditions where updated short descriptors cannot be generated (see [“Log Files” on page 1265](#)).

## Syntax

```
stsurfupdate -a | -f <surfidfile> | -s <surfid> | -p | -t [-n]  
[-d | -l <docidfile>] [-o] [-v] [-T]
```

**-a** Updates all in-box surfaces.

**-f <surfidfile>** Parses in a file full of surface ids for update, one surface id per line. The surfaceidfile name cannot start with a numeric character.

**-s <surfid>** Updates a single numeric surface id (for example, **-s 3004**).

**-p** Updates all primary in-box surfaces.

**-t** Updates all tranlog in-box surfaces.

The above five options are mutually exclusive.

**-n** Updates only if a mis-matched user numeric index value is detected in a short descriptor. The short descriptor numeric index is compared against the Index Database. If they mis-match, a new fixup short descriptor is generated. This new fixup short descriptor is taken from the Index Database. Note that besides updating the mis-matched user numeric index value, any other mis-matched indexes in the same descriptor are also updated into the optical. If this option is not used, by default, short descriptors whose indexes are not matching the indexes stored in Index Database are updated.

---

**Note** The **-n** option conserves space on the optical media and minimizes the number of updates.

---

**-d** Deletes short descriptors if not found in the Index and Permanent Database. This creates an updated short descriptor which is marked as "deleted." A short descriptor that is marked as "deleted" cannot be imported to a system unless the **-ignoredeleteupdate** option is used when running the stdocimp utility. The **-d** and the **-l** options are mutually exclusive.

**-l <docidfile>** Provides a list of document IDs which are updated if necessary. These are always interpreted as local documents. This is used as a filter. If a surface document ID is foreign and is in the local docs table, the original document ID is translated to a local document ID. This local docu-

ment ID is then compared with the document input list. The `-l` and the `-d` options are mutually exclusive.

- o** Forces to update foreign documents with a new short descriptor. If this option is used in conjunction with the `-d` option, the updated short descriptors are also marked as **deleted** for these foreign documents when the corresponding documents are not found in either the Index nor the Permanent Database. Note that a warning exception, **FB Overridden**, is counted when using this option. If this option is not selected, by default, no update short descriptor is created for foreign documents.

---

**Note** A WARNING exception “FB Overridden” is logged in the UpdExcpt.<job#> log file when using the `-o` option.

---

- v** Reports detailed mis-match index information in the log file (verbose mode).

---

**Tip** Depending on the other options you use, the `-v` option can generate an extremely large amount of output. If space is limited in the `bkglog` directory, use `-v` with caution.

If you decide to use the `-v` option, we recommend that you use it with the `-s <surfid>` option on one surface at a time.

---

- T** Test mode option. This option processes the surface but does not write to optical. Therefore, media/drive write compatibility is not checked. The surface is not marked with the update flag and no fixup short descriptors are written to optical. This option can be used to gather accurate statistics on updating.



## Log Files

After the stsurfupdate tool is run, the following three log files can be found in **/fnsw/local/logs/bkglog/**:

Updlog.000075 (000075 is the job number)

```
Information for surfupdate job number 75 started at 03/18/2005 10:15:05
Job parameters:
Update through all surfaces in the server: no
Verbose mode: no
Primary surface only: no
Tranlog surface only: no
Input surface id: 3070
Update only when mis-matched numeric index values are detected: no
Mark updated short descriptor as "deleted" if doc is not found in Index DB: no
Force update on foreign docs with mis-matched index values: no
Test mode: no

End of job information:
Number of errors encountered: 0
Number of surfaces scanned: 1
Number of surfaces skipped: 0
Number of short descriptors scanned from input media: 45
Number of short descriptors expected for update: 1
Last surface being processed: 3070
Job completed at 03/18/2005 10:15:37
```

## Portion of UpdProg.000075 (000075 is the job number)

```
SURFACE UPDATE PROGRESS REPORT : ibm28d2r (ssn: 10359) Filename : UpdProg.000075
=====

SURFACE
  3070 started scanning and updating at Fri Mar 18 10:15:05 2005
        index updated in document class: Erase_UDO2
                >>>: Description
  3070 scanned total of 45 doc headers, number of doc headers requires updates: 1
  3070 Short Desc File: 0, #Extent 1: Total allocated Sectors=3986 Total Used Sectors=1
  3070 Short Desc File: 1, #Extent 1: Total allocated Sectors=3986 Total Used Sectors=3
  3070 update expects using 1 sectors, 1825157 sectors is still available on surface
  3070 ended scanning and updating at Fri Mar 18 10:15:37 2005

SUMMARY REPORT:
        Date/Time started: Fri Mar 18 10:15:04 2005

        command invoked: stsurfupdate -s 3070
        Number of surfaces processed = 1
        Number of surfaces skipped = 0
.
.
.
```

## Portion of UpdSummary.000075 (000075 is the job number)

```
SURFACE UPDATE SUMMARY REPORT : ibm28d2r (ssn: 10359) Filename : UpdSummary.000075
=====

        Date/Time started: Fri Mar 18 10:15:04 2005

        command invoked: stsurfupdate -s 3070
        Number of surfaces processed = 1
        Number of surfaces skipped = 0
.
.
.
```

## sync\_write\_test

### Description

If your system uses a storage management system such as NAS (network-attached storage), SAN (storage area network), or HSM (Hierarchical Storage Management) for your MSAR directory (see [“Terms” on page 1273](#)), it **must** support synchronous writes. This requirement is not unusual. MSAR creation directories, any directories which are used for storing database files, and any directories used for storing CSM cache files must support synchronous writes. Otherwise, data can be lost. It is also a specific need of database vendors for storing database files.

---

#### CAUTION

If synchronous writes are not supported, surface file data, database, or cache information can be lost.

---

---

#### Note

For UNIX systems using NFS protocol, in the system file that is used for mounting the file system, verify that the **noac** option is set. It is also recommended that options, **vers=3** and **proto=tcp**, also be set. Refer to your NFS operating system manual.

---

The `sync_write_test` program can be used to determine if a given storage management system directory supports synchronous writes. This stand-alone program can also be used without other IS software. (In the IS 3.6 ESE release, this program was referred to as the `msar_sync_test` program.)

---

#### Note

Local SCSI magnetic disk drives and SAN devices always support synchronous writes. So, it is not necessary to run this tool on SCSI or SAN

devices. Local ATA magnetic disk storage devices (e.g., an IDE drive) don't always support synchronous writes, so they must be tested.

---

## Use

The `sync_write_test` program requires one parameter, the target (NAS, HSM, etc.) directory. A file will be created and opened with the platform-specific synchronous write through flags set (for UNIX: `O_SYNC`, for Windows: `FILE_FLAG_NO_BUFFERING` and `FILE_FLAG_WRITE_THROUGH`) in the directory. These flags are used by the IS software for opening MSAR, database, and cache files. The program operates as follows:

- The file will be written to in 1024 byte block size indefinitely. Each block will contain a block number repeatedly stored in long words and the blocks will be numbered sequentially.

For example, the first block written will contain 256 long words with the value of 1, and the second block written will contain 256 long word with the value of 2, and so on. This information will be useful it ever becomes necessary to run traces on the data that is coming over the network or storage device.

- Once the program has written 10k of data, the user will be prompted to disconnect the target storage system.

---

### Note

There is a 1 GB limit for the target file. If the file exceeds this limit during testing, the program will terminate and the `sync_write_test` will need to be re-run.

---

- While the program is running, perform a soft disconnect to break the connection. For example, log onto the remote storage device

and unexport the share. This will cause the sync\_write\_test program to fail on the next write to the target file. A message will be written to the screen indicating the last successfully written sector. The user will also be prompted to reconnect the target storage system.

- Next, the target connection should be restored. Once this is done, the sync\_write\_test program will start the verification phase where the content of the data is verified and ultimately indicate if the system passes the test.
- If the system passes, the target test file will be deleted. This indicates that the target directory does support synchronous writes and can be used as an MSAR creation directory or used to store FileNet database and cache files. If the system does not pass, the target file will not be deleted. This indicates that the target directory does not support synchronous writes and cannot be used as an MSAR creation directory or as storage for FileNet database and cache files.
- For Windows 2003 systems only: The sectors are written twice because NTFS does not always indicate failure the first time. The IS software rewrites the last 1024 bytes of each buffer written to MSAR.
- This test should be performed five times.

## Syntax

```
sync_write_test <Target_directory>
```

**<Target\_directory>** Target storage file directory.

## Procedure

- 1 Enter the following command:

```
sync_write_test <Target_directory>
```

**Note** In the examples below, we use **I:\msar\_stor** for the <Target\_directory>, and the target\_sync\_file has the unique process ID appended, which is 2952.

After you enter the command above, you will see a screen similar to the following.

```
D:\fns\bin>sync_write_test I:\msar_stor
Creating file I:\msar_stor\target_sync_file2952
```

- 2 Once the program has written 10k of data, you will be prompted to disconnect the target storage system.

```
D:\fns\bin>sync_write_test I:\msar_stor
Creating file I:\msar_stor\target_sync_file2952
Disconnect the target storage device. (10k written so far)
Disconnect the target storage device. (100k written so far)
Disconnect the target storage device. (1000k written so far)
```

While the test continues to run, perform a soft disconnect to break the connection. For example, log onto the remote storage device and un-export the share. This will cause the sync\_write\_test program to fail on the next write to the target file.

---

**Note** Additionally, if the sync\_write\_test is allowed to run and the target file exceeds 1073741824 bytes, the sync\_write\_test program will terminate. If this occurs, you will need to re-run the test.

---

**3.** The screen will display the following information.

```
After disconnection, Expected Failure occurred while writing to:
I:\msar_stor\targer_sync_file2952 file (Target) system error=1167
The target file has been closed
Target file 'I:\msar_stor\targer_sync_file2952' should be at least 21603k.
The target file has been closed
Re-connect the target storage device.
After you have re-connected. Type 'y' to continue or 'a' to abort.
```

You will also be prompted to reconnect the target storage system as seen in the example above.

**4** Restore the connection to the target system and answer **y** to the prompt. Once this is done, the sync\_write\_test program will start the verification phase where the content of the data is verified and ultimately indicate if the system passes the test.

```
Opening file I:\msar_stor\targer_sync_file2952
Verifying the data (one dot for every 100k verified):
.....
.....
.....
Last block verified=21603k

Passed SYNC test!!!

Target file has been deleted: I:\msar_stor\targer_sync_file2952
```

If the system passes, a **Passed Sync test!!!** notification will be displayed on the screen and the target test file will be deleted. This indicates that the target directory does support synchronous writes and can be used as an MSAR creation directory or used to store FileNet database and cache files.

If the file is smaller than the last successful write, the system will not pass. This will be evident by a screen similar to the following:

```
Target file has not been deleted: I:\msar_stor\targer_sync_file2952
This was done because the storage device failed the SYNC test.
```

In this case, the target file will not be deleted. This failure indicates that the target directory does not support synchronous writes and cannot be used as an MSAR creation directory or used to store FileNet database and cache files. If this occurs, you need to contact your storage provider for assistance.

**5** Repeat Steps 1 through 5 four more times.

It is necessary to do the test 5 times successfully because there can be a false positive result. If the test fails even once during the 5 tests, the system has **NOT** passed the test.

**Note**

---

In some cases, the operating system or storage system can flush its buffers before the disconnection is complete. This means there is a possibility of having a false positive result. It might be necessary to consult with the storage provider about the specific way to disconnect a storage system.

