Log Analyzer for IMS User Guide

Supporting

Version 1.5 of BMC Log Analyzer for IMS
Version 2.7 of BMC System Administration for IMS
Version 1.4 of BMC System Communication for IMS

December 2015
Contacting BMC Software

Several methods are available for contacting BMC Software.

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

Address: BMC SOFTWARE INC
2103 CITYWEST BLVD
HOUSTON TX 77042-2827 USA

Telephone: 1 713 918 8800
or
1 800 841 2031

Fax: 1 713 918 8000

Outside United States and Canada

Telephone: +01 713 918 8800
Fax: +01 713 918 8000

© Copyright 2007-2016 BMC Software, Inc.

BMC, BMC Software, the BMC logo, the BMC Software logo, and other BMC marks are the exclusive properties of BMC Software, Inc. and are registered or may be registered in the U.S. and in other countries. BladeLogic and other BladeLogic marks are the exclusive properties of BladeLogic, Inc. and are registered or may be registered in the U.S. and in other countries. All other trademarks or registered trademarks are the property of their respective owners.

Together is a trademark or registered trademark of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

CICS, DB2, DRDA, IBM, IBM z, IMS, MQSeries, RACF, VTAM, VTF, WebSphere, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2103 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.
Customer support

Support website
You can obtain technical support from BMC 24 hours a day, 7 days a week at http://www.bmc.com/support. From this website, you can:

- Read overviews about support services and programs that BMC offers
- Find the most current information about BMC products
- Search a database for problems similar to yours and possible solutions
- Order or download product documentation
- Download products and maintenance
- Report a problem or ask a question
- Subscribe to receive proactive e-mail alerts
- Find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail
In the United States and Canada, if you need technical support and do not have access to the web, call 1 800 537 1813 or send an e-mail message to customer_support@bmc.com. (In the subject line, enter SupID:yourSupportContractID, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC
Have the following information available so that Customer Support can begin working on your issue immediately:

- Product information
  - Product name
  - Product version (release number)
  - License number and password (trial or permanent)
- Operating system and environment information
  - Machine type
  - Operating system type, version, and service pack or other maintenance level such as PUT or PTF
  - System hardware configuration
  - Serial numbers
  - Related software (database, application, and communication) including type, version, and service pack or maintenance level
- Sequence of events leading to the problem
- Commands and options that you used
- Messages received (and the time and date that you received them)
  - Product error messages
  - Messages from the operating system
  - Messages from related software
License key and password information

If you have questions about your license key or password, contact Customer Support through one of the following methods:

- Send an e-mail message to customer_support@bmc.com. (In the Subject line, enter SupID:yourSupportContractID, such as SupID:12345.)
- In the United States and Canada, call 1 800 537 1813. Outside the United States and Canada, contact your local support center for assistance.
# Contents

## About This Book

- Related publications .............................................................. 15
- Conventions ........................................................................ 16
- Syntax statements .................................................................. 16
- Summary of changes .............................................................. 17

## Chapter 1         Overview of Log Analyzer

- Automation of analysis tasks .............................................. 19
- Purposeful research ........................................................... 19
- Features and benefits of Log Analyzer ................................. 20
  - Logical units of work ...................................................... 20
  - Log record selection ...................................................... 23
  - Flexible search criteria .................................................. 23
  - Log record sources ........................................................ 23
  - Interactive LUOW analysis ............................................ 24
  - Predefined reports .......................................................... 24
  - Information about read-only databases in LUOWs ............. 26
  - Application checkpoint information ................................. 26
  - Deadlock information ..................................................... 26
  - Outstanding customer support ........................................ 26
- A batch way to start using Log Analyzer ................................. 27
  - A simple example ........................................................... 27
  - A robust example ............................................................ 28
- An interactive way to use Log Analyzer .................................. 30
- Considerations .................................................................. 33
  - CPU authorization passwords for Log Analyzer ................. 33
  - APF authorization ............................................................ 34
  - Security access controls .................................................. 34
  - Resource consumption and production workloads .............. 34
  - Nonunique IMSIDs .......................................................... 35

## Chapter 2         Getting started with Log Analyzer

- Task summary for using Log Analyzer .................................. 37
- Log Analyzer architecture and processes ............................... 38
  - Log Analyzer JCL and control statements ......................... 39
  - Syntax processing ............................................................ 39
Log record source processing .................................................................40
Log record processing .........................................................................41
LUOW processing ................................................................................41
Output processing ................................................................................41
Log Analyzer reports ..........................................................................42
Using Log Analyzer ............................................................................44
Working with the Log Analyzer ISPF interface .....................................45
Overview of the ISPF interface ............................................................45
Accessing the ISPF interface ...............................................................46
Navigating between panels .................................................................47
Scrolling through information on a panel ..........................................48
Using field prompts ...........................................................................49
Using action bar commands ...............................................................49
Using global commands ...................................................................50
Setting preferences ...........................................................................51
Function-key definitions .....................................................................53
Using online Help .............................................................................53
Presetting profile variables ..............................................................54

Chapter 3 Working with Log Analyzer jobs ............................... 55
Specifying JOB EXEC and DD statements ........................................56
JOB statement ..................................................................................56
EXEC statement ................................................................................57
STEPLIB DD statement ......................................................................57
SYMDUMP DD statement ..................................................................58
SYSDUMP DD statement ....................................................................58
LUITRACE DD statement ...................................................................58
SYSIN DD statement ..........................................................................58
Specifying control statements .........................................................59
Samples of control statements .........................................................59
Control statement syntax ...............................................................59
Control statement commands .........................................................61
Minimum sets of control statements ................................................62
Specifying ANALYZE control statements .......................................63
ANALYZE control statements .........................................................63
ACTIVEOLDS keyword ..................................................................67
DB2BSDS keyword ..........................................................................68
DB2LOG keyword ...........................................................................68
ELAPSEDTIME keyword ...............................................................69
IMSID keyword ...............................................................................70
INPEXT keyword ............................................................................70
Specifying INTERVAL control statements ............................................................... 79
    INTERVAL control statement ............................................................................ 80
    START keyword ................................................................................................ 80
    STOP keyword .................................................................................................. 82
Specifying FILTER control statements ................................................................. 83
    FILTER control statement ............................................................................... 83
    CONTENT0103 keyword .................................................................................... 84
    LOGRECORDCODES keyword on the FILTER control statement .................. 85
    SELECT keyword .............................................................................................. 89
    ABENDS subkeyword ......................................................................................... 94
    APPC_LTERM subkeyword ................................................................................ 95
    CICSTRAN subkeyword ...................................................................................... 95
    CLIENTID subkeyword ....................................................................................... 95
    CONTENT subkeyword ....................................................................................... 96
    DBD subkeyword ............................................................................................... 97
    DEST subkeyword ............................................................................................. 98
    ETIME subkeyword ............................................................................................ 98
    LTERM subkeyword .......................................................................................... 99
    LUNAME subkeyword ....................................................................................... 100
    MPR subkeyword ............................................................................................ 100
    NODE subkeyword ........................................................................................... 101
    OIMS subkeyword ............................................................................................ 101
    ORIGIN subkeyword ......................................................................................... 102
    PORT# subkeyword .......................................................................................... 102
    PSB subkeyword on the FILTER control statement ............................................. 103
    PST subkeyword ............................................................................................... 103
    RDBD subkeyword ............................................................................................ 103
    TMEMBER subkeyword ..................................................................................... 104
    TPIPE subkeyword ........................................................................................... 104
    TPNAME subkeyword ....................................................................................... 105
    TRAN subkeyword ............................................................................................ 105
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDBD subkeyword</td>
<td>106</td>
</tr>
<tr>
<td>UOW1 subkeyword</td>
<td>106</td>
</tr>
<tr>
<td>USERID subkeyword</td>
<td>107</td>
</tr>
<tr>
<td>Specifying REPORTS control statements</td>
<td>107</td>
</tr>
<tr>
<td>REPORTS control statement</td>
<td>108</td>
</tr>
<tr>
<td>ABEND keyword</td>
<td>110</td>
</tr>
<tr>
<td>APPCHECK keyword</td>
<td>111</td>
</tr>
<tr>
<td>AUDIT keyword</td>
<td>115</td>
</tr>
<tr>
<td>DEADLOCK keyword</td>
<td>118</td>
</tr>
<tr>
<td>DUMPFORM keyword</td>
<td>119</td>
</tr>
<tr>
<td>DUMPREC keyword</td>
<td>119</td>
</tr>
<tr>
<td>DUMPVERT keyword</td>
<td>120</td>
</tr>
<tr>
<td>LOGRECORDCODES keyword on the REPORTS control statement</td>
<td>122</td>
</tr>
<tr>
<td>LUOWDETAILED DATA keyword</td>
<td>124</td>
</tr>
<tr>
<td>LUOWSUMM keyword</td>
<td>125</td>
</tr>
<tr>
<td>LUOWTSEQ keyword</td>
<td>125</td>
</tr>
<tr>
<td>OFFSET keyword</td>
<td>126</td>
</tr>
<tr>
<td>ORPHANS keyword</td>
<td>126</td>
</tr>
<tr>
<td>RBASTATS keyword</td>
<td>127</td>
</tr>
<tr>
<td>SUMMARY keyword</td>
<td>127</td>
</tr>
<tr>
<td>Specifying EXTRACT, INDEXFILE, and WORKFILE control statements</td>
<td>128</td>
</tr>
<tr>
<td>EXTRACT, INDEXFILE, and WORKFILE control statements</td>
<td>129</td>
</tr>
<tr>
<td>DATACLASS keyword</td>
<td>132</td>
</tr>
<tr>
<td>DCB keyword</td>
<td>132</td>
</tr>
<tr>
<td>DSN keyword</td>
<td>132</td>
</tr>
<tr>
<td>LIKE keyword</td>
<td>133</td>
</tr>
<tr>
<td>MGMTCLASS keyword</td>
<td>133</td>
</tr>
<tr>
<td>PRISPACE keyword</td>
<td>133</td>
</tr>
<tr>
<td>SECSPACE keyword</td>
<td>134</td>
</tr>
<tr>
<td>SPACEUNITS keyword</td>
<td>134</td>
</tr>
<tr>
<td>STORCLASS keyword</td>
<td>134</td>
</tr>
<tr>
<td>UNIT keyword</td>
<td>134</td>
</tr>
<tr>
<td>VOLSER keyword</td>
<td>135</td>
</tr>
<tr>
<td>Specifying END control statements</td>
<td>135</td>
</tr>
<tr>
<td>Checking results of Log Analyzer jobs</td>
<td>135</td>
</tr>
</tbody>
</table>