---



## Terms

- NAS** Network-attached storage (NAS) is hard disk storage that is set up with its own network address rather than being attached to the department computer that is serving applications to a network's workstation users. By removing storage access and its management from the department server, both application programming and files can be served faster because they are not competing for the same processor resources. The network-attached storage device is attached to a local area network (typically, an Ethernet network) and assigned an IP address. File requests are mapped by the main server to the NAS file server
- SAN** A storage area network (SAN) is a high-speed special-purpose network (or subnetwork) that interconnects different kinds of data storage devices with associated data servers on behalf of a larger network of users. SAN Storage Devices can be connected to the server via a network or locally attached.
- HSM** Hierarchical Storage Management (HSM) is policy-based management of file backup and archiving in a way that uses storage devices economically and without the user needing to be aware of when files are being retrieved from backup storage media. Although HSM can be implemented on a standalone system, it is more frequently used in the distributed network of an enterprise. The hierarchy represents different types of storage media, such as redundant array of independent disks systems, optical storage, or tape, each type representing a different level of cost and speed of retrieval when access is needed.

## Related Topics

**[“test\\_raw\\_partition” on page 1284](#)**

If you are interested in looking at the source files for the sync\_write\_test, they can be found in `/fnsw/etc/sample/sync_write_test.c`.

## sys\_log

### Description

The sys\_log program appends an entire string of message text to the system event log. Quotation marks around the message text are optional, as is the specification of an error tuple for the message. All of the following examples are valid:

```
sys_log 181,1,1 "this is an error"
```

```
sys_log 181,1,1 this is an error
```

```
sys_log "this is an error"
```

```
sys_log this is an error
```

### Use

You can invoke sys\_log using a script to add a message of your own to the event log. You can optionally assign a three-part error tuple to your message or just specify the message text to write to the event log.

## Syntax

```
sys_log [<error_tuple>] message [-p]
```

**<error\_tuple>** Specifies the three-part message number in the following format:

```
cat,func,num
```

where

**cat** specifies the category portion of the error tuple

**func** specifies the function portion of the error tuple

**num** specifies the number portion of the error tuple

**message** Defines message text to write to the event log; maximum string length is 1024 bytes; quotes are optional

**-p** Displays the message at the active window and writes the message text to the event log

## Example

```
sys_log 171,1,16 This is an error
```

## Procedure

No specific procedure is required.

# tapeser

## Description

The tapeser tool unconditionally writes a serial number (also called a volume label) to a tape. The serial number consists of six non-blank alphanumeric characters (letters, numbers, and spaces in any combination) or six spaces enclosed in quotes. If your serial number is less than six characters, pad the left-most digits with spaces or zeros (for example, 003456).

### CAUTION

The tapeser tool writes a label and two tape marks at the beginning of the tape, **effectively erasing all the data on the tape**.

You specify the serial number and an optional device. The device parameter specifies the density of the tape to which you are writing. If you do not specify a device, the program obtains the default device for your specific system from tape configuration files.

### UNIX

For UNIX systems, the tape configuration file is `/fnsw/local/sd/1/tape-config`. The contents of this file resembles the following:

```
TAPE_CONFIG 10
backup tape1:corona:FileNet
tape1:corona:FileNet,3,0,(1,/dev/rmt0.4,/dev/rmt0.5),\
(2,/dev/rmt0,/dev/rmt0.1)
```

### AIX

For example, for AIX systems, the default is `/dev/rmt0`, and the tape density for this device is high density.

## Use

Use tapeser to write a volume label with a serial number to a tape.

When attempting to read new blank tapes, the system scans the entire tape while checking for data. This is time-consuming and unnecessary. To avoid a complete scan of a blank tape on an initial read, use tapeser to write a volume label and two tape marks to new tapes.

## Syntax

```
tapeser -s <serialno> [-d <device>]
```

- s <serialno>** Specifies the serial number to assign to the tape  
The serial number must be one to six characters or spaces.
- d <device>** Specifies the tape device to which the volume label is written  
If not specified, the default device for your specific system is obtained from tape configuration files.

## Examples

The examples below are tapeser commands to label new tapes.

The following command assigns a serial number of 12345 to a tape volume:

```
tapeser -s 12345
```

You can label a tape with a serial number consisting of six spaces. This helps when labeling tapes for unattended online backups. To assign a label of six spaces to a tape, verify that you enclose the six spaces within quote marks as represented below:

```
tapeser -s "      " -d /dev/rmt/0m
```

You can control the use of tapes by assigning a particular character string as the volume label. For example, you can assign a day of the week as the volume label to control which tape to use on a particular day. To label a tape with the volume label "Monday," use the following command:

```
tapeser -s MONDAY -d /dev/rmt/0m
```

## Checklist

Before you use tapeser, be aware of the following:

- tapeser, in effect, erases all existing data on the tape.
- If you use spaces to label a tape, enclose the spaces in double quotes.

## Procedure

No specific procedure is required.

# TAP\_tool

## Description

TAP\_tool is a utility you can use to monitor and manage Image Services tape functions.

The report option of TAP\_tool creates a report on the internal state of the TAPt shared library (abstract). The output report also displays process IDs for workers, the remaining software-defined capacity of the current or last tape written, and identifies the default drive.

The clean option of TAP\_tool closes a client process if the process is not properly closed. If you attempt to close an active client, an error message displays indicating the client is still active.

You can also specify an option to collect debugging information. Debugging values are provided, when necessary, by your service representative.

## Use

Use TAP\_tool to create a report on the internal state of the TAPt shared library, close a client process if the client has not been closed using the proper procedure, and dynamically set the debug variable.

Use `-r` to obtain the process ID of a dead process. When instructed to do so by your service representative, use `-c` and the process ID of a dead process to close a client process.



## Syntax

TAP\_tool [-r] [-c <pid>] [-d <value>]

- r** Displays a report on the internal state of TAPt
- c<pid>** Closes a client if it has not been closed using the proper procedure; also cleans out “dead” TAPt clients
- d<value>** Dynamically sets the TAP\_debug variable to <value>  
Values are available from your service representative.

## Sample Output

The following sample is output from TAP\_tool with the report option:

```
$ TAP_tool -r
1 Service. Drive Descriptions:
tapel:costa3:FileNet type 3
  1 rew: /dev/rmt0.4      no rew: /dev/rmt0.5
* 2 rew: /dev/rmt0       no rew: /dev/rmt0.1
tape state: tapeCLOSED
keep_going 0
tape is closed (normal)
job ` `, reel 0
file ` `, reel 0
userblockseq = 0, tapehighwater = 0, blockswaiting = 0
tape capacity remaining = 0 MB
```

The default tape drive is identified with an asterisk (\*) in column 1 of the report. The “tape capacity remaining” field at the end of the report displays the remaining software-defined capacity (in megabytes) of the current or last tape written.

## Checklist

Before you use TAP\_tool, be aware of the following:

- You must consult with your service representative before running TAP\_tool with the `-c` or `-d` options.
- Since a system process automatically cleans up “dead” clients, you should use the `-c` flag only in abnormal situations and at the direction of your service representative.

## Procedure

- 1 If you need to set the debugging variable, call your service representative.
- 2 Start TAP\_tool.

At the command line, enter **TAP\_tool** and the options required to create a report, close a client, or set the debug variable.

## Related Topics

Depending on the method you use to back up and restore the Image Services software, refer to one of the following documents for information about labeling backup tapes:

- *System Administrator's Companion for UNIX*
- *System Administrator's Companion for Windows Server*
- *Enterprise Backup/Restore User's Guide*

To download IBM FileNet documentation from the IBM support page, see [\*\*"Accessing IBM FileNet Documentation" on page 21.\*\*](#)

## test\_raw\_partition

### Description

The **test\_raw\_partition** tool tests whether a UNIX platform supports raw partitions and synchronous writes to partitions and files. On UNIX platforms, the Image Services CSM page cache and the MKF transient, permanent, and security databases are stored in raw partitions. (Raw partitions are called logical volumes on some platforms.)

If your system uses a storage management system such as NAS (network-attached storage), SAN (storage area network), or HSM (Hierarchical Storage Management) for your MSAR directory (see [“Terms” on page 1291](#)), it **must** support synchronous writes.

Any directories that are used for storing database files, any directories used for storing CSM cache files, and any MSAR creation directories must support synchronous writes. It is also a specific requirement of all database management system vendors for storing database files. The file systems of all IS-supported platforms are high performance, and all support synchronous writes.

---

### CAUTION

If synchronous writes are not supported, surface file data, database, or cache information could be lost in the event of a system crash or power failure.

---

Although a platform supports synchronous writes, it might not be actually performing synchronous writes. Caching can occur at the OS level, the NAS/SAN level, and the disk drive level. Synchronous writes do not occur unless all such caching is disabled, so if the IS page cache or any of the MKF databases are placed on such a device, the caching must be disabled or the cache must be battery-backed-up.

## How test\_raw\_partition Works

The test\_raw\_partition tool counts the number of writes it can perform to the same sector of a file or partition within a fixed amount of time (20 seconds). If it's performing synchronous writes, the disk has to make one complete revolution between writes. Assuming one disk revolution per write, test\_raw\_partition calculates the speeds of the disk in RPM (revolutions per minute).

The RPM number calculation is the number of writes to the same sector divided by the time interval in seconds times 60 seconds per minute. (If the data is being cached, the speed is limited only by the speed of the CPU or memory, and the writes are asynchronous.)

Typically, server disks spin at rates of 3,600 to 7,500 revolutions per minute (RPM). The fastest spinning magnetic disks currently available might go as fast as 15,000 RPM. To allow for hardware improvements over the next number of years, test\_raw\_partitions considers 18,000 RPM to be achievable by a magnetic disk.

The test-raw-partition tool determines whether or not synchronous writes are being performed by performing these steps:

- 1** First, test\_raw\_partition computes the equivalent RPM number of the file or raw partition being tested based on the write rate described earlier.
- 2** Then test\_raw\_partition creates an ordinary file with asynchronous writes and computes the equivalent RPM number.
- 3** Finally, test\_raw\_partition compares the two RPM numbers:

If the RPM number of the file or raw partition being tested is less than the minimum of 18,000 RPM, and one half the RPM number of a file

with asynchronous writes, then the RPM number is assumed to be low enough to be achieved by synchronous writes to a magnetic disk. In this case, test\_raw\_partition passes the file or raw partition being tested.

If the RPM number of the file or raw partition being tested is equal to or greater than the minimum of 18,000, and one fourth the RPM number of a file with asynchronous writes, the file or raw partition being tested fails the test for synchronous writes.

## Use

Use the test\_raw\_partition tool to find out if a particular platform supports raw partitions and is performing synchronous writes.

The test\_raw\_partition tool can perform these tests:

- Create and test raw partitions
- Test existing raw partitions
- Create and test a file

Run test\_raw\_partitions as a user with **root** privileges to avoid permission problems.

## Syntax

```
test_raw_partition {vg=<vol_grp> | par=<partition_name> |  
file=<file_name>}
```

The vg, par, and file options are mutually exclusive.

Run test\_raw\_partition with no options to display the online help.

## Procedure

Run test\_raw\_partition as a user with **root** privileges.

### OPTION 1: Test a New Partition

Create a volume group for the test partition, or choose an existing volume group. Under the specified volume group, test\_raw\_partitions creates a test partition, runs a read/write test on the test partition, and then removes the test partition.

```
test_raw_partition vg=<volume_group_name>
```

The test\_raw\_partition tool creates a 16 MB (16\*1024\*1024) test partition fn\_test\_partn within the specified volume group, runs the read/write test on it, and then removes the test partition.

The test\_raw\_partition tool also creates a small temporary file, /tmp/fn\_file\_test, and automatically removes it when the test is finished.

### OPTION 2: Test an Existing Partition

You must manually create a test partition mapped to the devices of interest before running this program. The partition must be at least 16 MB (16\*1024\*1024) in size.

### CAUTION

---

There must be NO valuable data in the test partition as DATA WILL BE DESTROYED by the disk write test.

---

```
test_raw_partition par=<full_path_name_of_the_test_partition>
```

The test\_raw\_partition tool creates a small temporary file, /tmp/fn\_file\_test, and automatically removes it when the test is finished. When test\_raw\_partition is done, you must remove the test partition manually.

### OPTION 3: Test an Existing Directory

Some platforms support synchronous writes, but not in certain file systems or directories. For example, some platforms do not support synchronous writes to files in /tmp. In such cases, no errors or warnings are produced.

```
test_raw_partition file=<full_path_name_of_test_file>
```

The test\_raw\_partition tool creates the specified test file, runs the synchronous write test, and then removes the test file. The test\_raw\_partition tool also creates a small temporary file, /tmp/fn\_file\_test, and automatically removes it when the test is finished.

### Sample Output

#### Success Case

The following command was run to test whether files in the /msar\_new directory support synchronous writes:

```
test_raw_partition file=/msar_new/abc
```



The following output was produced:

```
PASS: partition opened for read/write
Starting 10 second write test on /tmp/fn_file_test
Performed 562816 writes to same sector in 10 secs
Corresponds to 3376896 RPM
Starting 20 second synchronous write test on /msar_new/abc
Performed 1520 writes to same sector in 20 secs
Corresponds to 4560 RPM
PASS: synchronous write test (RPM < min(18000, 844224))
PASS: open/close/read/sync-write test
PASS: ALL TESTS PASSED
DONE
```

### Explanation

The file `/tmp/fn_file_test` was created and tested for synchronous writes, then deleted. That file was created such that the writes would be asynchronous, so its RPM number was very high (3,376,896 RPM). There are no magnetic disks that spin anywhere near that fast.

Then the test file `/msar_new_/abc` was created and tested. The file was created such that the writes were supposed to be synchronous. The RPM number was 4560 RPM, which falls within the normal range for magnetic disks.

The minimum of 18,000 RPM and one fourth the asynchronous file RPM number is  $\min(18000, 0.25 * 3376896)$ , which is equal to  $\min(18000, 844224)$ , or 18000. Since 4560 is less than 18000, synchronous writes were performed to the test file `/msar_new/abc`. Therefore, the test file `/msar_new/abc` passed the synchronous write test.

---

## Failure Case

The following command was run on a system different from the previous example to test whether files in the /msar directory support synchronous writes:

**test\_raw\_partition file=/msar/hello**

The following output was produced:

```
PASS: partition opened for read/write
Starting 10 second write test on /tmp/fn_file_test
Performed 713621 writes to same sector in 10 secs
Corresponds to 4281726 RPM
Starting 20 second synchronous write test on /msar/hello
Performed 44804 writes to same sector in 20 secs
Corresponds to 134412 RPM
FAIL: writes not synchronous (or RPM > min(18000, 1070430))
FAIL: open/close/read/sync-write test
FAIL: SOME TESTS FAILED.
DONE
```

## Explanation

The file /tmp/fn\_file\_test was created, as before, to get the RPM number for files using asynchronous writes (that is, the OS performs caching). The computed number was 4,281,726 RPM, far faster than any magnetic disk.

Then the test file /msar/hello was created and tested. The computed number was 134,412 RPM, far faster than any magnetic disk. However, since the number was much smaller than for the /tmp/fn\_file\_test file, it implies that the "disk" was much slower. In fact, the equivalent FPM number is so low (but still faster than any magnetic disk) that it might be a SAN or NAS device that performs caching.

Since 134,412 is less than the minimum of 18,000 and 1,070,430 (one fourth the write of files with asynchronous writes), the test file /msar/hello failed the synchronous write test.

## Terms

- NAS** Network-attached storage (NAS) is hard disk storage that is set up with its own network address rather than being attached to the department computer that is serving applications to a network's workstation users. By removing storage access and its management from the department server, both application programming and files can be served faster because they are not competing for the same processor resources. The network-attached storage device is attached to a local area network (typically, an Ethernet network) and assigned an IP address. File requests are mapped by the main server to the NAS file server
- SAN** A storage area network (SAN) is a high-speed special-purpose network (or subnetwork) that interconnects different kinds of data storage devices with associated data servers on behalf of a larger network of users. SAN Storage Devices can be connected to the server via a network or locally attached.
- HSM** Hierarchical Storage Management (HSM) is policy-based management of file backup and archiving in a way that uses storage devices economically and without the user needing to be aware of when files are being retrieved from backup storage media. Although HSM can be implemented on a standalone system, it is more frequently used in the distributed network of an enterprise. The hierarchy represents different types of storage media, such as redundant array of independent disks systems, optical storage, or tape, each type representing a different level of cost and speed of retrieval when access is needed.

## Related Topics

### [“sync\\_write\\_test” on page 1267](#)

If you are interested in looking at the source files for the sync\_write\_test, they can be found in /fnsw/etc/sample/**sync\_write\_test.c**.

## TLIB\_tool

TLIB\_tool controls Exabyte tape library functions. The TLIB\_tool allows you to initialize or reset the tape library, lock or unlock the front access door, acquire inventory information about the tape library, move or load or unload media, position the tape library arm, eject media, and search or move media by bar code label.

For additional information, refer to Appendix A of the *Enterprise Backup/Restore User's Guide*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## uncompressdir

**HPUX****AIX**

The uncompressdir tool is available only on an Image Services for AIX/6000 or Image Services for HP-UX. It does not run on Image Services for the Solaris Operating Environment or Image Services for Windows Server.

### Description

The uncompressdir tool restores a compressed file to its original uncompressed format. The uncompressdir tool calls the uncompress program, passing flags as options. The uncompress program recursively descends each specified directory. It examines each file for an extension of .Z, which indicates the file has been compressed with the compressdir tool. If you do not specify any directories, it uncompresses all applicable files starting with the current directory.

**Note**

---

The uncompressdir tool is not supported in Image Services for Windows Server or Image Services for the Solaris Operating Environment.

---

### Use

Use uncompressdir to restore a file compressed with compressdir to its original uncompressed format.

## Syntax

```
uncompressdir [ flags ] [ directory ... ]
```

- flags** See “Flags” below.
- directory** One or more directories that contain files you want to uncompress. If you do not specify a directory, the program uncompresses all applicable files starting with the current directory.

## Flags

- c** Writes to the standard output without changing the files (concatenate)
- q** Runs in quiet mode: without displaying progress messages
- v** Runs in verbose mode: displays progress messages with the expansion (in percent) of each file being uncompressed
- V** Displays the version and selected options

## Example

The following example requests the program to uncompress all compressed files in the /tmp/dumps directory.

```
corona(root)/> uncompressdir /tmp/dumps
```

## Checklist

Before you use uncompressdir, be aware of the following:

- If the input file is a directory, it remains unchanged.
- If you do not specify any directories, the program uncompresses all applicable files in all directories, starting with the current directory.
- The uncompressdir tool is not supported in Image Services for Windows Server or Image Services for the Solaris Operating Environment.

## Procedure

No specific procedure is required.

## Related Topics

**[“compressdir” on page 160](#)**



---

# vi

## Description

The vi (view log) tool displays entries in the FileNet event log on a local or remote host server. If you specify a remote host, it must be running TM\_daemon. However, if you specify the local host, vi does not require the host to run TM\_daemon.

## Use

Use vi when you want to examine the entries in the event log of a particular server.

---

**Note** You can also use the Event Logs option from the Task Manager Monitor menu to examine the entries in the FileNet event log. See the Task Manager online help or the “System Management” chapter of your *System Administrator’s Handbook* for details. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

---

## Syntax

```
vi [-t] [-h <hostname>] [-v <viewer>] [-d]
```

- t** Displays entries as the program writes them to the event log (tail mode)
- h <hostname>** Retrieves the event log from the specified host. Default is the local host name. The specified host server must run TM\_daemon.
- v <viewer>** Uses program specified by <viewer> for viewing the event log

- 
- d Displays contents of TM\_daemon log file. This option is valid only when specified for the local host.

## Sample Output

---

```
corona(root)/> vl
96/11/20 10:15:29.295 211,1,17 <fnsw> TM_daemon (3961) ...
An SNMP trap was issued for this error with trap code d30001, trap severity '1'

96/11/20 10:15:29.553 211,1,11 <fnsw> TM_daemon (3961) ...
syslog semaphore has been re-established.

96/11/20 10:15:29.953 211,1,11 <fnsw> TM_daemon (3961) ...
Terminating FileNet software...

96/11/20 10:15:30.550 211,1,11 <fnsw> TM_daemon (3961) ...
Shutting down security database...

96/11/20 10:15:31.327 211,1,11 <fnsw> TM_daemon (3961) ...
Shutting down permanent database...

96/11/20 10:15:31.712 211,1,11 <fnsw> TM_daemon (3961) ...
Shutting down transient database...

96/11/20 10:15:32.639 211,1,11 <fnsw> TM_daemon (3961) ...
Terminating processes...

96/11/20 10:15:50.692 211,1,11 <fnsw> TM_daemon (3961) ...
Shutting down index database...

96/11/20 10:15:54.729 <fnsw> /fnsw/bin/sys_log (4008) ...
Shutdown Oracle data base...

96/11/20 10:16:04.714 211,1,11 <fnsw> TM_daemon (3961) ...
initiate_procs: Processes terminated successfully on a 'stop' command

96/11/20 10:16:04.739 211,1,11 <fnsw> TM_daemon (3961) ...
Termination of FileNet software completed.
```

---

## Checklist

Before you use vl, be aware of the following:

- The vl tool displays the event log of the server on which you are running.
- You can also use the Event Log option of the Task Manager Monitor menu to view FileNet event logs.

## Procedure

No specific procedure is required.

## Related Topics

See the Task Manager online help or the “System Management” chapter of your *System Administrator’s Handbook* for information about the Task Manager Monitor. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

## WALSEC\_debug

### Description

The WALSEC\_debug tool enables support personnel to trace execution within the Security Services subsystem of the FileNet IS Toolkit formerly known as the WorkFlo Application Language (WAL) Application Programming Interface (API). This tool provides different trace levels, based on specified command options.

FileNet software defines classes of functions within shared libraries (also called “abstracts”). Each function, called an “entrypoint,” can be a member of exactly zero or one of the following classes: entry, ventry, reentry, sentry, and ientry. If it is not a member of any of these classes, the function is referred to within the WALSEC\_debug program as a “uentry” for “unexposed entry.”

You can use the WALSEC\_debug tool to enable or disable trace debugging for each entrypoint class within a given shared library or set of shared libraries. The debugging system for IS Toolkit contains two shared libraries (SEC and SECr). For example, you can turn on or off debugging for entries in SEC or sentries and reentries in SECr. This enables you to minimize the number of debugging statements logged to solve a specific problem, thus maximizing system performance, while maintaining maximum possible debugging coverage.

Security Service modules contain built-in debugging messages, sent to the standard IS Toolkit log.

## Use

Use WALSEC\_debug from the command line while the IS Toolkit client application is running to activate or deactivate debugging for a set of entrypoint classes for the SEC and SECr shared libraries. You can also use it to view the debugging status at any time.

---

**Note** Do not use the WALSEC\_debug tool unless explicitly instructed to do so by your service representative.

---

Normally, the WALSEC\_debug tool terminates immediately after changing the debugging levels. When enabled through WALSEC\_debug, debugging remains enabled until disabled using this program or by releasing the shared memory for IS Toolkit clients. The time at which the program releases the shared memory for IS Toolkit clients is platform dependent; therefore, it is not a reliable method to disable debugging.

You should usually run the WALSEC\_debug program after the IS Toolkit client application starts but before it terminates. However, if the IS Toolkit client has a short run time or you need a complete trace of all program calls, you should start the WALSEC\_debug program using the WAIT option prior to running the IS Toolkit client.