**Chapter 4** Building Log Analyzer JCL through the ISPF interface 137

Overview of building JCL in the ISPF interface 137
Accessing JCL functions 138
Providing SYSIN, JOB, and STEPLIB information 139
<table>
<thead>
<tr>
<th>Chapter 6</th>
<th>Working with reports in the ISPF interface</th>
<th>211</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accessing reports in the ISPF interface</td>
<td>212</td>
</tr>
<tr>
<td></td>
<td>Filtering LUOWs</td>
<td>215</td>
</tr>
<tr>
<td></td>
<td>Listing LUOWs</td>
<td>216</td>
</tr>
<tr>
<td></td>
<td>Listing destinations</td>
<td>220</td>
</tr>
<tr>
<td></td>
<td>Listing origins</td>
<td>221</td>
</tr>
<tr>
<td></td>
<td>Listing user IDs</td>
<td>222</td>
</tr>
<tr>
<td></td>
<td>Listing PSBs</td>
<td>223</td>
</tr>
<tr>
<td></td>
<td>Listing CICS transactions</td>
<td>224</td>
</tr>
<tr>
<td></td>
<td>Listing other LUOWs</td>
<td>225</td>
</tr>
<tr>
<td></td>
<td>Analyzing deadlock information</td>
<td>226</td>
</tr>
<tr>
<td></td>
<td>Obtaining summary information</td>
<td>229</td>
</tr>
<tr>
<td></td>
<td>Sorting listed items</td>
<td>232</td>
</tr>
<tr>
<td></td>
<td>Obtaining LUOW details</td>
<td>235</td>
</tr>
<tr>
<td></td>
<td>Listing log records for an LUOW</td>
<td>237</td>
</tr>
<tr>
<td></td>
<td>Obtaining log record details</td>
<td>240</td>
</tr>
<tr>
<td></td>
<td>Listing adjacent log records</td>
<td>242</td>
</tr>
<tr>
<td></td>
<td>Dumping log records</td>
<td>245</td>
</tr>
<tr>
<td></td>
<td>Browsing log records</td>
<td>246</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 7</th>
<th>Allocating log record sources</th>
<th>249</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicitly specifying IMS log data sets to allocate</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>Overview of explicit specification</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>Specifying SLDS and OLDS data set names</td>
<td>251</td>
<td></td>
</tr>
</tbody>
</table>
Explicitly specifying log data sets that are not cataloged ........................................... 251
Specifying the data set name of an active OLDS .......................................................... 251
Implicitly specifying IMS log data sets to allocate .................................................... 252
  Overview of implicit specification ........................................................................ 252
  Specifying RECON data set names .................................................................... 253
  Allocating RECON data sets automatically ....................................................... 254
  Specifying IMSIDs ......................................................................................... 255
  Specifying a maximum number of logs to read ................................................ 255
  Using an active OLDS through implicit specification ..................................... 256
  Implicitly specifying log data sets that are not cataloged ................................ 257
  Using implicit and explicit specifications in the same job .............................. 257
Using extract files as input ....................................................................................... 257
  Overview of input extract files ......................................................................... 258
  Allocating input extract files .......................................................................... 258
  Reducing the number of selected log records .................................................. 259
  Gathering data from remote systems .............................................................. 259
Using Energizer journals as input ............................................................................. 260
  Overview of Log Analyzer processing of Energizer journals ...................... 260
  Allocating Energizer journals explicitly ....................................................... 261
  Allocating Energizer journals implicitly ....................................................... 261
Using DB2 logs as input ............................................................................................ 262
  Overview of DB2 log processing .................................................................. 262
  Allocating DB2 logs explicitly .................................................................... 263
  Allocating DB2 logs implicitly ................................................................. 263
Using WebSphere for MQ extract data sets ............................................................. 265
  Creating WebSphere for MQ extract files ................................................... 266
  Specifying WebSphere for MQ extract files ............................................... 266
  WebSphere for MQ log records in LUOWs .................................................. 266
  WebSphere for MQ log records in reports and panels .................................... 267

Chapter 8 Identifying candidate records and creating LUOWs 269

Skipping and limiting log records ........................................................................... 270
  Overview of skip and limit tests .................................................................... 270
  Specifying SKIP and LIMIT keywords .......................................................... 271
Identifying log record codes ................................................................................... 271
  Overview of the log record code test ............................................................. 271
  Omitting the LOGRECORDCODES keyword ............................................. 272
  Specifying the LOGRECORDCODES keyword ............................................. 273
  Automatically included log record types ..................................................... 274
  Summary of keywords that affect the log record code test ......................... 274
  Using a log record code PDS ...................................................................... 275
Specifying intervals ................................................................. 277
  Overview of the interval test .................................................. 277
  Specifying start and stop times ............................................. 278
  Specifying first and last log records ..................................... 279
  Specifying log sequence numbers ....................................... 279
  Specifying multiple intervals in an analysis job .................... 280
Creating LUOWs ........................................................................ 281
  Overview of LUOWs and LUOW creation ............................. 281
  Unresolved associations when creating LUOWs .................... 282

Chapter 9  Selecting log records and LUOWs 283
Specifying SELECT filters .......................................................... 284
  Overview of SELECT filters .................................................. 284
  Specifying a filter type ........................................................ 284
  Specifying values in SELECT filters ..................................... 286
  Using logical operators in SELECT filters ............................. 286
  Searching for content in 01 and 03 records ......................... 287
  Specifying multiple SELECT filters with Boolean operators ... 288
Selection process for log records and LUOWs ......................... 289
  Record evaluation .............................................................. 289
  LUOW evaluation .............................................................. 289

Chapter 10  Working with analysis data 291
Overview of working with analysis data .................................. 291
Producing batch reports .......................................................... 291
  Overview of batch reports .................................................. 292
  Choosing batch reports to produce ..................................... 293
  Specifying log record codes for batch reports ..................... 294
  Including data from type 01 and type 03 records .................. 295
  Specifying the format of dumped log records ....................... 295
Working with extract files ....................................................... 296
  Overview of extract files .................................................. 296
  Allocating an extract file .................................................... 297
  Using an extract file through the ISPF interface .................... 298
  Using an extract file as input to another analysis job .......... 299
  Using the Reformat Extract File utility .............................. 299
Working with index files .......................................................... 300
  Overview of index files ..................................................... 300
  Allocating an index file ..................................................... 300
  Using an index file ........................................................... 302
Interpreting information in reports .......................................... 302
### Contents

Types of LUOWs ................................................................. 302
LUOW numbers and UOW-1 numbers ............................... 304
Variable data snipped from log records .............................. 304
Unresolved associations field .............................................. 305
Refining the results of automated analysis ............................ 305

**Chapter 11** Using Log Analyzer to diagnose problems and answer questions 309
Auditing user actions ........................................................ 310
Viewing database update activities ...................................... 312
Analyzing message-switching transactions and MSC traffic 313
  A message-switching and MSC-traffic scenario ...................... 313
  Analyzing transactions if data is available from a remote site 314
  Analyzing transactions if data is not available from a remote site 317
Filtering LUOWs by elapsed time ....................................... 319
Identifying abnormal terminations (abends) .......................... 319
  Abends in LUOWs .......................................................... 320
  Abends as individual incidents ........................................... 321
Working with DL/I call trace information ............................. 321
Obtaining information about application checkpoints .............. 323
  Overview of application checkpoints ................................... 323
  Overview of the Application Checkpoint report .................. 324
  Requesting the Application Checkpoint report .................... 326
  Controlling checkpoint frequency with AR/CTL .................. 327

**Appendix A** Troubleshooting and FAQs 329
General troubleshooting procedures and tools ........................ 329
  Contacting BMC Customer Support ................................... 329
  Using the Trace facility .................................................... 329
  Obtaining information about Log Analyzer messages ............ 330
Troubleshooting analysis jobs ............................................. 331
  An analysis job runs too long ......................................... 331
  An analysis job consumes too many resources ..................... 332
  Storage compression has failed ....................................... 332
Troubleshooting LUOW and log record selection .................. 333
  An expected log record was not selected ......................... 334
  One or more filters seem to be ignored ............................. 334
  No log records were selected ......................................... 334
  Log records seem to be missing from an LUOW .................. 335
Troubleshooting reports and interactive analysis ................... 335
  An expected log record is not displayed ........................... 335
  Analysis results cannot be displayed in the ISPF interface ...... 336
An error occurs during allocation of output data sets ................................. 336

### Appendix B Sample and syntax reference 337
- Samples .................................................................................................................. 337
  - Samples in the control library ........................................................................... 337
  - Samples in the sample library .......................................................................... 338
- Control statement summary .................................................................................. 341

### Appendix C Log record and LUOW reference 345
- IMS log record types .............................................................................................. 345
- Non-IMS log record types ..................................................................................... 361
- Log records types in transaction-type LUOWs .................................................... 366
- Nontransaction and other LUOW types ................................................................. 367
- Log record types that are included automatically ................................................ 369
- Variable data types ............................................................................................... 374
About This Book

This book contains detailed information about the Log Analyzer component for the IBM IMS environment. Log Analyzer is available with the following solutions and products:

- BMC Log Analyzer for IMS
- BMC System Administration for IMS
- BMC System Communication solution for IMS

This book is intended for IMS system programmers, IMS database administrators, IBM z/OS system programmers, auditors, and others who need to obtain information from IMS log records. To use this book, you should have a working knowledge of the information that can be obtained from IMS log records.

Like most BMC documentation, this book is available in printed and online formats. To request printed books or to view online books and notices (such as release notes and technical bulletins), see the support website at http://www.bmc.com/support.

Note
Online books are formatted as PDF or HTML files. To view, print, or copy PDF books, use the free Adobe Reader from Adobe Systems. If your product installation does not install the reader, you can obtain the reader at http://www.adobe.com.

The software also offers online Help. To access Help, press F1 within any product or click the Help button in graphical user interfaces (GUIs).

Related publications

From the BMC Support Central website, you can use the following methods to access related publications that support your product or solution:

View Quick Course videos (short overviews of selected product concepts, tasks, or features), which are available from the following locations:

— Documentation Center (primary center and secured center)

— Support Central (at http://www.bmc.com/support/mainframe-demonstrations)

— BMC Mainframe YouTube channel (https://www.youtube.com/user/BMCSoftwareMainframe)


Products with online interfaces also offer online Help via the F1 key or, for graphical user interfaces (GUIs), via a Help button.

Tip

If you prefer hardcopy documentation, you can order it from your BMC sales representative or from Support Central. Also, from Support Central you can subscribe to receive proactive e-mail alerts when BMC issues notices.

Conventions

This document uses the following special conventions:

— All syntax, operating system terms, and literal examples are presented in this typeface.

— Variable text in path names, system messages, or syntax is displayed in italic text: testsys/instance/fileName

— Menu sequences use a symbol to convey the sequence. For example, Actions => Create Test instructs you to choose the Create Test command from the Actions menu.

Syntax statements

This topic explains conventions for showing syntax statements.
A sample statement follows:

```
COMMAND KEYWORD1 [KEYWORD2 | KEYWORD3] KEYWORD4={YES | NO} fileName...
```

The following table explains conventions for syntax statements and provides examples:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Example</th>
</tr>
</thead>
</table>
| Items in italic type represent variables that you must replace with a name or value. If a variable is represented by two or more words, initial capitals distinguish the second and subsequent words. | alias
databaseDirectory
serverHostName |
| Brackets indicate optional items. Do not type the brackets when you enter the option. A comma means that you can choose one or more of the listed options. You must use a comma to separate the options if you choose more than one option. | [tableName, columnName, field]
[-full, -incremental, -level] |
| Braces indicate that at least one of the enclosed items is required. Do not type the braces when you enter the item. | {DBDName | tableName}
UNLOAD device={disk | tape, fileName | deviceName}
{-a | -c} |
| A vertical bar means that you can choose only one of the listed items. In the example, you would choose either commit or cancel. | {commit | cancel} |
| An ellipsis indicates that you can repeat the previous item or items as many times as necessary. | columnName... |

**Summary of changes**

The following enhancements are added to version 1.5.00 of the BMC Log Analyzer for IMS product:

- Support for IBM IMS Version 14.1

  Log Analyzer now supports Version 14.1 of the IBM IMS system.
• Audit report enhancements
  The Log Analyzer Audit report now includes x'5F' Log Records. Additionally, this release adds Database Update records (including segments) and Log Record sequence number fields to the user interface.
  For more information, see “AUDIT keyword” on page 115.

• LUOW enhancements
  Log Analyzer now supports ODBM, providing reporting and correlation between IMS Log Records and Energizer for IMS Connect Journal records for ODBM data. Also, additional Energizer OTMA log record events are now included within the LUOW.
Overview of Log Analyzer

Log Analyzer collects, organizes, and presents IMS log information.

IMS logs contain the answer to almost any question that you could ask about your IMS environment and the activities and events in that environment. But the primary purpose of IMS logs is to record the information, not to organize and present it in a way that you can easily use. Records have cryptic contents, with codes and fields that can be recognized only with expert-level knowledge. In addition, the quantity of records in logs can present a challenge; your answer might be found in a handful of records among millions.

Automation of analysis tasks

Log Analyzer automates tedious and difficult tasks that are required for analysis of data in IMS log records. Log Analyzer performs the following tasks:

- Reduces "noise" by eliminating log records that are not relevant for your needs
- Organizes log records by taking the following actions:
  - Finding records of interest
  - Grouping related records into logical units of work (LUOWs)
  - Sorting records for logical presentation
  - Identifying records and fields with understandable labels
- Presents selected, filtered, and organized log information in easily accessed formats to help you with interactive analysis (through the Log Analyzer ISPF interface) or manual analysis (through batch reports)

Purposeful research

You can use Log Analyzer to conduct research on IMS log records for many purposes, including these:
Diagnosing problems, such as transaction failures and delays, in an IMS environment

Log Analyzer makes it easy to trace all events that are related to an activity. For example, you can see all IMS records for a transaction, from beginning to end, even for message-switching transactions and transactions that execute across Multiple Systems Coupling (MSC) links.

Searching for specific log records

Log Analyzer provides powerful and flexible filters for finding log records. For example, you can perform the following tasks:

— Search for various types of targets (such as user IDs, transaction origins and destinations, and database names)

— Include and exclude specific types of log records (such as including sign-on and sign-off records and excluding padding records)

— Use a combination of filters with Boolean logic

— Search for content in IMS transaction input and output messages

Auditing user, terminal, database, and other activities

Log Analyzer produces clear, accurate reports to help auditors, security analysts, and administrators track access and usage and ensure compliance with internal and external regulations.

Features and benefits of Log Analyzer

Log Analyzer features offer unparalleled power, flexibility, and convenience.

Logical units of work

Log Analyzer organizes log information into logical units of work (LUOWs), which provide a more comprehensive, useful view of activities in your IMS systems than a simple unit of work (UOW) can provide.

A UOW is a familiar concept in the IMS environment:

— In an IMS database management (DB) environment, a UOW comprises all input and output messages that are associated with a transaction.

— In an IMS transaction management (TM) environment, a UOW comprises a single IMS message.
In an IMS Shared Queues (CQS) environment, a UOW comprises a client-defined grouping of data objects.

The following figure shows typical activities in a transaction UOW. A message arrives in IMS and is placed on a message queue. The message is scheduled for execution. Application processing occurs, including DL/I database calls. All processing is completed, a sync point is taken, and an output message is sent.

Figure 1: Unit of work example

Log Analyzer identifies LUOWs to provide more powerful, meaningful insights into complex activities than a view of isolated UOWs can provide. An LUOW captures the entire application flow across boundaries of UOWs, message switches, IMS systems, sysplexes, and LPARs. Processing on non-IMS platforms is also represented in an LUOW to give you a complete view of the application flow.
The following figure shows typical activities in an LUOW.

**Figure 2: Logical unit of work example**

For this LUOW example, two sysplex environments have three logical partitions (LPARs). Sysplex A, with LPAR 1 and LPAR 2, is running the IMS1 system and the IMS2 system. Sysplex B, with LPAR3, is running the IMS3 system. A transaction can enter IMS from any of several sources, such as IMS/TM, IBM CICS-DB/CTL, or the web. Processing for this LUOW proceeds as follows:

1. A transaction request (represented by UOW1) is issued to IMS1, and UOW1 is processed according to the UOW example.

2. A message switch occurs to another transaction (represented by UOW2) that executes on IMS2. IBM DB2 access is performed in UOW2 before the transaction is completed.
3 Another message switch occurs (represented by UOW3) to request information in IMS3. Processing in WebSphere for MQ is performed in UOW3.

4 A final message switch occurs (represented by UOW4), and information is passed back to the originating IMS1. UOW4 is executed, and the results are returned to the requestor.

Log record selection

If you do not have expert knowledge of log records or if you prefer to focus on search criteria other than log record types, you can use Log Analyzer intuitively by omitting log record specifications.

Log Analyzer examines your search criteria and automatically identifies and selects relevant types of log records for the search.

If you know exactly which log records are of interest for your purposes, you can specify log record types and subtypes to include or exclude.

In either case, Log Analyzer provides readily accessible information about log record codes to help you identify unfamiliar record types.

Flexible search criteria

With Log Analyzer, you can search log records and LUOWs for a broad range of targets, such as user IDs, transactions, database names, node names, origins and destinations, and specified content in data and other fields.

Log Analyzer can find LUOWs that have abended and LUOWs with overall elapsed times that fall outside of specified boundaries. through Boolean logic supports complex searches on combinations of targets.

Log record sources

Log Analyzer can read log records from a variety of sources, including:

- IMS system log data sets (SLDSs)
- IMS online log data sets (OLDSs)
- Log Analyzer extract data sets (containing log records that were extracted from log sources in a previous execution of Log Analyzer)
Extract log data sets that were created by a third-party utility (containing IMS log records)

DB2 logs containing DB2 unit of recovery (UR) information

Journals containing IMS Connect information from the BMC Energizer for IMS Connect product

Extracted log data from an IBM WebSphere for MQ (formerly known as IBM MQSeries) environment

Log Analyzer can read multiple sources in a single analysis job.

Log Analyzer can identify and allocate many sources of log records automatically from system resources such as IMS RECON data sets and DB2 bootstrap data sets.

Interactive LUOW analysis

When Log Analyzer has performed a batch analysis of your log data and has created LUOWs that match your search criteria, you can use the Log Analyzer ISPF interface to analyze LUOWs and log records interactively.

With unprecedented speed and ease, you can zoom in on details of objects, zoom out on wider views of an object in context with other objects, switch from one view of an object to a different view, and switch from a view of one object to a different object. Analysis tasks that could take days with traditional manual methods can be performed in minutes with interactive methods.

Predefined reports

Log Analyzer includes predefined batch and ISPF reports that are sensible, informative, and expertly designed to meet the requirements of the people who use them. The extensive design and setup work that other products require has already been performed for you.

LUOW-based reports

Log Analyzer can produce the following reports that are organized by LUOWs to provide a logical, integrated view of the actions and events that occurred in the specified time frame:

- The Analysis Summary report summarizes information about the log records that Log Analyzer handled during the analysis job, about how Log Analyzer has
organized these records into LUOWs, and about the types of log records that were read.

- The Auditing Report provides information that is relevant to an audit of events.
- The LUOW Summary report summarizes information about the LUOWs that Log Analyzer created during the analysis job.
- The LUOW Detail report lists key information about log records that Log Analyzer associated with an LUOW.
- The Record Dumps report displays the contents of log records in a standard format or a vertical format.
- The Time Sequence report lists log records that have been selected for analysis in the order in which they were written to an IMS log.
- The Orphans report provides information about records for which Log Analyzer was unable to resolve associations and relate the record to an LUOW.
- The Application Checkpoints report identifies application programs that issue too many or too few checkpoints. The report supports both basic and extended checkpoints. You can use this information to tune the frequency of checkpoints, which helps to ensure optimum data availability, application performance, and resource usage.

**Single-record reports**

Log Analyzer can produce single-record reports that are organized by individual log records instead of logically related sets of records (LUOWs).

To produce these reports, Log Analyzer evaluates one record at a time and can process a large amount of input data without incurring memory shortages. The following single-purpose reports are available:

- The ABEND report analyzes all type 07 records in the input log sources to identify transactions or programs that have terminated abnormally. This report displays type 07 records without an LUOW context, producing a streamlined view of the information. For greater depth and understanding, you can request a report of abend information within the context of LUOWs.

- The Deadlocks report provides information about record lockout situations, also known as a deadly embrace between two or more transactions. You can use this information to identify LUOWs that were involved in a deadlock, to analyze specific deadlocks and determine causes and effects, and to analyze the general frequency and severity of deadlocks in an environment.
The RBA Buffer Statistics report provides information about the distribution of updated relative byte addresses (RBAs) or block numbers in full-function IMS databases. You can use this information to tune buffers for the EXTENDED BUFFER MANAGER (XBM) Snapshot Copy function from BMC.