---

**Note** The WALSEC\_debug program affects all running IS Toolkit applications at the same time.

---

## Syntax

WALSEC\_debug [+<type>] [-<type>] [STATUS] [WAIT]

---

**Tip** To display a help screen for this tool, enter the WALSEC\_debug command without any parameters.

---

## Options

**+<type>** Turns on debugging for the specified shared library or executable and entrypoint type.

<type> is one of the following:

SEC_ENTRY	A shared library (abstract) and an entrypoint classification.
SEC_SENTRY	
SEC_VENTRY	
SEC_RENTRY	
SEC_IENTRY	
SEC_UENTRY	
SECr_ENTRY	
SECr_SENTRY	
SECr_VENTRY	
SECr_RENTRY	
SECr_IENTRY	
SECr_UENTRY	

SEC_ALL	Shortcut to a group of entrypoint classifications.
SECr_ALL	

ALL	Shortcut to all shared libraries and entrypoint classifications.
-----	--

For details, see [“Types” on page 1304](#).

**-<type>** Turns off debugging for the specified shared library or executable and entrypoint type.

<type> is one of the following:

SEC_ENTRY	A shared library (abstract) and an entrypoint classification.
SEC_SENTRY	
SEC_VENTRY	
SEC_RENTRY	
SEC_IENTRY	
SEC_UENTRY	
SECr_ENTRY	
SECr_SENTRY	
SECr_VENTRY	
SECr_RENTRY	
SECr_IENTRY	
SECr_UENTRY	

SEC_ALL	Shortcut to a group of entrypoint classifications.
---------	--

SECr_ALL	
----------	--

ALL	Shortcut to all shared libraries and entrypoint classifications.
-----	--

For details, see [“Types” on page 1304](#).

**STATUS** Prints a table to standard output of each debugging type and its status. It also displays a status table if a command changed any debug settings whether you added the STATUS parameter or not.

**WAIT** Pauses debugging so it will wait for a key press before terminating. If you have a short running IS Toolkit client application or need a full trace, add the WAIT option to the command line and run WALSEC\_debug before starting your IS Toolkit application.

## Types

Types let you enable or disable debugging for all entrypoint types within a particular shared library at the same time. The final type, “ALL,” affects debugging for all shared libraries and all entrypoints at once.

You can combine types (using a plus or minus sign) on the command line by separating them with a space to produce the desired effects.

For example, to enable debugging for sentry and reentry functions within the SEC shared library and disable debugging for ventry functions in the SECr shared library with one command, type the following:

```
SEC_debug +SEC_SENTRY +SEC_RENTRY -SECr_VENTRY
```

The order of parameters on the command line is not significant.

If the command enables a debugging type that is already enabled or disables a type that is already disabled, the command has no effect on that debugging type. This is not an error.

However, if you enable and disable a single debugging type with the same command, the command generates an error. For example:

```
WALSEC_debug +SEC_SENTRY -SEC_SENTRY
```

This combination is invalid because it is ambiguous whether the program should enable or disable SEC\_SENTRY debugging.

```
WALSEC_debug +ALL -SEC_RENTRY
```

This combination is invalid because it is ambiguous whether the program should enable or disable SEC\_RENTRY debugging.



The WALSEC\_debug tool rejects and reports an invalid command due to ambiguity.

## Sample Output

The following examples show the output that displays after running the SEC\_debug command using various options.

The following sample shows output after entering the following command when all debugging is turned off.

### SEC\_debug STATUS

```
WALSEC_debug:  Manages SEC Enhanced Runtime Debugging
```

```
-----
```

```
All debugging is currently off.
```

The following sample shows output after entering the following command when SEC\_sentry debugging is turned on:

### SEC\_debug STATUS

```
WALSEC_debug:  Manages SEC Enhanced Runtime Debugging
```

```
-----  
Debugging Status:  
-----
```

Abst	Entry	Ventry	Rentry	Sentry	Ientry	Uentry	
SEC	OFF	OFF	OFF	ON	OFF	OFF	
SECr	OFF	OFF	OFF	OFF	OFF	OFF	

```
-----
```

The following sample shows output after entering the following command when no debugging is turned on.

SEC\_debug +SEC\_IENTRY +SECr\_ALL

```
WALSEC_debug:  Manages SEC Enhanced Runtime Debugging
-----
Settings changed.

Debugging Status:
-----
| Abst | Entry | Ventry | Rentry | Sentry | Ientry | Uentry |
-----
| SEC  | OFF   | OFF    | OFF    | OFF    | ON     | OFF    |
-----
| SECr | ON    | ON     | ON     | ON     | ON     | ON     |
-----
```

## Checklist

Before you use WALSEC\_debug, be aware of the following:

- You should use this tool only under the explicit direction of your service representative. However, some IS Toolkit developers might find it useful to log debug messages within the Security Services IS Toolkit entry points. IS Toolkit entry points in Security Services correspond to entry functions in the SEC shared library (SEC\_ENTRY). IS Toolkit developers should use this tool at their own risk.
- Since WALSEC\_debug runs while the IS Toolkit client is running, it must share memory resources to make debugging calls to the software. Furthermore, since each security operation can make tens or hundreds of function calls, the system log fills quickly, causing severe performance degradation. Use debugging sparingly and with extreme caution.

## Procedure

Run this program from the command line. See the [“Syntax” on page 1309](#) for details on command line options.

## Related Topics

[“SEC\\_debug” on page 1120](#)

See the “Security Administration” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

# whatsup

## Description

The whatsup program displays a multi-column report of the FileNet programs currently loaded into the memory of the server. The information is formatted differently for UNIX platforms and Windows Server platforms. For a description of the report columns, see [“Sample Output” on page 1310](#).

## Use

Use the whatsup tool to determine if the FileNet software is running and which FileNet processes are active. The preferred method for determining this information is through the FileNet Task Manager, which provides a graphical user interface for changing the software state, reporting the software state, and reporting the active processes. However, if you are not able to run the graphical user interface, such as when support personnel dial in to a site for problem determination, you can use whatsup from the command line to obtain the same information.

Use the initfnsw tool to change the Image Services software state. You specify parameters of initfnsw to change the state. After you run initfnsw, you can run whatsup to determine if the expected state has been achieved and if the programs you expect to be active are running. (For details, see [“initfnsw” on page 583](#).)

## Syntax

```
whatsup
```

## Sample Output

The sample output from the whatsup tool shows the FileNet programs currently loaded into memory. The output differs between UNIX platforms and Windows Server platforms.

### UNIX

The sample below is from a UNIX system:

---

```
FileNet Processes on server 'monterey':
User  PID      PPID      Start Time  Processes
fnsw  1812     1758     11/16/1996  bes_commit 2 Bes1:monterey:FileNet
fnsw  1836     1812     11/17/1996  bes_commit 1 Bes1:monterey:FileNet
fnsw  3488     3487     12:07:46 PM OCOR_Listen CWH 2
fnsw  1817     1816     11/16/1996  OCOR_Listen -pt -d20 -t3600
fnsw  1816     1806     11/16/1996  OCOR_Listen master
fnsw  1827     1811     11/17/1996  CSM_daemon
fnsw  1811     1758     11/16/1996  ds_init
fnsw  1824     1811     11/16/1996  ds_notify
fnsw  1825     1811     11/16/1996  ds_notify
fnsw  1826     1811     11/17/1996  dtp_tran
fnsw  1829     1811     11/17/1996  fbc_commit
fnsw  3535     3520     12:16:22 PM fnlogon
fnsw  3537     3535     12:16:29 PM fnlogon
fnsw  2193     2162     08:44:25 AM fnlogon
fnsw  2195     2193     08:44:33 AM fnlogon
fnsw  1840     1       11/17/1996  /fnsw/bin/fn_snmpd -f 8001
fnsw  1841     1       11/17/1996  /fnsw/bin/fn_trapd
fnsw  1759     1758     11/16/1996  /fnsw/bin/ilc_daemon
fnsw  1820     1809     11/16/1996  INXbg
fnsw  1818     1810     11/16/1996  INXu
fnsw  1760     1758     11/16/1996  MKF_clean
fnsw  1802     1801     11/16/1996  MKF_writer 3
fnsw  1761     1758     11/16/1996  MKF_writer 0
fnsw  1796     1795     11/16/1996  MKF_writer 2
fnsw  1790     1       11/16/1996  MKF_writer 1
fnsw  3489     3487     12:07:46 PM NCHs
fnsw  1807     1758     11/16/1996  NCH_daemon -pt
fnsw  1828     1811     11/17/1996  osi_migrate
fnsw  1805     1758     11/16/1996  perf_mon -f/fnsw/local/sd/perf_mon.script -f/fnsw/
```

---

Each column of the report from a UNIX system is defined below:

**User** is the owner of the process.

**PID** is the system assigned process ID number in decimal format.

**PPID** is the parent process ID number, in decimal format, for the program that started the process listed in the PID column.

**Start Time** is the starting date or time of the process.

**Processes** is the process or command.

### UNIX

On UNIX platforms, an OCOR\_listen process marked as CWH in the Processes field represents the connection warehouse process that owns the file descriptors of queued connections. The characters CWH are followed by the hexadecimal number of the remote program. Zero or more of these processes can exist per remote program, depending on whether any RPCs have been processed or queued.

---

### Note

Connection queuing is not supported for the Windows Server platforms.

---

All OCOR\_listen processes descend from a common ancestor. The output of whatsup identifies the common ancestor process as **master** in the Processes field.

### WIN

A Windows Server report, as shown in the sample below, differs slightly from that of a UNIX system. Report column heading TID is PPID on a UNIX platform report and values in TID and PID fields of the Windows Server report are in hexadecimal.

Each column of the report is defined below the example:

---

```

FileNet Processes on server 'iceberg.filenet.com':
User      PID      TID      Start Time   Processes
saran    0x77     0x91     11/13/1996  bes_commit 1 Bes1:iceberg:FileNet
saran    0xbc     0xd8     11/13/1996  bes_commit 2 Bes1:iceberg:FileNet
saran    0xad     0xe7     11/13/1996  OCOR_Listen -pt -d20 -t3600
saran    0xec     0xd5     11/13/1996  CSM_daemon
saran    0xde     0xa9     11/13/1996  ds_init
saran    0xe3     0xeb     11/13/1996  ds_notify
saran    0xe6     0xb1     11/13/1996  ds_notify
saran    0xd0     0xb8     11/13/1996  dtp_tran
saran    0xe1     0xc1     11/13/1996  fbc_commit
saran    0xa4     0xac     11/13/1996  fn_snmpd.exe
saran    0xd4     0xf0     11/13/1996  fn_trapd.exe
saran    0xab     0x83     11/13/1996  INXbg -s IndexServer
saran    0xb3     0xa1     11/13/1996  INXu
saran    0xc4     0xba     11/13/1996  ipc_daemon
saran    0xed     0xca     11/13/1996  MKF_clean
saran    0xd1     0xc9     11/13/1996  MKF_writer 0
saran    0xbd     0xe9     11/13/1996  MKF_writer 2
saran    0xdb     0xc6     11/13/1996  MKF_writer 1
saran    0xda     0xce     11/13/1996  MKF_writer 3
saran    0xd2     0x4c     11/13/1996  NCH_daemon -pt
saran    0xd6     0xb6     11/13/1996  osi_migrate
saran    0xcc     0xb7     11/13/1996  perf_mon -f/fnsw/local/sd/perf_
mon.script
pt -f/fnsw/lib/perf/perf_mon.script
saran    0xdc     0xb2     11/13/1996  PRI_check
saran    0xc5     0xb5     11/13/1996  PRI_daemon
saran    0xa6     0xcf     11/13/1996  PRI_notify
saran    0xe0     0xdf     11/13/1996  PRI_worker
saran    0xa2     0xcb     11/13/1996  rmt_commit
saran    0xaa     0xb4     11/13/1996  SEC_daemon
saran    0x9d     0x84     11/13/1996  TM_daemon -s
Enter <cr> to continue:

```

---



**User** is the owner of the process.

**PID** is the system assigned process ID number in hexadecimal format.

**TID** is the thread ID number, in hexadecimal format, for the program that started the process listed in the PID column.