**Information about read-only databases in LUOWs**

Log Analyzer can include information about read-only activity against IMS databases in logical units of work (LUOWs).

Because IMS does not log this read-only activity, previously you had no way to determine the amount of time that transactions spent reading a database, nor to identify which transactions accessed the database for read-only purposes. Working with the records that the BMC MainView for IMS Offline product creates, Log Analyzer provides a complete picture of all database activity for accurate troubleshooting and auditing results.

**Application checkpoint information**

Log Analyzer can identify application programs that issue too many or too few checkpoints. The report supports both basic and extended checkpoints. You can use this information to tune the frequency of checkpoints, which helps to ensure optimum data availability, application performance, and resource usage.

**Deadlock information**

Log Analyzer can provide information about record lockout situations, also known as a deadly embrace between two or more transactions. You can use this information to identify LUOWs that were involved in a deadlock, to analyze specific deadlocks and determine causes and effects, and to analyze the general frequency and severity of deadlocks in an environment.

**Outstanding customer support**

BMC backs its products with outstanding customer support. Expert help is always on call. You can speak directly with a support specialist who has years of experience with IMS systems. You can e-mail Customer Support, and you can obtain technical support on the Customer Support web page or by telephone 24 hours a day, seven days a week.
A batch way to start using Log Analyzer

Log Analyzer executes as a batch utility. It reads log data sets, identifies and organizes log records of interest, and writes the results of the analysis to batch reports and output data sets (depending on the control statement keywords that you specify). The following procedure outlines a batch way to start using Log Analyzer (after it is installed and configured in your environment).

To start using Log Analyzer in batch

1. Prepare the JCL to execute an analysis job.

   To get started, you can copy and modify samples named LUI#JCLx from the product control library and samples named LUIxxxxx from the sample library. “Working with Log Analyzer jobs” on page 55 provides complete details about Log Analyzer JCL and control statements. “Sample and syntax reference” on page 337 provides an annotated list of Log Analyzer samples.

2. Submit the prepared JCL for execution.

3. View the results of the job.

   Log Analyzer batch reports are written to SYSOUT-type data sets. “Working with batch reports” on page 175 provides complete details about Log Analyzer batch reports.

A simple example

The following example shows a simple Log Analyzer job step:

```//ANALYZE EXEC PGM=LUIMAIN,REGION=OM //STEPLIB DD DISP=SHR,DSN= //          DD DISP=SHR,DSN= //          DD DISP=SHR,DSN= //SYSOUT DD SYSOUT=* //SYSIN ANALYZE SLDS=(ims.log.data.set.Rvr) FILTER LOGRECORDCODES=(ALL,¬48) END```

The JCL and control statements in this example tell Log Analyzer to perform the following actions:

- Read all records from a specified input log data set (ims.log.data.set) that was created by the specified version of IMS (Rvr)

   The input log can be a system log data set (SLDS) or an online log data set (OLDS). You can specify multiple input log data sets by using multiple SLDS
keywords. If you specify other keywords, Log Analyzer can read only a portion of the input data set.

- Consider all types of log records (except padding records)
  You can specify any type of log record to include in or exclude from the search.
  Log Analyzer automatically includes log record types that are related to specified record types.

- Produce the Analysis Summary report (because no REPORTS, EXTRACT, or INDEXFILE control statements were specified)

The following example shows a portion of the Analysis Summary report. Log Analyzer can produce other types of batch reports.

### Figure 3: Analysis Summary report

<table>
<thead>
<tr>
<th>2008-139</th>
<th>Log Analyzer for IMS V1.2.00.00</th>
<th>Page 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Analysis Summary (SUMMARY)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOG time span: FROM 2008-030 00:13:19.7 TO 2008-030 00:17:53.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>**********************</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Record and LUOW Counts</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log records read 00006049</td>
<td>Passed to select process 00005972</td>
<td>Selected log records 000005971</td>
</tr>
<tr>
<td>LUOWs 00001381</td>
<td>Unresolved associations 00000001</td>
<td>Extract file count 00000000</td>
</tr>
<tr>
<td>IMSIDs encountered: I9Z</td>
<td></td>
<td></td>
</tr>
<tr>
<td>********************</td>
<td></td>
<td></td>
</tr>
<tr>
<td><em><em>LUOW Lists (</em> indicates abend occurred in some LUOWs)</em>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Origin summary</td>
<td>Destination summary</td>
<td>User ID summary</td>
</tr>
<tr>
<td>Total number of Origins 00008002</td>
<td>Dests 00008003</td>
<td>Total number of Users 00000002</td>
</tr>
<tr>
<td>LUOWs 00007332</td>
<td>LUOWs 00007332</td>
<td>LUOWs 00007332</td>
</tr>
<tr>
<td>-Origin -LUOWs-</td>
<td>-Dests -LUOWs-</td>
<td>-User -LUOWs-</td>
</tr>
<tr>
<td><em>CICS</em> 0000632</td>
<td><em>CICS</em> 0000632</td>
<td>nonusid 0000100</td>
</tr>
<tr>
<td>ims 0000100</td>
<td>DSFSTCFI 0000001</td>
<td>BOLTSMI 0000632</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>... data omitted ...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### A robust example

The following example shows a Log Analyzer job that employs many of the powerful and flexible features that you can request:

```sql
//LUITEST1 JOB standard job statement
//ANALYZE EXEC PGM=LUIMAIN,REGION=OM
//STEPLIB DD DISP=SHR,DSN=bmclui.loadlib
// DD DISP=SHR,DSN=bmclui.BBLINK
// DD DISP=SHR,DSN=bmclui.ICOLIB
// DD DISP=SHR,DSN=ims.mdalib
//SYSDUMP DD DSN=your.sysmdump.dataset,DISP=(,DELETE,CATLG), X
// UNIT=SYSDA,SPACE=(CYL,(0,100),RLSE), X
// RECFM=FB,LRECL=4160,BLKSIZE=4160
//SYSDOUT DD SYSOUT=* **
//SYSIN DD *
//ANALYZE IMSID=imsid
//INTERVAL START=yyyyyyddd/hmmsssth STOP=yyyyyyddd/hmmsssth
//FILTER SELECT=(USERID=userId)
//REPORTS SUMMARY=ALL AUDIT=ALL
```
The JCL and control statements in this example tell Log Analyzer to perform the following actions:

- Allocate a set of RECON data sets by using a member of the IMS dynamic allocation (MDA) library
  The MDA library is not used if you specify the names of the RECON data sets or the names of the log data sets to read.

- From RECON COPY1, obtain a list of log data sets that were written by the specified IMS system (imsid) during the specified time interval (yyyyddd/hhmmssth)
  RECON access is not required if you specify the names of the log data sets to read.
  Log Analyzer can process log data sets for multiple IMS systems and multiple intervals.

- Find all records that are related to the specified user
  Log Analyzer supports many types of search criteria, including destination (transaction), origin (logical terminal), user, database name, application name, and so on. You can search for a specified string in the record content. You can combine search criteria with Boolean logic.

  In this example, Log Analyzer identifies relevant log record types automatically.
  Log Analyzer supports a specified keyword that you can use to specify log record types and subtypes which are eligible for the search.

- Produce all lines of batch reports that describe the results of the analysis
  You can control the types of batch reports that are produced and the number of lines that are written to the reports.

- Produce an extract file that contains all records which met the search criteria
  You can use this extract file as input to a subsequent analysis job.

The following example shows a small portion of data from many of the available reports. (This data was not produced by the job in the previous example.)
An interactive way to use Log Analyzer

Although log analysis is always executed as a batch utility, Log Analyzer provides an ISPF interface that can help novice and experienced users get results quickly and easily.

The following procedure outlines an interactive way to start using Log Analyzer (after it is installed and configured in your environment).

To start using Log Analyzer in the ISPF interface

1. Access the ISPF interface.

An interactive way to use Log Analyzer

2007-335
Log Analyzer for IMS V1.1.02
Page 2
If the installer used the sample CLIST to provide access to the Log Analyzer ISPF interface, you can enter the following command in the **Command** area of any ISPF panel that supports TSO commands:

**TSO %LUICI@00**

2 Select control statements and keywords, and enter keyword values.

The Build JCL panel (**Figure 5 on page 31**) provides a central location for accessing the panels that you use to select control statements and keywords and enter keyword values. Although a general knowledge of Log Analyzer JCL and control statements can be helpful for this task, it is not required. You can press **F1** to view Help panels for explanations of everything that you need to know.

**Figure 5: Build JCL panel**

```
File  Edit  Options  Help
-----------------------------------------------------------------------------
Log Analyzer                       Build JCL
Command ===> _________________________________________________________________

Control statements. Type an action code. S=Update, /=Include, blank=Exclude
Note: A / indicates the control statement will be included in the generated JCL
/ ANALYZE   - Specify which logs to read
/ FILTER    - Specify which LUOWs to keep
/ INTERVAL  - Specify the time frames within the logs
/ REPORTS   - Specify which reports to produce
/ EXTRACT   - Specify the output SLOs extract file
/ INDEXFILE - Specify the output report index file
/ WORKFILE  - Specify the sort work file allocation information