**Start Time** is the starting date or time of the process.

**Processes** is the process or command.

## Checklist

Before you use whatsup, be aware of the following:

- The whatsup tool displays only FileNet programs loaded into the memory of the server to which you are logged on.
- The preferred method for obtaining FileNet system status is with the Task Manager Monitor.
- The output from whatsup is slightly different between UNIX platforms and the Windows Server platforms.
- In a UNIX environment, running whatsup automatically restarts a terminated TM\_daemon. However, in a Windows Server environment, whatsup does not automatically restart TM\_daemon. For more information on starting and stopping TM\_daemon in a Windows Server environment, see *System Administrator's Companion for Windows Server*. To download IBM FileNet documentation from the IBM support page, see [\*\*"Accessing IBM FileNet Documentation" on page 21.\*\*](#)

## Procedure

No specific procedure is required.

## Related Topics

**[“initfnsw” on page 583](#)**

See *System Administrator’s Handbook* for information about the Task Manager.

See *System Administrator’s Companion for Windows Server* for information about starting and stopping TM\_daemon in Windows Server.

To download IBM FileNet documentation from the IBM support page, see **[“Accessing IBM FileNet Documentation” on page 21](#)**.

# WQMpack

## Description

The WQMpack tool compresses WorkFlo Queue Services queues. WQMpack can operate on all queues of a workspace or the queues of a single table or multiple tables. Tables can be those owned by users other than f\_sw.

WQMpack uses space in the temporary directory (/tmp on UNIX platforms or \tmp on Windows Server platforms) for working storage. In addition, a list of all queues is written to a file in the temporary directory. Sufficient space to hold all the queues you are compressing must be available in the temporary directory before you start WQMpack.

If WQMpack cannot create GDB\_exim control files, error messages indicate which files the program could not create.

---

**Note** For IBM DB2, this tool is not supported.

---

## Use

Use WQMpack to move WQS queues from one server to another, such as you might do when upgrading to a new server.

---

**Note** All queues in a given workspace must reside on the same WorkFlo Queue server.

---

To determine individual table names for use in WQMpack, use WQS\_ tool.

## Syntax

```
WQMpack <-c | -s | -T> [-n] [-u | -o <table_owner> ] <tablename  
| all> \
```

```
[-I #] [-E #] [-P #] [-F #] [-U #] [-J #] [-K #] [-L #] [-M #]
```

- c** Displays space information, then compacts the queue, and displays space information again. See [“Optional RDBMS Data and Index Parameters” on page 1317](#) for options you can use with this command.
- s** Displays space information and writes the information to the .WQM-pack.space file
- T** Truncates a table

---

**CAUTION** The truncate command discards **all** data in the table!

---

- n** Writes out GDB\_exim files but does not execute GDB\_exim
- u <username>** Specifies a user (the table owner) other than f\_sw, which is the default
- o<username>** Specifies a user (equivalent to the -u command)
- <tablename>** Identifies one or more tables to export. If you specify multiple table names, separate each with a space. Mutually exclusive with **all**.
- all** Specifies all queues on the server. Mutually exclusive with <table-name>.

### Optional RDBMS Data and Index Parameters

You can use these options only with the `-c` command. The number sign (#) in the option is the number to which the data or parameter is changed:

Option	Data or Index Parameter
<code>-I #</code>	data initial extent
<code>-E #</code>	data increment extent
<code>-P #</code>	data percent increase
<code>-F #</code>	data percent free
<code>-U #</code>	data percent used
<code>-J #</code>	index initial extent
<code>-K #</code>	index increment extent
<code>-L #</code>	index percent increase
<code>-M #</code>	index percent free

## Example

To display space information for WQM001q000001, run the following command:

```
WQMpack -s wqm001q000001
```

To truncate all queue tables, use the following command:

```
WQMpack -T all
```

To compact WQM001q000001 and display space usage before and after the compaction, use the following command:

```
WQMpack -c wqm001q000001
```

Use the following command with options to compact all queue tables and change the data initial extent, data increment extent, data percent free, index percent increase, and index percent free to the values specified:

```
WQMpack -c all -I 100 -E 200 -F 50 -L 50 -M 30
```

## Checklist

Before you use WQMpack, be aware that:

- You must set the F\_MAINT\_PW environment variable before using WQMpack.
- The truncate option deletes all data from the table.
- All queues in a given workspace must reside on the same WorkFlo Queue server.
- Users should not access WorkFlo queues during a WQMpack import or export operation.
- You must ensure enough free space exists in the UNIX /tmp or Windows server \tmp directory to import or export the queues.
- It does not matter what directory you are in when you run WQMpack. However, WQMpack creates log files and other working files in the user's current working directory that do not get deleted when the tool is finished. Most of these files do not create a problem, but certain errors can create a **WQMpack.rsi** or **WQMpack.rse** file for resolving and restarting the command. If either of these files is present, there can be a problem with subsequent executions of WQMpack in this directory. Before running WQMpack, check your current working directory for WQMpack.rsi and WQMpack.rse. If these files exist, delete them before running WQMpack.

## Procedure

- 1 Use a tool such as spacerpt to determine if sufficient working space is available in the temporary directory to accommodate the queues.  
  
If insufficient space exists, you can import or export one (or a few) queues at a time or expand the temporary directory.
- 2 On the WorkFlo Queue server, enter the **WQMpack** command.

## Related Topics

[“GDB\\_exim” on page 511](#)

[“sgs” on page 1183](#)

[“spacerpt” on page 1203](#)

[“WQS\\_tool” on page 1325](#)



## WQS\_move\_q

### Description

The WQS\_move\_q tool copies WorkFlo Queue Services queues between servers. You can copy a single queue, all queues in a single workspace, or all queues in all workspaces. You can specify the program to delete queues from the source server as it copies them to the target server.

The program displays the number of the entries to copy as WQS\_move\_q begins. As processing continues, the program displays the number of queues it copies for each one per cent (1%) of the total queues copied but at least for every 100 entries copied.

---

**Note** Workspaces themselves are not moved.

---

### Use

Use WQS\_move\_q to copy WorkFlo Queue Services queues from a source server to a target server. For example, if you add a new Application server to your FileNet system, you can use WQS\_move\_q to move WQS queues from the Combined server to the new Application server.

---

**Note** You must define WorkFlo Queue Services on the target server before attempting to move queues with WQS\_move\_q.

---

---

## Syntax

WQS\_move\_q [<flags> <arguments>]

### Flags

**-d** Deletes source queues after copy

### Required Arguments

<b>Wfl_server_name</b>	Identifies the server to which the program copies the queues
<b>domain</b>	Specifies the NCH domain in which both servers reside
<b>organization</b>	Specifies the NCH organization
<b>user</b>	Specifies the FileNet user ID
<b>password</b>	Specifies the FileNet password
<b>data (y/n)</b>	Specifies whether to include data: specify y to move data or n to move description only
<b>source workspace</b>	Specifies the name of the source workspace; '*' to specify all workspaces. If source workspace is '*', source queue name is ignored if specified.
<b>source queue name</b>	Specifies the name of the source queue; '*' to specify all queues. Source queue name is ignored if source workspace is '*'.

### Optional Arguments

<b>target workspace</b>	Specifies the workspace to receive moved queues, if different than source workspace
-------------------------	---

**target queue** Specifies the new name of moved queue, if different than source queue name

## Examples

The following command moves WQS queues from the current Combined server to a new Application server, WflServer. This moves data and deletes the queues after the move completes.

```
vixen(fnsw)> WQS_move_q -d WflServer aurash FileNet SysAdmin SysAdmin y *
* wtest1 qtest1
WQS_move_q started ...
... wtest1.qtest1 -> wtest1.qtest1
                2 total entries
2 entries done
```

## Checklist

Before you use `WQS_move_q`, be aware of the following:

- You must be a member of `fnadmin` or have a valid `fnlogon` session to use `WQS_move_q`.
- WorkFlo Queue Services must be defined on the target server before attempting to move queues with `WQS_move_q`.
- Workspaces themselves are not moved.

## Procedure

No specific procedure is required.

## Related Topics

### [“WQS\\_tool” on page 1325](#)

See the *Index and WorkFlo Database Contents Manual* for information on WorkFlo Queue Database tables. To download IBM FileNet documentation from the IBM support page, see [“Accessing IBM FileNet Documentation” on page 21](#).

# WQS\_tool

## Description

WQS\_TOOL provides commands to view WorkFlo queue information. Information you can view includes table IDs, queue names, and table field names. Other WQS\_tool commands allow you to change the WorkFlo database. With these commands, you can create WorkFlo queues as well as initialize queues and table IDs.

---

**Note** Changing the WorkFlo Queue Database requires a special password. Call your service representative to obtain the password.

---

## Use

Use WQS\_tool when you are troubleshooting problems in your Image Services system. For example, an error message can return the name of a queue or the table ID for a queue. Use WQS\_tool to obtain additional information about the queue or table ID that can help you identify the problem. In addition, you can use WQS\_tool to gather statistics that help you isolate performance problems.

## Syntax

WQS\_tool

When you enter **WQS\_tool** at the command line, the WQS\_tool prompt displays:

<WQS\_tool>

Enter WQS\_tool commands at the prompt. For a list of commands, see [“Commands” on page 1326](#) and [“Administrative Commands” on page 1334](#).

## Commands

The following paragraphs describe the commands you can use with WQS\_tool. They are not case sensitive and cannot be abbreviated.

**TABLE <WORKSPACE> <QUEUE NAME>**

Displays workspaces, queue names, database table IDs, and WorkFlo server names

You can use wildcard (\*) designations for the workspace and queue name parameters to list all workspaces and queue names.

The partial listing below is a sample of output using wild cards:

```
<WQS_tool>table * *
```

Workspace	Queue name	table id	Queue Server
-----	-----	-----	-----
space1	queue1	01001	WflServer
space1	queue2	01005	WflServer
Wkspac	Wflq	01002	WflServer
Wkspac	Wflq1	01003	WflServer
Wkspac	Wflq2	01004	WflServer
cmat	cmatq1	01006	WflServer

The sample above shows information on all WorkFlo queues for all workspaces. You can limit the output by specifying a workspace or a workspace and a queue name. The example below lists only the queues for the workspace called space1:

```
<WQS_tool>table space1 *
```

Workspace	Queue name	table id	Queue Server
-----	-----	-----	-----
space1	queue1	01001	WflServer
space1	queue2	01005	WflServer

To further limit the output, specify both a workspace and queue name:

```
<WQS_tool>table spacel queue1
```

Workspace	Queue name	table id	Queue Server
-----	-----	-----	-----
spacel	queue1	01001	WflServer

### DESCQUE <WORKSPACE> <QUEUE>

Displays field information on the WorkFlo queue you specify

The output shows the names of all fields in the queue, the database column ID for the field, and whether the field is indexed or unique. Wildcards are not valid; you must specify a workspace and a queue.

```
<WQS_tool>descque spacel queue1
```

```
Queue: spacel/queue1 Server: WflServer:corona:FileNet Table id: 01001
```

Field	DB Column	Indexed?	Unique?
-----	-----	-----	-----
field1	UF000		
field2	UF001		
field3	UF002		

```
NCH queue object verified.
Verifying DB table for queue ...
done.
```

A YES in the **Indexed?** column indicates that the data for the field is sorted, or indexed, for faster retrieval.