Job card.
//PAGTEST JOB (5711),LOG ANALYZER*,CLASS=Q,MSGCLASS=X,TIME=1440,
 // NOTIFY=&SYSUID
```

**STEPLIB libraries.**

'BMCLUI.LOADLIB'
'BMCLUI.BBLINK'
'BMCLUI.ICOLIB'

3 Generate the JCL, and submit it for execution.

When you have finished selecting control statements and keywords and entering keyword values, enter the **VIEW** command in the command area. The ISPF interface uses the information that you have provided to generate the analysis JCL and display it in a temporary data set. You can view the generated statements, modify them, and submit the JCL for execution.
The log analysis job is executed, and the job performs the same actions as described in “A simple example” on page 27 and “A robust example” on page 28. Almost all of the data gathering, calculations, and preparation for data presentation are performed during this batch job; relatively few resources are needed for working with the results in the interface.

4 View the results of the job.

With the Log Analyzer ISPF interface, you can work with the results of an analysis job interactively. You can navigate between different logical units of work, log records, and levels of detail. You can also change the sort order and filter out (hide) information dynamically.

The LUOW List panel (Figure 6 on page 32) provides summary information about the LUOWs that were created.

**Figure 6: LUOW List panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>LUOW List</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>Row 61 of 3041</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command ===&gt;</td>
<td>Scroll ===&gt;</td>
<td>CSR</td>
<td></td>
</tr>
</tbody>
</table>

Report index file: 'LXM.LUI.INDEX1'

Type one or more action codes. Enter LEFT or RIGHT for additional columns.

S=LUOW detail  L=List log records

Tran/Sort/Options/Help

A LUOW Id Origin LTERM Userid Timestamp Elapsed Time

From the LUOW List panel, you can select an LUOW to view more details about it or to view a list of log records that were included in the LUOW (Figure 7 on page 32).

**Figure 7: LUOW luowid Log Records panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>LUOW 00000001 Log Records</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>Row 1 of 21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command ===&gt;</td>
<td>Scroll ===&gt;</td>
<td>CSR</td>
<td></td>
</tr>
</tbody>
</table>

Report index file: 'RIHDGW.LUI.JIMIPR.INDEX'

Type one or more action codes. Enter LEFT or RIGHT for additional columns.

S=Log record detail  T=List adjacent records

D=Dump record  B=Browse record  ?=Log code help  Sorted by: TIMESTAMP
From the LUOW luowid Log Records panel, you can select a log record to dump or browse, or you can see the other log records that were created before and after the selected record (Figure 8 on page 33).

Figure 8: Log Records - Time Sequence panel

“Working with reports in the ISPF interface” on page 211 provides complete details.

Considerations

Certain considerations apply to Log Analyzer.

CPU authorization passwords for Log Analyzer

A valid CPU authorization password is required for executing a Log Analyzer analysis job.

Log Analyzer accepts passwords for the following products:

- BMC System Administration solution for IMS

  Passwords for this product are created with the IPT product code.
BMC System Communication solution for IMS
Passwords for this product are created with the CSU product code.

BMC Log Analyzer product for IMS
Passwords for this product are created with the LUI product code.

If multiple passwords are installed, Log Analyzer checks for an IPT password first, then checks for a CSU password, and finally checks for a LUI password.

For information about obtaining, installing, and maintaining passwords, see the Installation System documentation

**APF authorization**

If the Log Analyzer analysis job is accessing any RECONs that have been upgraded to IMS Version 10.1 or later, APF authorization is required for all data sets that are included in the STEPLIB concatenation of the analysis job.

APF authorization is not required if all RECONs are at IMS Version 9.1 or earlier or if the analysis job is accessing no RECONs. APF authorization is not required for any functions of the Log Analyzer ISPF interface.

**Security access controls**

Log Analyzer reads IMS logs and can display the contents of IMS log records. To prevent unauthorized access to the information in IMS logs, you should implement the same access controls for Log Analyzer extract files and index files as for IMS system log data sets and online log data sets.

Your user ID must have Security Access Facility (SAF) READ access to the RECON data sets if either of the following conditions exists:

- You specify the RECON keyword.
- Log Analyzer reads the RECON data sets to determine logs to allocate through implicit specification.

**Resource consumption and production workloads**

A Log Analyzer analysis job consumes as much memory and CPU cycles as it needs and as are available. Depending on the number of input log records and on specified
filter criteria, this resource consumption could adversely affect other jobs that are running in the system.

Consider running the analysis job on a test or development system. If you must run the process on a production system, BMC recommends that you implement workload manager parameters, dispatch priorities, or both to ensure that the process receives the desired percentage of resources.

**Nonunique IMSIDs**

In a Multiple Systems Coupling (MSC) environment, a problem can occur if the IMSIDs of participating IMS systems are not unique.

The problem occurs only if local transactions are processed against a remote MSC-connected system that has the same IMSID; if the remote IMS system is used only to route transactions and process routed transactions, the problem does not occur.

If Log Analyzer is analyzing records from multiple connected IMS systems and those systems do not have unique IMSIDs, Log Analyzer might associate log records with a single LUOW when those log records have no logical relationship to each other. No error is detected in this case.
Getting started with Log Analyzer

This chapter describes how the BMC Log Analyzer component for IMS works and helps you get started with basic Log Analyzer tasks.

Task summary for using Log Analyzer

Following is a summary of the actions that you complete in order to use Log Analyzer.

**Note**
For more information, see the detailed procedure in “Using Log Analyzer” on page 44.

1. Create JCL and control statements to execute a Log Analyzer analysis job.
   You can specify JCL and control statements manually or generate them through the Log Analyzer ISPF interface. Control statement keywords provide a simple but powerful method for specifying the sources of log records to read, the log records to select for analysis, the reports and files to produce, and the options to use during processing.

2. Submit the JCL to execute the analysis job.
   The analysis job reads and processes log records from specified log record sources, creates logical units of work (LUOWs), and produces output data sets that contain the results of the analysis.

3. Work with the output of the analysis job.
   The analysis job can create batch reports, an index file, and an extract file. Batch reports contain information about LUOWs and log records. Batch reports typically are written to SYSOUT-type data sets, which you can view through SDSF or any JES spool display utility. The index file contains information about LUOWs and log records. You can view and work with this information through the Log Analyzer ISPF interface. The extract file contains the log records that are associated with LUOWs of interest. You can view information in this file through the ISPF interface or use the file as input to a subsequent analysis job.
Log Analyzer architecture and processes

The following figure shows the components of Log Analyzer and their interaction with elements in the IMS environment.

Figure 9: Components of Log Analyzer
Log Analyzer JCL and control statements

To analyze log records automatically and create LUOWs, you must submit an analysis batch job.

You control the execution of the analysis job by using standard JCL and by using control statements that are specific to Log Analyzer. For a typical example, see the following JCL:

**Figure 10: Typical Log Analyzer job**

```plaintext
//PAGTEST JOB (5711),MSGCLASS=X,CLASS=Q,MSGLEVEL=(1,1), NOTIFY=&SYSUID,TIME=1440,REGION=OM
//LOGTEST EXEC PGM=LUIMAIN,REGION=OM
//STEPLIB DD DISP=SHR,DSN= bmclui.loadlib
// DD DISP=SHR,DSN= bmclui.BBLINK
// DD DISP=SHR,DSN= bmclui.ICOLIB
//SYSDUMP DD DSN=your.sysmdump.dataset,DISP=(,DELETE,CATLG), UNIT=SYSDA,SPACE=(CYL,(0,100),RLSE), RECFM=FB,LRECL=4160,BLKSIZE=4160
//SYSOUT DD SYSOUT=*  
//SYSIN DD SYSOUT=*  
//ANALYZE SLDS=(IMS1.DLSD480,R91) FILTER SELECT=TRAN=MYTRAN REPORTS SUMMARY=ALL LUOWSUMM=ALL LUOWDETAIL=ALL LUOWTSEQ=10000 END
```

Log Analyzer JCL contains a standard JOB statement, an EXEC statement that executes program LUIMAIN, a few DD statements that allocate system data sets, and a SYSIN control statement data set. Commands, keywords, and values in the control statement data set specify input log record sources to read, criteria to filter log records and LUOWs, output reports to produce, output data sets to allocate, and other options to use during the analysis job.

You can use the following techniques to work with Log Analyzer JCL:

- Create the JCL manually. Log Analyzer provides JCL samples as members named LUI#JCLx in the product control (CNTL) library. For more information, see “Working with Log Analyzer jobs” on page 55.
- Generate JCL through the Log Analyzer ISPF interface. For more information, see “Building Log Analyzer JCL through the ISPF interface” on page 137.

**Syntax processing**

When you submit the analysis job for execution, Log Analyzer reads control statements from the SYSIN data set and checks to ensure that the statements are correct and complete. It writes syntax error messages to the SYSOUT data set.
Log record source processing

When syntax checking is complete, Log Analyzer builds a list of sources that contain the log records to be analyzed. These sources can include:

- System log data sets (SLDSs)
- Online log data sets (OLDSs)
- Log Analyzer extract data sets (containing log records that were extracted from log sources in a previous execution of Log Analyzer)
- Extract log data sets that were created by a third-party utility (containing IMS log records)
- Logs containing DB2 log records
- Journals containing IMS Connect information from the BMC Energizer for IMS Connect product
- Extract files that contain log data from an IBM WebSphere for MQ (formerly known as IBM MQSeries) environment

To identify the log record sources to be processed, use either of the following techniques:

- Implicit specification
  To use implicit specification, you omit specific log record sources from the control statements. Log Analyzer allocates log record sources from entries in the IMS RECON. Log Analyzer can identify the RECON data set names from IMS dynamic allocation (MDA) members, or you can provide RECON data set names with a control statement. This technique is valid for SLDSs and OLDSs.
  If you want to use implicit specification for DB2 logs, you omit specific DB2 log data set names and specify one or more DB2 bootstrap data sets. This technique is valid for DB2 logs.
  You can also use implicit specification for Energizer journals. Log Analyzer can determine the journals to use by searching for a specified data set name prefix in the system catalog and using journals that match the time range of the analysis.

- Explicit specification
  To use explicit specification, you provide the data set name of the source with a control statement. This technique is valid for SLDSs, OLDSs, extract data sets, Energizer journals, DB2 logs, and WebSphere for MQ extract files.

For more information about log record sources, see “Allocating log record sources” on page 249.
Log record processing

When log record sources have been identified, Log Analyzer performs the following tasks to process log record:

- Reads log records from log record sources
- Determines whether log records are candidates for the selection process and passes candidate log records through a filtering mechanism to determine whether each record should be selected or rejected

You can allow Log Analyzer to identify candidate log record types automatically, based on the search criteria (such as logical terminal ID, a user ID, or a database name) that you specify. Or you can specify the range, types, and subtypes of log records that are candidates for the log record selection process. For more information, see “Identifying candidate records and creating LUOWs” on page 269 and “Selecting log records and LUOWs” on page 283.

To produce certain reports, Log Analyzer automatically selects the required log records that are required for that type of report.

LUOW processing

When Log Analyzer is producing LUOW-based reports and identifies a candidate log record, Log Analyzer associates the candidate with an LUOW.

An LUOW is a set of related log records that represent all of the actions that were performed to accomplish a particular activity, such as executing a transaction, in the IMS environment. Log Analyzer associates log records with an LUOW by matching the values of certain fields in those records.

During LUOW processing, Log Analyzer determines whether each LUOW (and its associated records) should be retained as an LUOW of interest or discarded.

For more information about how LUOWs are created, see “Selecting log records and LUOWs” on page 283.

If producing a single-record report, Log Analyzer bypasses LUOW processing.

Output processing

Log Analyzer prepares the results of the analysis job and writes them to the following output data sets as requested:
Log Analyzer writes batch reports to SYSOUT-type data sets. These reports provide information about the LUOWs that were created (if applicable) and the log records that were processed during the analysis job. You must use a control statement to specify whether and how to produce batch reports.

If producing LUOW-based reports, Log Analyzer writes information about LUOWs and associated log records to an index file. The index file is required for access to analysis data when you are analyzing LUOWs through the ISPF interface. You must use a control statement to specify whether and how to allocate this file.

If producing LUOW-based reports, Log Analyzer writes copies of all log records that were associated with LUOWs of interest to an extract file, which can be used as input to a subsequent analysis job. You can examine the data in this file through the Log Analyzer ISPF interface. You must use a control statement to specify whether and how to allocate this file.

For more information about how these output data sets are created and used, see “Working with batch reports” on page 175, “Working with reports in the ISPF interface” on page 211, and “Working with analysis data” on page 291.

Log Analyzer reports

The following table lists major types of information that is available in Log Analyzer reports and indicates the location of the information within batch reports and the ISPF interface.

Information is similar between the batch reports and ISPF interface. Some reports are based on logical units of work (LUOWs); others are based on a single type of log record.

**Note**

To produce LUOW-based reports, Log Analyzer holds log records in memory while constructing LUOWs. For these reports, memory shortages can occur if Log Analyzer attempts to process a large amount of input data.

In contrast, Log Analyzer evaluates one record at a time to produce DEADLOCK, RBASTATS, and ABEND reports. For these reports, Log Analyzer can process a large amount of input data without incurring memory shortages.

For more information, see “Working with batch reports” on page 175 and “Working with reports in the ISPF interface” on page 211.
<table>
<thead>
<tr>
<th>Type of information</th>
<th>Type</th>
<th>Batch report</th>
<th>ISPF panel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information about transactions and programs that have terminated abnormally</td>
<td>Single-record</td>
<td>Abend</td>
<td>None</td>
</tr>
<tr>
<td>Facts about the overall results of the analysis job</td>
<td>LUOW-based</td>
<td>Analysis Summary</td>
<td>Report Index File Summary</td>
</tr>
<tr>
<td>Information about application checkpoints</td>
<td>LUOW-based</td>
<td>Application Checkpoint</td>
<td>None</td>
</tr>
<tr>
<td>Information relevant to an audit of events in the environment</td>
<td>LUOW-based</td>
<td>Auditing</td>
<td>None</td>
</tr>
<tr>
<td>Information about deadlocks</td>
<td>Single-record</td>
<td>Deadlocks</td>
<td>Deadlock Victims List</td>
</tr>
<tr>
<td>List of LUOW that were created</td>
<td>LUOW-based</td>
<td>LUOW Summary</td>
<td>LUOW List</td>
</tr>
<tr>
<td>Summary information about LUOWs</td>
<td>LUOW-based</td>
<td>LUOW Summary</td>
<td>LUOW List</td>
</tr>
<tr>
<td>Destination List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Origin List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Userid List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other LUOWs List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailed information about an LUOW</td>
<td>LUOW-based</td>
<td>LUOW Detail</td>
<td>LUOW luowid</td>
</tr>
<tr>
<td>List of log records that are associated with an LUOW and summary information about those records</td>
<td>LUOW-based</td>
<td>LUOW Detail</td>
<td>LUOW luowid Log Records</td>
</tr>
<tr>
<td>Detailed information about a log record</td>
<td>LUOW-based</td>
<td>LUOW Detail</td>
<td>Log Record Detail</td>
</tr>
<tr>
<td>Distribution of updated relative byte addresses (RBAs) or block numbers in full-function IMS databases</td>
<td>Single-record</td>
<td>RBA Buffer Statistics</td>
<td>None</td>
</tr>
<tr>
<td>Dump of a log record</td>
<td>LUOW-based</td>
<td>Record Dumps</td>
<td>Log Record logrecnum</td>
</tr>
<tr>
<td>List of log records in context with other log records in the same time frame</td>
<td>LUOW-based</td>
<td>Time Sequence</td>
<td>Log Records - Time Sequence</td>
</tr>
</tbody>
</table>

**Table 1: Location of information in reports**
Using Log Analyzer

To use Log Analyzer, you must prepare and submit an analysis job, work with resulting LUOWs (if applicable) and log records, and (if necessary) refine results.

Before you begin

Complete the following tasks:

- Install and customize the product libraries. For more information, see the Installation System documentation.

- Configure product components. Configuration tasks include handling the product authorization password and configuring the CLIST that is used to invoke the ISPF interface. For more information, see the customization guide.

- Depending on how security is managed for resources in your environment, you might need to be granted authority to create Log Analyzer files. For more information, see the customization guide.

To use Log Analyzer

1. Create a batch job that performs an automated analysis of your data by using either of the following methods:

   - Specify JCL and control statements manually by using ISPF EDIT functions. For more information, see “Working with Log Analyzer jobs” on page 55.
     Log Analyzer provides samples that you can copy and modify. For more information, see “Samples” on page 337.

   - Build JCL and control statements interactively by using the Log Analyzer ISPF interface. For more information, see “Building Log Analyzer JCL through the ISPF interface” on page 137.

2. Submit the analysis job for execution.

If you have selected LUOW-based reports, the result of the analysis job is a set of LUOWs that Log Analyzer has created and the log records that are associated
with those LUOWs. Log Analyzer can also write batch reports and prepare information for analysis through the ISPF interface.

If you have selected a single-record report, the result of the analysis job is a batch report that contains information about that record.

3 Work with the information by using an applicable method:

- If you have selected LUOW-based reports, you can view batch reports that provide information about LUOWs and their associated log records. For more information, see “Working with batch reports” on page 175.

- If you have selected LUOW-based reports, you can use the Log Analyzer ISPF interface to work with LUOWs and log records interactively. For more information, see “Working with reports in the ISPF interface” on page 211.

- If you have selected a single-record report, you can view the batch report. For more information, see “Working with batch reports” on page 175.

4 If the automated analysis did not produce all of the results that you want, change the control statements in the JCL and continue the process at Step 2 on page 44.

For suggestions about changes to achieve the results that you want, see “Refining the results of automated analysis” on page 305.

Where to go from here

If a problem (such as abnormal termination) occurs, see “Troubleshooting and FAQs” on page 329.

Working with the Log Analyzer ISPF interface

Log Analyzer includes an ISPF interface that novice and expert users can use to obtain results quickly and with little effort. You can use the Log Analyzer ISPF interface to work with Log Analyzer JCL and obtain LUOW information.

Overview of the ISPF interface

The ISPF interface provides the following functions:

- JCL functions

The ISPF interface can build JCL to execute a Log Analyzer analysis job. You enter information and choose values that control how Log Analyzer selects log record
sources, filters and extracts log records, produces reports, and handles external files. Using the supplied information, the interface generates JOB, EXEC, DD, and control statements so that you do not have to understand and meet syntax requirements. The interface can display generated JCL, which you can view, edit, and submit.

- Analyze functions

The ISPF interface can display information about LUOWs and their component log records. You can view lists of LUOWs sorted by various types of objects (such as destination, origin, and user ID), filter LUOWs to limit lists, view details about LUOWs and log records, and view records that are adjacent to a selected record. Interactive analysis of LUOWs through the ISPF interface can be faster and easier than analysis of LUOWs through batch reports.

**Note**
Values for many fields in the ISPF interface are saved in your ISPPROF data set. Initial values for these fields can be set in member LUIZUSER of the Log Analyzer panel library. These initial values can be set for your environment during Log Analyzer configuration, as described in the configuration guide.

## Accessing the ISPF interface

Access the Log Analyzer ISPF interface with a TSO command or a selection from an ISPF panel.

**To access the ISPF interface**

1. Invoke the ISPF interface by using the TSO command (or the selection from an ISPF panel) that was implemented by the installer as the method for accessing Log Analyzer.

   If the installer used the sample CLIST to provide access to the Log Analyzer ISPF interface, you can enter the following command in the **Command** area of any ISPF panel that supports TSO commands:

   ```
   TSO %LUICI@00
   ```

   For more information about how to invoke the ISPF interface in your environment, contact the person who installed Log Analyzer.

   The first time that you invoke the ISPF interface, the Log Analyzer *for IMS* panel is displayed. This panel, also known as the Logo panel, provides copyright information about Log Analyzer.
To bypass the Log Analyzer for IMS panel on subsequent access to the interface, type a slash (/) in the **Suppress display of this panel on startup** field or set a user preference, as described in “Setting user preferences” on page 52. The next time that you invoke the ISPF interface, the Main Menu is displayed instead of the Log Analyzer for IMS panel.

2 On the Log Analyzer for IMS panel, press **Enter**.

The Main Menu is displayed as follows:

**Figure 11: Main Menu**

<table>
<thead>
<tr>