A YES in the **Unique?** column indicates the data is sorted for fast retrieval and each row has a different, or unique, value for this field. This column also contains YES if this is a rendezvous field. (See *WorkFlo Script Developer's Guide* for a description of rendezvous fields.) A unique field is by definition also an indexed field.

#### **QIDENT <TABLE ID> <WORKSPACE>**

Displays the name of the queue that corresponds to the table ID you specify. The QIDENT command is useful when an error message returns only the table ID for the WorkFlo queue. Use this command to find the queue name that corresponds to the table ID.

You can specify a workspace in which to search for the queue or use a wildcard (\*) to search all workspaces for the corresponding queue name. The following is an example of using a wildcard to search all workspaces for the table ID 01005:

```
<WQS_tool>qident 01005 *  
  
FOUND  space1/queue2  for table WQM001Q001005
```

Table ID 01005 belongs to workspace **space1** and queue **queue2**. The format for table identification is:

<workspace name>/<queue name>

---

In IDMIS Releases 3.1.0 and later, the format of a WorkFlo table name is:

WQMsssQttttt

where

**sss** is a 3-digit decimal number corresponding to the server ID, and **ttttt** is a 6-digit decimal number corresponding to the table ID.

If the table ID is less than 6 digits, the value is padded with leading zeroes.

---

**Note** A queue table, once created, does not change by upgrading to a new release. A table created prior to IDMIS Release 3.1.0 is in the format WQM10tttt. Queues created in IDMIS Release 3.1.0 or later have the WQMsssQttttt format, for example WQM001Q001234.

---

### NEXTID

Displays the WorkFlo table ID to be used for the next table created. The following example shows the output from NEXTID:

```
<WQS_tool>nextid
Next table: 1010
```

---

The NEXTID command is helpful when moving or restoring queues on a server. The new queues could have table IDs that conflict with those already existing on the system. See INITNEXTID under [“Administrative Commands” on page 1334](#).

**DUMPQS <WORKSPACE> <QUEUE> [<RESET>]**

Displays performance statistics for the specified queue. You can use a wildcard (\*) for both the workspace and the queue. To obtain statistics on all queues, type:

```
<WQS_tool>dumpqs * *
```

The optional <reset> parameter, which must be either T(rue) or F(alse), specifies whether to initialize the counters. Use T if you want to initialize the counters; use F if you do not want to initialize the counters (F is the default).

---

**Note** Do not use the reset option unless you want to clear all timing statistics. Reset sets all counts to 0; if you issue the DUMPQS command after using reset, the **count** and **avg time** columns read 0. The reset option requires the ALLOWUPDATES command and should only be used at the direction of your service representative.

---

DUMPQS output is formatted according to WorkFlo queue functions. Any executed function appears in the first column of the table. Statistics display for the function itself, for the RDBMS database, and for interlock usage.

The following is an example of DUMPQS output for all queues of workspace **space1**:

```
<WQS_tool>dumpqs space1 *
GENERAL COUNTS - Wed Sep 9 13:55:09 1992

-----
              general                dbms                lock
count  avg time    count  avg time    count  avg time
CREATE QUEUE      4    2513.00      4    1812.00      0     0.00
GET QUEUE DESC    9     32.22      0     0.00      0     0.00
OPEN QUEUE        1    153.00      0     0.00      0     0.00
CLOSE QUEUE       1     1.00      0     0.00      0     0.00
COUNT ENTRIES   1     34.00      1    28.00      0     0.00
START_DUMP        1    183.00      0     0.00      0     0.00
END_DUMP          1     0.00      0     0.00      0     0.00
```

The output columns of the DUMPQS report are described below:

### GENERAL

Time used to execute the WorkFlo function (Create Queue, Get Queue Desc, etc.). The **count** is how many times the function is called. The **avg time** is the average time, in milliseconds, to execute the function.

### DBMS

Time used to access the RDBMS database. The **count** is the number of accesses to the RDBMS database. The **avg time** is the average time, in milliseconds, to access the database.

### LOCK

Time used when writing to an interlock. The **count** is the number of writes to an interlock. The **avg time** is the average time, in milliseconds, to write to an interlock.

**OUTPUTFILE <FILE NAME>**

Specifies the file name to which WQS\_tool output is to be directed. Use this command when you need save WQS\_tool output in a file. A typical situation might be that your service representative needs to see the information displayed by WQS\_tool to debug a problem.

The following example saves output to the file named wqsoutput in the /tmp directory:

```
<WQS_tool>outputfile /tmp/wqsoutput  
<WQS_tool>output on
```

---

**Note** OUTPUTFILE **must** be used with the OUTPUT ON command. After you issue an OUTPUTFILE command, be sure to use the OUTPUT ON command to begin writing output to the file.

---

**OUTPUT [OFF | ON]**

Toggles output writing to the file indicated in the OUTPUTFILE command. To start writing to the output file, enter **OUTPUT ON**; to stop writing to the output file, enter **OUTPUT OFF**.

**QUIT**

Exits WQS\_tool

**HELP OR ?**

Displays a brief description, including syntax, of each command

## Administrative Commands

Administrative commands are those that allow you to maintain the WorkFlo Queue database, its tables and queues.

### ALLOWUPDATES

Activates commands that make direct changes to the WorkFlo database, such as Initnextid, Qinstall, and Qfunc. Allowupdates requires a password, supplied by your service representative. Call your service representative if you need to make changes to the database.

### QINSTALL <WORKSPACE> [<OLD\_QUEUE\_NAME> [<NEW\_QUEUE\_NAME>]]

Builds a new queue in the specified workspace. You can use QINSTALL to duplicate a queue on the same server, or build queues that are moved from another server. In each case, the binary file that describes the queue must exist on the system before running QINSTALL.

The description of the queue is based on the specified old queue. You can use a wildcard (\*) for the **old\_queue\_name**; if used, all queue description files qualify. You can specify a **new\_queue\_name** only when specifying an old queue; in such a case, the old queue cannot contain a wildcard. If you do not specify a new queue name, the new queue has the same name as the old queue.

**QNCH <WORKSPACE> <QUEUE>**

Adds entries for WorkFlo queues to the Network Clearinghouse (NCH) database for all workspaces indicated. Use QNCH when you move queues to a new server and must update their domains. The System Configuration Editor initializes all entries in the NCH database except those for WorkFlo tables. The QNCH command does for WorkFlo tables what the System Configuration Editor does for all other system resources.

Indicate the workspace and queue that you want to add to the NCH database. The specifications for workspace and queue can contain wildcard characters (\*).

**Note**

---

If you are moving queues to a new server, you must import both WorkFlo database tables and WorkFlo queue definitions to the new server before you run QNCH.

---

**INITNEXTID <TABLE ID>**

Sets the next available database table ID. The valid range of numbers for table ID is 1 through 99999.

Table IDs could be out of order after moving new tables. The system usually uses INITNEXTID after moving or restoring a number of queues on a server to correct this out-of-order situation.

As an example, assume the last table was assigned an ID of 15. Using the NEXTID command, you find the next table ID will be 16. Now you add 10 queues from another server. Assume further that the highest table ID associated with those 10 queues is 48.

The system does not take into account those newly copied tables and still identifies the next table ID as 16. So you must change the next table ID to a value one greater than the largest table ID added. To do this, enter the following command at the WQS\_tool prompt:

```
<WQS_tool>INITNEXTID 49
```

### **QUNLOAD <WORKSPACE> <QUEUE>**

Closes the specified queue and flushes the queue description from memory. When a number of queues are opened but are no longer in use, use QUNLOAD to release memory allocated to these unused queues and functions.

QUNLOAD is helpful when recovering from abnormalities in which a queue is opened but not properly closed. Since the queue is considered open, the system cannot perform functions that require a closed queue (such as update\_queue). Use QUNLOAD to force the queue to close.

### **QFUNC**

Allows you to directly call WorkFlo Queue Services functions. These functions create workspaces and queues, access database information, and edit queues. After typing QFUNC, the following prompt appears:

```
Please specify which server to logon to.  
Service_name (<CR> for default)=
```

Press carriage return to accept the default WorkFlo server or enter the name of another server. The QFUNC prompt then displays:

```
q_func>
```



Enter function calls at the q\_func> prompt from this list of valid functions:

close_queue	delete_workspace	get_workspace_names	read_queue
continue	end_dump	insert_entry	start_dump
count_entries	get_default_service	logoff	update_entry
create_queue	get_queue_desc	logon	update_queue
create_workspace	get_queue_names	open_queue	update_workspace
delete_and_insert_entry	get_server_name	qlogon	
delete_entry	get_table_name	read_dump	
delete_queue	get_workspace_info	read_entry	

Type the name of the function and enter the information needed, such as workspace and queue names, at the prompts.

You can open up to five queues simultaneously under QFUNC. If a single queue is open, the selected function operates on the single queue. However, if more than one queue is open, any function to be performed on an open queue will prompt you to choose which queue is to be operated upon. The default is the last queue used.

Use the SHOW command to display the set of open queues and the current or last used queue. See [“Examples” on page 1338](#).

### SHOW

Displays the set of open queues and the name of the Queue Service. A designator of “>” indicates either the current queue or the last queue opened.

## Examples

The following QFUNC examples open multiple queues and use the SHOW command to display the current queue after a function selection. (Information you supply is in bold text. Omitted information is designated with ellipses.)

### Example 1

This example shows the open\_queue command entered twice to open two queues, followed by the SHOW command to display the open queues:

```
q_func> open_queue
:workspace name = space1
:queue name = queue_one
...
q_func> open_queue
:workspace name = space1
:queue name = queue_two
...
q_func> show
Service name: WflServer:sparta:FileNet
Open queues are:
  1. space1.queue_one
> 2. space1.queue_two
:queue number (default=2) = 1
```

The queue named queue\_two was the last opened and is therefore the current open queue. A > character in the first column of the open queue list indicates the current queue and as does the **default=** field of the SHOW output.

## Example 2

If a single queue is open and you request a function, the operation takes place on the single open queue. However, if multiple queues are open, QFUNC prompts you to select a queue for the operation. You can either press the Enter key to select the default queue or enter a number from the list for the queue you want, as shown in the following example:

```
q_func> read_queue
Open queues are:
  1. spacel.queue_one
> 2. spacel.queue_two
:queue number (default=2) = 1
...
q_func> show
Service name: WflServer:sparta:FileNet
Open queues are:
> 1. spacel.queue_one
  2. spacel.queue_two
:queue number (default=2) = 1
```

After the `read_queue` operation completes, `queue_one` is now the default queue, as the `SHOW` command indicates above.

## Checklist

Before you use WQS\_tool, be aware of the following:

- To use WQS\_tool, you must logon as the root or fnsw user, or as a member of the fnadmin group with SysAdmin privileges, or have a valid fnlogon session.
- The WorkFlo database must not be shut down.
- You can run all commands except DUMPQS from any server. Run DUMPQS at the server on which WorkFlo Queue Services resides.
- Call your service representative if you need to make changes to WorkFlo queues. You need a password to make such changes.

## Procedure

- 1 Call your service representative to obtain the password for Allowupdates, if necessary.
- 2 Enter **WQS\_tool** at a command line prompt.  
  
The WQS\_tool prompt displays.
- 3 At the <WQS\_tool> prompt, enter commands.

## Related Topics

[“WQS\\_move\\_q” on page 1321](#)

## WRT\_clean

### Description

The WRT\_clean tool deletes write requests from the transient database. Based on options you specify, WRT\_clean deletes write requests when document services is up or down.

If document services is up, WRT\_clean deletes write requests for all documents except those in the process of being committed and those being processed by the document transfer process (dtp).

If document services is down, WRT\_clean deletes all write requests with no exceptions.

If you run WRT\_clean when document services is down and commitments are in progress, you should also delete in-progress batches with the Batch Services clean tool, bes\_clean. (See [“bes\\_clean” on page 138](#).)