<th>Menu</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>Main Menu</td>
<td></td>
</tr>
<tr>
<td>Command ===&gt;</td>
<td>_________________________________________________________________</td>
<td></td>
</tr>
</tbody>
</table>

Log Analyzer option. Choose a selection.

- 1. JCL - Build, view, submit JCL for extraction and analysis
- 2. Analyze - Analyze a report index file

3 In the **Log Analyzer option** field, enter one of the following values:

- **1** Access the JCL functions of the ISPF interface. For more information, see “Building Log Analyzer JCL through the ISPF interface” on page 137.
- **2** Access the Analyze functions of the ISPF interface. For more information, see “Working with reports in the ISPF interface” on page 211.

**Navigating between panels**

In general, you can take the following actions to navigate from one panel to another panel:

- To display the next panel in a hierarchical sequence of panels, type a selection in a choice entry field or a selection field and press **Enter**.
- To save changes that you have made to data entry fields on a panel and display the previous panel in a hierarchy, enter the END command (which is usually assigned to **F3**).
To discard changes that you have made to data entry fields and display the previous panel in a hierarchy, enter the CANCEL command (which is usually assigned to F12).

**Scrolling through information on a panel**

If more information is available than can be displayed in a single view of a panel or list, a scroll indicator (More: +, More: -+, or More: -) indicates that more information is available, as shown in the following example panel.

**Figure 12: Scroll indicator**

![Scroll indicator example panel](image)

To view more information when the **More: +** indicator is displayed, use the DOWN command (which is usually assigned to F8). To view more information when the **More: -** indicator is displayed, use the UP command (which is usually assigned to F7). To view more information when the **More: -+** indicator is displayed, use the DOWN command or the UP command.

If the panel displays a table, a Row indicator (**Row n of n**) is displayed in the top right corner of the panel, as shown in the following example panel. The row indicator shows the number of the top row that is displayed and the total number of rows in the table.

**Figure 13: Row indicator**

![Row indicator example panel](image)

To scroll through the rows of a table, use the DOWN command and the UP command.
Using field prompts

A plus sign (+) to the right of a data entry field, as shown in the following panel example, indicates that a field prompt is available to automatically populate the field value.

**Figure 14: Panel with prompt field**

Specify which log records to include in the LUOWs.

| Included log codes | + |

**To use a field prompt**

1. Place the cursor anywhere within the data entry field and press **F4**.

   Instead, you can type the PROMPT command in the **Command** area, place the cursor in the data entry field, and press **Enter**.

   A pop-up window provides a selection list of valid values.

2. Select the value or values that you want to use.

3. Press **Enter**.

   The data entry field is populated with the selected values.

Using action bar commands

An action bar is displayed at the top of each panel in the ISPF interface. The menus on the action bar and the commands on the menus are relevant to the currently displayed panel.

**Figure 15: Action Bar**

<table>
<thead>
<tr>
<th>File</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
</table>

**To select an action bar command**

1. Place the cursor on the menu that you want to select and press **Enter**.

2. In the choice entry field on the pull-down menu, enter the number that corresponds to the command that you want to select.

   The following table lists action bar menus and commands that are available on most panels. Other menus and commands might be available, depending on the
panel. For information about other action bar menus and commands, see the online Help for the panel.

Table 2: Action bar menus and commands

<table>
<thead>
<tr>
<th>Menu</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File or Menu</td>
<td>Exit</td>
<td>Save any changes you that have made in data entry fields, and return to the previous panel. This command has the same effect as the END command, which is usually assigned to F3.</td>
</tr>
<tr>
<td>Options</td>
<td>Preferences</td>
<td>Set data-display preferences for Log Analyzer panels. For more information, see “Setting user preferences” on page 52.</td>
</tr>
<tr>
<td>Options</td>
<td>Colors</td>
<td>Set color and attribute preferences for Log Analyzer panels. For more information, see “Setting color and attribute preferences” on page 53.</td>
</tr>
<tr>
<td>Options</td>
<td>Function keys</td>
<td>Assign commands to function keys for Log Analyzer panels. For more information, see “Setting function-key preferences” on page 53.</td>
</tr>
<tr>
<td>Help</td>
<td>Help for help</td>
<td>View information about how to use the Help system.</td>
</tr>
<tr>
<td>Help</td>
<td>Extended Help</td>
<td>View Help information for the current panel.</td>
</tr>
<tr>
<td>Help</td>
<td>Log code help</td>
<td>View information about log records that are mapped by Log Analyzer.</td>
</tr>
<tr>
<td>Help</td>
<td>Messages</td>
<td>View information about Log Analyzer messages. For more information, see “Obtaining information about Log Analyzer messages” on page 330.</td>
</tr>
<tr>
<td>Help</td>
<td>About</td>
<td>View version and contact information about Log Analyzer.</td>
</tr>
</tbody>
</table>

Using global commands

The following table lists global commands that you can enter in the Command area of most panels. (You can also assign these commands to function keys.) Other commands might be available, depending on the panel. For information about other commands, see the online Help for the panel.

Table 3: Global commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABOUT</td>
<td>AB</td>
<td>View version and contact information about Log Analyzer.</td>
</tr>
<tr>
<td>Command</td>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ATTR</td>
<td>CO</td>
<td>Set color and attribute preferences for Log Analyzer panels. For more information, see “Setting color and attribute preferences” on page 53.</td>
</tr>
<tr>
<td>CANCEL</td>
<td>CAN</td>
<td>Discard changes, and display the previously displayed panel.</td>
</tr>
<tr>
<td>COLORS</td>
<td>CO</td>
<td>Set color and attribute preferences for Log Analyzer panels. For more information, see “Setting color and attribute preferences” on page 53.</td>
</tr>
<tr>
<td>COMMANDS</td>
<td>CMDS</td>
<td>View information about Log Analyzer global commands.</td>
</tr>
<tr>
<td>END</td>
<td>END</td>
<td>Save changes, and display the previously displayed panel.</td>
</tr>
<tr>
<td>HELP</td>
<td>HELP</td>
<td>Display Help information for the panel.</td>
</tr>
<tr>
<td>LOGCODES</td>
<td>LOG or LC</td>
<td>View information about the log records that are known to Log Analyzer.</td>
</tr>
<tr>
<td>MESSAGES nnnnn</td>
<td>MS nnnnn</td>
<td>View information about Log Analyzer messages. You can specify the optional parameter as follows:</td>
</tr>
<tr>
<td></td>
<td>MSGS nnnnn</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PREF</td>
<td>PR</td>
<td>Set user preferences for Log Analyzer panels For more information, see “Setting user preferences” on page 52.</td>
</tr>
<tr>
<td>VERSION</td>
<td>VR</td>
<td>View version and contact information about Log Analyzer.</td>
</tr>
</tbody>
</table>

**Setting preferences**

You can set preferences to control how the Log Analyzer ISPF interface displays information.
Setting user preferences

To set user preferences, enter the PREF command in the Command area or select the Preferences command from the Options action bar menu. The User Preferences pop-up window is displayed. Type values or select options in the panel fields, and press F3 to save your changes. The following table describes the fields on the User Preferences pop-up window.

Table 4: Fields on the User Preferences pop-up window

<p>| Area                              | Field | Description                                                                                                                                                                                                 |
|-----------------------------------|-------|--------------------------------------------------------------------------------------------------------------------------------============================================================================|
| Log Records - Time Sequence       | Before| Specify the default number of records (1 through 99999) to view on the Log Records - Time Sequence panel:                                                                                                      |
|                                   | After |                                                                                                                                                                                                           |
|                                   |       | ■ Use the Before field to control display of records that have earlier timestamps than the record that you select when you choose to list adjacent records.                                                      |
|                                   |       | ■ Use the After field to control display of records that have later timestamps than the selected record.                                                                                                   |
|                                   |       | For more information, see “Listing adjacent log records” on page 242.                                                                                                                                     |
| Log record dump panel options     | Format| Choose the default format to use for viewing dumped log records:                                                                                                                                               |
|                                   |       | ■ To view records in a standard dump format, select 1 (Standard).                                                                                                                                              |
|                                   |       | ■ To view records in an vertical format, select 2 (Over/ under) and select whether to display character or hexadecimal data.                                                                                |
|                                   |       | For more information, see “Dumping log records” on page 245.                                                                                                                                                |</p>
<table>
<thead>
<tr>
<th>Area</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Startup Preferences</td>
<td>Display Logo panel at Startup</td>
<td>To display the Log Analyzer for IMS panel when you first access the ISPF interface, enter a slash (/) in this field.</td>
</tr>
</tbody>
</table>

**Setting color and attribute preferences**

To set color, intensity, and highlight preferences for panel elements, enter the COLORS command or the ATTR command in the Command area, or select the Colors command from the Options action bar menu. The CUA Attribute Change Utility pop-up window is displayed. For information about how to use this window, see the online Help.

**Setting function-key preferences**

To set function-key preferences, select the Function keys command from the Options action bar menu. The Keylist Utility for LUI pop-up window is displayed. For information about how to use this window, see the online Help.

**Function-key definitions**

The Log Analyzer ISPF interface offers function-key definitions that are specific to individual panels. For example, UP and DOWN commands are assigned to function keys for panels that have scrollable areas, and the PROMPT command is assigned to a function key for panels that have promptable fields.

If you prefer to use your global set of function-key definitions for all panels in the ISPF interface, enter the KEYLIST OFF command on the Command line. To use the Log Analyzer-defined definitions, enter the KEYLIST ON command.

To view the function-key definitions that are in effect, enter the PFSHOW ON command. You can hide the definitions by entering PFSHOW OFF.

**Using online Help**

You can access Help information about elements of the ISPF interface as follows:
For information about a panel, place the cursor in the Command area or in any area of the panel that is not in the action bar or in a data entry field. Then enter the HELP command (which is usually assigned to F1).

Instead, you can select the Extended help command from the Help action bar menu.

For information about a field, place the cursor in the data entry area of the field (not in the field label), and enter the HELP command.

Instead, you can view the panel-level Help and navigate through links on that panel to the field Help.

For information about the commands on an action bar menu, place the cursor in the menu, and enter the HELP command.

For information about global commands, enter the COMMANDS command in the Commands area on any panel.

For information about how to use Help panels and pop-up windows, select the Help for help command from the Help action bar menu.

Instead, you can enter the HELP command when a Help panel or pop-up window is displayed.

Presetting profile variables

The Log Analyzer ISPF interface provides the LUIZUSER panel that a product administrator in your environment can use to preset profile variables. For information about how to preset profile variables, see the Installation System documentation.
Working with Log Analyzer jobs

To analyze log records automatically and create LUOWs (if applicable), you must execute an analysis batch job. You control the execution of the analysis job by using standard JCL and by using control statements that are specific to Log Analyzer.

Control statements specify:

- Sources of log records to read
- Log records to select for analysis
- Reports and files to produce
- Options to use during processing

You can prepare the JCL manually (as explained in this chapter), or you can use the Log Analyzer ISPF interface to build the JCL automatically from information that you specify on a series of panels (as explained in “Building Log Analyzer JCL through the ISPF interface” on page 137).

Following is a summary of the tasks you perform to work with a Log Analyzer job manually:

1. Specify required JOB, EXEC, and DD statements for the job.
   For more information, see “Specifying JOB EXEC and DD statements” on page 56.

2. Specify control statements in the SYSIN data set.
   For general information about Log Analyzer control statements, see “Specifying control statements” on page 59.
   For detailed information about commands and keywords that you can specify on control statements, see the following sections:
   - “Specifying ANALYZE control statements” on page 63
   - “Specifying INTERVAL control statements” on page 79
   - “Specifying FILTER control statements” on page 83
   - “Specifying REPORTS control statements” on page 107
Log Analyzer provides control statement samples that you can copy and modify for your own purposes. For an annotated index of samples, see “Samples” on page 337.

3 Submit the prepared job for execution.

4 Check the results of the job.

For more information, see “Checking results of Log Analyzer jobs” on page 135.

Specifying JOB EXEC and DD statements

JCL to execute a Log Analyzer job includes a JOB statement, an EXEC statement, and various DD statements. The following generic JCL example runs a Log Analyzer job.