### Use

This tool must be run on the Storage Library server. Use WRT\_clean to delete documents that are committed but not yet written to storage media. If a problem occurs that prevents documents from being written from cache to storage media, use WRT\_clean to delete documents from cache. You can also use WRT\_clean to delete unwritten documents prior to running the del\_osvr tool. (See [“del\\_osvr” on page 288](#).)

## Use with FileNet P8 Content Federation Services (IS 4.0 SP3)

The output of the WRT\_clean tool can report some confusing information when being used with the FileNet P8 Content Federation Services functionality. Since the function of WRT\_clean is to delete committed documents from the DOCS table and the DOCTABA table, the CE catalog entry can reference an image that it believes does not exist when the documents are stored on the Image Services system but indexed on the Content Engine system.

Additionally, in the instances where documents are indexed on the Image Services system and the WRT\_clean tool is run, Index Services will create an export\_log entry to tell the FileNet P8 CFS Server for Image Services to delete the corresponding index entry on the Content Engine system. Refer to the table below to determine when documents on the IS server will not have the index information associated with them (the **Ns**) because if index information does not exist in DOCTABA for an image, there is no way to notify the Content Engine system to delete its catalog information for that image.

Document Activity	Index on CE	Index on IS
New documents entered via Content Engine	Y	N
New documents entered via Image Services		
• Indexed on both IS and C	Y	Y
• Indexed only on CE	Y	N
Existing document images transferred to CE		
• Indexed on both IS and CE	Y	Y
• Index deleted from IS	Y	N

## Syntax

```
WRT_clean { -d | -u } -f <filename>
```

**-d** Deletes all write requests when document services is down

Document services must stay down until WRT\_clean completes.

---

**Note** **fn\_util startdb** must be run before running the -d option of WRT\_clean because the databases must be up in order for the deletion to work.

---

**-u** Deletes all write requests when document services is up

**-f <filename>** Specifies a file containing document IDs to be deleted

The specified file contains one document ID per line in ASCII text. WRT\_clean deletes all write requests for each specified document. This option can **only** be used when document services is up.

## Checklist

Before you use WRT\_clean, be aware of the following:

- The file name option (-f) can only be used when document services is up.
- The databases must be up before running this tool. This is especially important when running the tool with the "-d" option because the program will hang.
- You can run WRT\_clean on a Storage Library server to delete documents with outstanding write requests on that server only. To delete documents for **all** outstanding write requests in the Image

Services system, you must run WRT\_clean on each Storage Library server.

- If the databases reside remotely, verify that the IS software and databases are all up and running on the remote servers to avoid a program hang. For example, when running this tool on a Storage Library server that is not the document locator server, verify that the software and databases are started. Also, if the RDBMS is configured remotely, verify that it is also brought up by the Database Administrator.

## Procedure

No specific procedure is required to run WRT\_clean with document services up. Enter **WRT\_clean** at the command line prompt. The databases should be up.

To run WRT\_clean with document services down, use the following procedure:

- 1 Enter **initfnsw -y stop** at the command line.
- 2 Enter **fn\_util startdb** to start just the Image Services databases.

This ensures that document services is down but the transient database is accessible to WRT\_clean.

- 3 Enter **WRT\_clean -d**.



## Related Topics

[“bes\\_check” on page 131](#)

[“checkwrt” on page 158](#)

[“del\\_osvr” on page 288](#)

[“initfnsr” on page 583](#)

# XPR\_print

## Description

All FileNet graphical user interface (GUI) applications automatically invoke XPR\_print to print application reports. However, you can print a file from the command line with the XPR\_print tool. XPR\_print prints one file at a time.

## Use

Use XPR\_print to print files. You can run XPR\_print from the command line of an X window on UNIX platforms or an MS-DOS window on the Windows Server platforms.

When you enter XPR\_print at the command line, a selection window similar to the one below displays:

**Print File or Report**

Data:

Service:  ▼

Device Type:  Printer  FAX  Additional Print Servers

Printer:  ▼

Priority:  0  1  2  3  4  5  6  7  8  9

Paper size:  ▼

Start time:

Copies:

Note:

Options:

Job Header  Collation

If you specified a file name with the XPR\_print command, the file name displays in the Data: box. If you did not specify a file name when you started XPR\_print, enter the file name you want to print in the Data: box.

You can select either a printer or facsimile (fax) device by selecting the appropriate radio button.

The default print service displays in the Service: box. Click the down arrow to display additional print services, if defined. The default printer that satisfies your print request displays in the "Printer:" box. Click on the down arrow to display additional local and remote Print servers, if they have been defined.

Click Status to display the current status of the selected device. To select a different local or remote Print server, click on the “Additional Print Servers” box.

Select other options as appropriate. Click the OK button to print your file.

## Syntax

```
XPR_print [filename]
```

## Example

The following example uses the XPR\_print command to print the .login file in the user’s /home directory:

```
XPR_print /home/.login
```

See additional information and an example under [“Use” on page 1346](#).

## Checklist

Before you use XPR\_print, be aware of the following:

- You can specify only one file name per print request.
- Wildcard characters are not allowed in the file name.

## Procedure

### UNIX

To print UNIX files with XPR\_print, the window manager must be running. At system startup, if you did not answer **yes** to the prompt to start X windows, you can start the window manager in a number of ways, for example, by typing:

**xinit**

or

**mwm &**

To print a server-based file, perform the following steps:

- 1 Enter **XPR\_print** and a file name at the command line.

If you do not specify a file name, you must enter the file name in the “Data:” box of the “Print File or Report” window.

- 2 Click the OK button to send the output to the currently selected printer.

If you didn’t specify a file name in step 1, enter the file name into the “Data:” box of the window before clicking the OK button.

To select a different printer, click the arrow button to the right of the “Printer:” box to display a printer selection window. Highlight the desired printer and click the OK button. To see all available local and remote Print servers, click on the “Additional Print Servers” box.

- 3 Print additional files if desired.

To print other files without leaving the XPR\_print, enter a new file name in the “Data:” box of the “Print File or Report” window and click the OK button.

**4** Exit XPR\_print.

When you have no further print requests, click the CANCEL button to leave XPR\_print.

## Related Topics

See the “Printing” chapter of your *System Administrator’s Handbook*. To download IBM FileNet documentation from the IBM support page, see [\*\*“Accessing IBM FileNet Documentation” on page 21.\*\*](#)

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing

---

2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106-0032, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and



---

(ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
J46A/G4  
555 Bailey Avenue  
San Jose, CA 95141-1003  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

## Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is

available on the Web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

FileNet is a registered trademark of FileNet Corporation, in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

## U.S. Patents Disclosure

This product incorporates technology covered by one or more of the following patents: U.S. Patent Numbers: 6,094,505; 5,768,416; 5,625,465; 5,369,508; 5,258,855.

## Numerics

911 69

## A

aborted transactions 733  
add\_osvr 126  
AIX/6000 systems, connecting to 565  
analyzing storage media 886

## B

background job 1232  
Background Job Control  
    consolidating media 282  
    erasing media 282  
    import log 933  
backup 41  
    enterprise 47  
    environment, establishing 579, 583  
balancing system load 126, 826  
batch  
    components, verifying 131  
    debugging 148  
    ID values, changing 151  
    removing uncommitted 138  
    services report 1002  
    table rows, locating orphaned 131  
    tables, analyzing 148  
batches 41  
BES object consistency, checking 133  
bes\_check 131  
bes\_clean 138

bes\_debug 148  
bes\_setid 151  
broadcasting identification requests 1100  
buffer pool  
    statistics 252  
    thrashing 733  
byte  
    reordering 782  
    updating order 780

## C

cache 42, 1193  
    adding partition 209  
    corruption, correcting 235  
    initializing 218  
    initializing partition 219  
    manipulating 204  
    services report 1003  
Cache Export/Import Program 191  
cache objects  
    checksum, calculating 211  
    creating 212  
    deleting 215  
    displaying 204  
    dumping 216  
    fast batch 314  
    import/export 190  
    memory requirement for exporting 191  
cache statistics  
    displaying 235  
    retrieving 204

- capability locks, listing 601
- character set
  - creating translation map 872
  - default FileNet system 880
- check\_page 154
- checkpoint verification, automatic 1197
- checksum, displaying 890
- checkwrt 158
- claimed application interlocks,
  - displaying 629
- COLD 314
- communication problems, diagnosing
  - in cross-system environments 834
  - in multiple system environments 834
- compressdir 160
- compressing
  - files 160
  - WorkFlo queues 1315
- compression, forcing 161
- configuration 43
  - system 1183, 1187
- configuration information, collecting 546
- connection, dial-up
  - kermit 298
  - testing 296
- Content Federation Services
  - dbverify 256
  - deldocs 283
  - remove\_docs\_ref 1085
  - stdoccpy 1240
  - stdocimp 644, 936, 1250
  - WRT\_clean 1342
- continuation character 24
- control files, GDB\_exim 1315

- conventions 22
- converting images 1239
- copy, global 498
- copying
  - files between systems 498
  - media 1238
  - storage media files 891
- COR handler requests
  - analyzing 163
  - dynamic information, displaying 166
- cordebug 163
- core files 43
  - analyzing 246
- common 166
- corruption
  - locating MKF database 816
- Courier
  - handle table, displaying 166
  - timing measurements 175
- CPT\_test 175
  - automatic test 176, 179
  - RPC latency test 178
  - throughput test 177
- cranuser
  - ISRA 186
  - LDAP 186
- CSAR 336, 356
- CSM\_exim 190
- CSM\_tool 204
- cstat 246
- cti 249
- customer service report 1005

**D**  
Data Definition Language

- creating 715
- file contents 712
- data dictionary 45
  - importing/exporting 266
- Database Maintenance application 126
- databases 44
  - index 49
  - MKF 53
  - NCH 56
  - Oracle 56
  - permanent 58
  - security 62
  - SQL server 53
  - transient 65
  - WorkFlo 68
- dbp 252
- dbverify 255
- dclview 262
- ddexim 266
- del\_osvr 288
- deldocs 282
- deleting
  - documents 282
  - Storage Library server 288
- dialout 295
- Disaster recovery
  - platter (p) option 334, 340
- displaying
  - checksum 890
  - current FileNet processes 633
  - document class information 891
  - memory-resident programs 1309
- DOC\_dumpfb 314
- DOC\_tool 326

- docchk 303
- doccnt 310
- docfetch 324
- document
  - class information, displaying 891
  - committal 45
  - deletion 45
  - headers, locating 913
  - IDs, preventing duplicate 938, 1196
  - importing 932, 1249
  - migration 1257
  - retrieval 46
  - services 46, 1009
  - verification 255
- dumping
  - document header files 894
  - media sectors 925

**E**

- EBR 47
- EBR\_clean 379
- EBR\_genscript 379
- EBR\_label 379
- EBR\_orreset 380
- EBR\_tdir 380
- EBR\_ulmk 380
- eladisp 381
- endbackup 401
- enlarge\_ncol 409, 576, 709
- erasing
  - media 282
  - tape data 1276
- error messages, analyzing NCH 835
- event log 47
  - PPMOI output 1047

viewing 1297  
export cache objects 190

**F**

fast batch objects, analyzing 314  
fax 1195  
file  
    compressing 160  
    displaying characteristics 1236  
    manipulation 48  
    setting characteristics 1234  
    transfer 650  
    uncompressing 1294  
FileNet P8 Content Federation Services  
    dbverify 256  
    deldocs 283  
    remove\_docs\_ref 1085  
    stdoccpy 1240  
    stdocimp 644, 936, 1250  
    WRT\_clean 1342  
FileNet software 48  
    preparing for restore 583  
    starting 583  
    stopping 583, 584  
FileNet System Manager 414, 963  
flat 417  
    basic display commands 424  
    display modes 424  
    interface 420  
    miscellaneous display commands 429  
    packet description file 445  
    quick reference 418  
    viewing and display commands 431  
fn\_edit 447  
fn\_ldif\_xfer 448

fn\_msg 464  
fn\_perm 466  
fn\_setup 478  
fn\_util 478  
fnadmin 1073  
fnddcfg 479  
FNL\_disp 481  
fnlogon 489

**G**

gaddr 493  
gcp 498  
GDB\_exim 511  
    command files 512  
    job file names 513  
    rules file 513  
GDBcheckdb 503  
GDBdebug 505  
get\_rdbms 519  
getreports 523  
getstamps 532  
getstatus 546  
getty 302  
glogin 565  
glg 568  
gsh 570

**H**

host  
    addresses, displaying on AIX/6000 493  
    configuration, determining 494  
HP optical jukebox 917

**I**

ident 572

- identifying
    - files 572
    - systems 1100
  - Image
    - client page request report 1018
  - image
    - banded 154
    - converting 154, 231, 1239
    - corruption, analyzing 231
    - inspecting 303
    - prefetch activity report 1017
    - retrieving into page cache 324
    - TIFF 154
    - tiled 154
  - import
    - cache objects 190
    - documents 932, 1249
  - import/export
    - document class information 266
    - index information 266
    - RDBMS data 511
  - index database 49
    - columns, enlarging 403
    - statistics 643
  - index services
    - capability locks 598
    - performance data, gathering 593
    - report 1013
  - initbackup 579
  - initfns 583
    - ispaused 584
    - pause 584
    - resume 584
  - initializing
    - FileNet software 583
    - NCH database 857
  - INSmsar 340, 345
  - integral SDS 50
  - interlock
    - displaying 481
    - processes, waiting 630
    - use, displaying 632
  - INX\_export 595
  - INX\_tool 598
  - INXdebug 593
  - ipc\_tool 610
  - ipcs 642
  - ipcsrm 642
  - ISRA 186
  - ixdb\_stat 643
- K**
- kermit data transfer
    - interactive 651
    - modes 650
    - server mode 651
  - killfns 656
  - killfns help 658
  - killing
    - address manager shared memory
      - segments 658
    - processes 658
    - TM\_daemon 658
- L**
- LAN analysis 417
  - language translation 55
  - ldap\_exp 662
  - ldap\_import 671



ldap\_password 690  
LDIF 448  
less 692  
licence report 1172  
Lightweight Directory Interchange  
    Format 448  
Listener 963  
load balancing 126, 826  
log file, creating performance 696  
log\_create 696  
log\_dir 699  
log\_extract 702  
logging 52  
    quick 59  
logon  
    security interface 489  
    status, checking RDBMS 503

**M**

Magnetic storage  
    checking on paused state 584  
    pausing 584  
    resuming 584  
Magnetic Storage and Retrieval 54  
managing tape functions 1280  
manifest 706  
manual organization 17  
media  
    analyzing 886  
    copying 1238  
    definition 22  
    loading 923  
    repairing 886  
    sector, finding next available 916  
memory 52

message  
    text, retrieving 464, 466  
    tuple decoding 464, 466  
messages 52  
migrating documents 1257  
MKF  
    before-image journal, zeroing 823  
    copying data 756  
    data, importing 770  
    recovery logs, zeroing 820  
MKF database 53  
    creating empty 712  
    dumping 738  
    editing tables 801  
    hangs, diagnosing 721  
    in-place recovery 764  
    modifying a block 747  
    renaming 785  
    shutting down 790  
    starting 793  
    statistics, collecting 795  
    validating contents 816  
    viewing tables 801  
    zeroing a block 747  
MKF\_ddl 712  
MKF\_debug 721  
MKF\_dump 738  
MKF\_export 756  
MKF\_fixup 764  
MKF\_import 770  
MKF\_order 780  
MKF\_rename 785  
MKF\_shutdown 790  
MKF\_startup 793

- MKF\_stats 795
- MKF\_tool 801
- MKF\_verify 816
- MKF\_zeroaij 820
- MKF\_zerobij 823
- modifying
  - Oracle database tables 1226
  - WorkFlo queue 1325, 1334
- monitoring tape functions 1280
- move\_disk 826
- moving
  - RDBMS data between systems 511
  - WorkFlo queues 1321
- MSAR 54
- msar\_io\_test 829
- msar\_sync\_test 831
- N**
- NCH 55
  - error messages, analyzing 835
  - network number, updating 870
  - properties, listing 859
  - report 1015
  - resource name 842
  - server, locating 856
  - variable value, displaying or setting 868
  - wait conditions, analyzing 837
- NCH cache
  - flushing 839
  - resizing 839, 867
  - statistics, displaying 852
- NCH database 56
  - collecting information 841
  - initializing 857
- NCH domain
  - adding 851
  - default 854
  - displaying 864
  - listing 857
  - name, changing 870
  - removing 867
  - removing remote domains 867
- NCH object
  - changing 853
  - creating 853
  - deleting 854
  - listing 858
- NCH property
  - deleting 854
  - names 861
  - value, displaying 868
- nch\_check 832
- nch\_flush 839
- nch\_tool 841
- network
  - broadcast 1100
  - data transfer efficiency 179
  - problems, diagnosing 832
  - trace analysis 417
- NLT\_build\_maps 872
- ntdm\_exp 881
- numeric index values, increasing 403
- O**
- oddump 886
  - firmware 914
  - hponline 917
- odrecover 932
- online drive repair 917
- optical storage

- library 1190
- Oracle
  - archive log mode status,
    - determining 940
  - database 56
  - modifying database tables 1226
  - viewing database tables 1215
- oraloglist 940
- OSAR 896
  - error log summaries 942
- osschk 942
- overview 28

## P

- partitions 1194
  - adding cache 209
  - initializing cache 219
- password 690
- perf\_mon 954
- perf\_report 968
- performance 57
  - analysis, index services 593
  - displaying records 699–701
  - extracting records 702–705
  - log file, creating 696–697
  - statistics, collecting 546
- permanent database 58
- phases
  - stdocopy 1238
- Philips drives
  - downloading firmware 914
- ppm\_log 1028
- PPMOI 1030
  - FAQs 1051
  - statistics 1050

- PRI\_tool 1056
- print activity report 1019
- printers 1195
- printing 59
- process control
  - analyzing 481
  - records, displaying 481
- processes
  - displaying FileNet 633
  - killing 658
- processor summary report 1012
- programs, displaying memory-
  - resident 1309

## Q

- QLG\_dump 1075
- QLG\_flush 1078
- QLG\_start 1080
- qpref 1044

## R

- RDBMS
  - checking logon status 503
  - configuration information 519
  - deadlock retry testing 506
  - performance information 519
  - tracing 505
- reassign media 826
- release partition, maintaining 706
- remote
  - commands, running 570
  - communications 60
  - system directories, listing 568
- remote server 1186
- remove\_docs\_ref 1084

remove\_surf 1095  
renaming an MKF database 785  
repairing storage media 886  
reports  
    batch services 1002  
    cache services 1003  
    client page request 1018  
    customer service 1005  
    document services 1009  
    image prefetch activity 1017  
    index services 1013  
    license 1171  
    NCH 1015  
    predefined 973  
    print activity 1019  
    processor summary 1012  
    samples 1000  
    security services 1020  
    service activity 1022  
    SQL services 1021  
    storage library load 1016  
    WorkFlo activity 1024  
request handlers 60  
restore, enterprise 47  
retrieving  
    images into page cache 324  
    message text 464, 466  
RSEC\_imp 1103

**S**

SC\_convert 1116  
scalar numbers table, updating 1196  
SDS 334, 340  
SEC\_debug 1120  
SEC\_imp 1129

SEC\_init 1142  
SEC\_map 1146  
SEC\_rm\_config 1155  
SEC\_tool 1163  
security 62  
    analyzing services 1163  
    database 62  
    defaults, resetting 1142  
    logon 489  
    objects, initializing database 1142  
    rebuilding database 1143  
    services report 1020  
semaphore status 627  
server  
    displaying details 1186  
    ID assignment 128  
    performance information, collecting 523  
    stubs 60, 1048  
services  
    activity report 1022  
    request handlers 1048  
setting file characteristics 1234  
sgs 1183  
shared library mapping 624  
shared memory  
    blocks, displaying 615  
    dump 625  
    segments, dumping 636  
    use summary 622  
shutting down  
    FileNet system 656  
    MKF database 790  
SLAC key license 1007, 1172  
SLU usage 1007, 1172

- software, FileNet 48
- space
  - analyzing RDBMS 1203
  - reclaiming 160
  - usage 63
- spacerpt 1203
- spinup 929
- SQL
  - server database 53
  - services report 1021
- sqlplus 1215
- ssn 1231
- st\_msar\_convert 1232
- stamp 1234
- stampro 1236
- starting
  - FileNet software 583
  - MKF database 793
- statistics
  - buffer pool 252
  - collecting MKF database 795
- status, displaying Image Services software 583
- stdoccpy 1238
  - phases 1238
- stdocimp 1249
- stmigrate 1257
- stopping FileNet software 583, 584
- storage
  - media 61
- storage library
  - adding to server 126
  - disabling 334, 340
  - load report 1016

- locating server ID 128
- removing from server 288
- server 63
- statistics, collecting 326
- Storage Library Control 326
- stsurfupdate 1262
- subsystems 1192
- sync\_write\_test 1267
- sys\_log 1274
- system
  - configuration settings 1183
  - identification 1100
  - information 64
  - log message, adding 1274
  - performance statistics, collecting 546
  - resource information, collecting 841
  - serial number, displaying 1231
- System Manager 414, 963
- System Manager listener agent
  - always run 415
  - temporarily run 415

**T**

- TAP\_tool 1280
- tape
  - data, erasing 1276
  - drives 64, 1190
  - functions, managing and monitoring 1280
  - serial number, writing 1276
- tapeser 1276
- Task Manager, FileNet 579, 585
- test\_raw\_partition 1284
- TIFF images 154
- TLIB\_tool 1293

TM\_daemon  
   killing 658  
   restarting 660, 1313  
   starting in Windows Server 585  
   stopping in Windows 657  
   stopping in Windows Server 585  
   terminating 584  
 tool selection 41  
 transient database 65  
 troubleshooting 69  
   communication problems 834

**U**

uncompressdir 1294  
 uncompressing files 1294  
 unified logon 662, 671, 881, 1103, 1129  
 unique index values, counting 249  
 unused sectors, locating 913, 914

**V**

validating MKF database contents 816  
 verifying  
   batch components 131  
   documents 255  
   NCH\_daemon 832  
   server's default domain 832  
 viewing  
   event log 1297  
   file characteristics 1236  
   files 692  
   Oracle database tables 1215  
   system serial number 1231  
   WorkFlo queue information 1325  
 views, creating document class 262  
 vl 1297

**W**

wait times, diagnosing Document  
   Services 252  
 WALSEC\_debug 1300  
 whatsapp 1309  
 WorkFlo activity report 1024  
 WorkFlo queue  
   compressing 1315  
   database, modifying 1334  
   modifying 1325  
   moving 1321  
   name, locating 1329  
   viewing 1325  
 WQMPack 1315  
 WQMPack.rse  
   deleting 1319  
 WQMPack.rsi  
   deleting 1319  
 WQS\_move\_q 1321  
 WQS\_tool 1325  
 write requests  
   analyzing 158, 1341  
   deleting 1341  
   pending 288  
 WRT\_clean 1341

**X**

XPR\_print 1346

**Z**

zeroing MKF  
   before-image journal 823  
   recovery logs 820





Program Number: 5724-R95

Printed in USA

GC31-5543-01