```jcl
//jobname JOB ...
//*
//LUIMAIN EXEC PGM=LUIMAIN,REGION=0M
//STEPLIB DD DISP=SHR,DSN=bmclui.loadlib
// DD DISP=SHR,DSN=bmclui.BBLINK
// DD DISP=SHR,DSN=bmclui.icolib
//SYSDUMP DD DSN=your.sysmdump.dataset,DISP=(,DELETE,CATLG)
// UNIT=SYSDA,SPACE=(CYL,(0,100),RLSE)
// RECFM=FB,LRECL=4160,BLKSIZE=4160
//SYOUT DD SYSOUT=* 
//* PROVIDE THE FOLLOWING STATEMENT ONLY AT THE REQUEST OF BMC CUSTOMER SUPPORT
//LUITRACE DD SYSOUT=* 
//SYSIN DD *
// comment
ANALYZE keywords
INTERVAL keywords
FILTER keywords
REPORTS keywords
EXTRACT keywords
INDEXFILE keywords
WORKFILE keywords
END
//*
//
```

JOB statement

Provide a JOB statement that conforms to the standards of your environment.
EXEC statement

Provide an EXEC statement that specifies LUIMAIN as the program name.

The recommended region size is 0 MB. If your environment does not allow the specification of REGION=0M, specify the largest region size that is allowed.

Note
A Log Analyzer job can consume a large amount of system resources. To prevent resource contention that can affect production workloads, consider executing the Log Analyzer job on a nonproduction system or implementing workload management parameters, dispatch priorities, or both, to prevent overconsumption of resources.
For more information, see “Troubleshooting analysis jobs” on page 331.

STEPLIB DD statement

Provide a STEPLIB DD statement that includes the following modules in the data set concatenation, or ensure that the modules are available in the JOBLIB or LNKLST concatenation:

- Load modules for executing Log Analyzer processes
- Product authorization module for Log Analyzer
  This module contains the CPU authorization password that permits execution of an analysis job on a CPU. This module typically is located in the same library as Log Analyzer load modules. For information about valid passwords, see “CPU authorization passwords for Log Analyzer” on page 33.
- Load modules for executing product authorization processes
  These load modules are distributed in FMID BBAPW32 and typically are located in the BBLINK library after installation.
- Load modules for executing the standard BMC application programming interface (API) for DBRC
  Log Analyzer calls these load modules, including module BMCDBRC0, during analysis jobs that access the IMS RECON (when the ANALYZE statement includes the RECON keyword). These load modules are distributed in FMID XICO120 and typically are located in the ICOLIB library after a non-merged installation or in the IMLIB library after a merged installation.
- IMS dynamic allocation (MDA) members for allocating RECON data sets
  These members are required if you want Log Analyzer to discover and allocate
  RECON data sets automatically. Otherwise, they are optional.

**SYSMDUMP DD statement**

You can provide a SYSMDUMP DD statement to define a standard SYSMDUMP
data set.

**SYSOUT DD statement**

Provide a SYSOUT DD statement to contain an "echo" of the control statements that
are used to execute the job and any error and informational messages that are issued
during processing. This statement is usually specified to allocate a SYSOUT=* data
set.

**Note**

If you have requested a batch report by specifying a keyword on the REPORTS
control statement, Log Analyzer writes the report to a dynamically allocated output
data set. The ddname of this data set is the same as the keyword.

**LUITRACE DD statement**

At the request of BMC Customer Support, provide an LUITRACE DD statement to
contain trace data from the Log Analyzer job. This statement is usually specified as a
SYSOUT=* data set. If this statement is present, the Log Analyzer job can generate a
potentially large amount of output.

**SYSIN DD statement**

Provide a SYSIN DD statement to contain statements that control the Log Analyzer
job.

You can specify this statement to specify an in-stream data set (SYSIN DD *), a
standalone sequential data set, or a member of a partitioned data set. If you allocate
the data set to DASD, the record format (RECFM) must be fixed blocked (FB), and
the logical record length (LRECL) must be 80 bytes.
For information about how to specify control statements in the SYSIN data set, see “Specifying control statements” on page 59.

Specifying control statements

Specify control statements for a Log Analyzer job in the SYSIN data set.

Samples of control statements

Log Analyzer provides control statement samples that you can copy and modify for your own purposes.

For an annotated index of samples, see “Samples” on page 337.

The Log Analyzer ISPF interface can build JCL, and you can save and modify this JCL to create your own samples. For more information about using the interface for these tasks, see “Building Log Analyzer JCL through the ISPF interface” on page 137.

Control statement syntax

A Log Analyzer control statement consists of a command and, in many cases, one or more valid expressions. Each expression consists of a keyword and its valid value (you can provide multiple values for some keywords). The following figure shows generic syntax rules for Log Analyzer control statements.

Figure 17: Syntax of Log Analyzer control statements

* A comment has an asterisk in column 1 and is ignored.
* You can specify commands and expressions anywhere in columns 1-71.
* You can specify expressions for a command in any order.
  command keyword=value
  command keyword=value
  command akeyword = value    bkeyword=(value   value   value)
  command ckeyword = (value   value   value)    dkeyword=value
* You can specify commands, keywords, and values on separate lines.
  command
  keyword=value
  command keyword =
  keyword = (value1, value2, value3)
  keyword = (value1
             value2
             value3)
* You can continue values if necessary.
*---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
Syntax for Log Analyzer control statements is free-form, flexible, and easy to use. In general, you can specify keywords in any order in columns 1 through 71. You can use lines and spaces for enhanced readability. The following table explains syntax rules in detail.

**Table 5: Syntax of Log Analyzer control statements**

<table>
<thead>
<tr>
<th>Element</th>
<th>Rules</th>
</tr>
</thead>
</table>
| Overall control statement| - You must enter a command to start a control statement (a control statement cannot start with a keyword, and a control statement that starts with an asterisk is interpreted as a comment).  
  - You cannot enter data in columns 73 through 80. Data in these columns is ignored because a data set that contains syntax statements is assumed to be numbered. |
| Commands                 | - You can enter a command and its keywords on multiple lines (on multiple input records in the control statement data set).  
  - You must enter one or more spaces or a new line to separate a command from an expression (a keyword and its associated values) and to separate an expression from another expression. |
| Keywords                 | - You can enter the keywords (and their values) that are valid for a command in any order, but all keywords for a command must precede the next command in the control statement data set.  
  - You can enter a keyword and its values on the same line or on separate lines, and you can enter multiple values for a keyword on multiple lines.  
  - You must enter an equal sign (=) to separate a keyword from its values. You can enter any number of spaces (or no spaces at all) preceding and following an equal sign.  
  - You can enter a keyword in any column as long as the keyword does not extend past column 72. |
<table>
<thead>
<tr>
<th>Element</th>
<th>Rules</th>
</tr>
</thead>
</table>
| Values  | ■ You must use parentheses to enclose multiple values for a keyword. You can enter any number of spaces (or no spaces at all) preceding and following a parenthesis.  
■ You must enter one or more spaces, a comma, a comma and one or more spaces, or a new line to separate one value from another value for a keyword.  
■ You can enter a value in any column as long as the value does not extend past column 71 (unless you want to continue the value to the next line).  
■ You can enter a single value on multiple lines according to the following rules:  
— Enclose the value within single quote marks (').  
— Place the last nonblank character of the first line in column 72.  
— Place the first nonblank character of the following line in column 1. |
| Comments | ■ You can enter an asterisk (*) in column 1 to cause the syntax processor to ignore the line as a comment. |

**Control statement commands**

The following table lists the types of control statements that you can specify in a Log Analyzer control statement data set.

A control statement is identified by the command, which is the first nonblank element in the control statement. For more information about each command and its valid keywords, see the indicated topic.

**Table 6: Log Analyzer control statements**

<table>
<thead>
<tr>
<th>Control statement (abbreviation)</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANALYZE (AN)</td>
<td>This required statement specifies the logs to be read and defines general options for the Log Analyzer job. All keywords are optional unless you omit the INTERVAL statement.</td>
<td>“ANALYZE control statements” on page 63</td>
</tr>
</tbody>
</table>
### Control statement (abbreviation) | Description | Topic
--- | --- | ---
INTERVAL (INT) | This statement specifies the beginning and end of a range of records to process. If log data sets are allocated from RECON information, the INTERVAL statement is required. If you specify the SLDS keyword, the INPEXT keyword, or both keywords on the ANALYZE control statement, the INTERVAL statement is optional. | “INTERVAL control statement” on page 80
FILTER (FIL) | This optional statement specifies criteria for including and excluding log records during the Log Analyzer job. | “FILTER control statement” on page 83
REPORTS (REP) | This optional statement specifies batch reports to be produced. | “REPORTS control statement” on page 108
EXTRACT (EXTR) | This optional statement specifies parameters for allocating the extract data set. | “EXTRACT, INDEXFILE, and WORKFILE control statements” on page 129
INDEXFILE (INDEXF) | This optional statement specifies parameters for allocating the index data set. | “EXTRACT, INDEXFILE, and WORKFILE control statements” on page 129
WORKFILE (WORKF) | This optional statement specifies parameters for allocating the work data set. | “EXTRACT, INDEXFILE, and WORKFILE control statements” on page 129
END | This required control statement indicates the end of the commands in the control statement data set. | “Specifying END control statements” on page 135

### Minimum sets of control statements

The following examples show minimum sets of control statements that you can provide in a Log Analyzer control statement data set to produce LUOW-based reports.

Log Analyzer uses default values for all keywords and control statement that you do not provide. By default, all log records within selected sources of log records are included for analysis (no filter is applied). No extract file or index file is written, and interactive analysis through the ISPF interface is not possible. The only output that is produced is the Analysis Summary report.
**Example**

In the following example, a single SLDS is the source of log records to be analyzed:

```
ANALYZE SLDS=(logDataSetName,Rnn)
END
```

In the following example, a single extract data set is the source of log records to be analyzed:

```
ANALYZE INPEXT=extractDataSetName
END
```

When you want Log Analyzer to search the RECON for log data sets that have start times which are between the specified start and stop times:

```
ANALYZE INTERVAL START=yyyyddd/ STOP=yyyyddd/
END
```

You can specify a similar set of minimum control statements by identifying the source of log records and requesting a single-record report (DEADLOCK in this example) as the only output.

In the following example, a single SLDS is the source of log records to be analyzed, and the Deadlocks report is the only output:

```
ANALYZE SLDS=(logDataSetName,Rnn)
REPORTS DEADLOCK=ALL
END
```

---

**Specifying ANALYZE control statements**

Use the ANALYZE control statement (or the AN abbreviation) to specify the logs to be read and to define general options for the Log Analyzer job.

The control statement data set can contain one ANALYZE statement. This statement must be the first statement (other than comments) in the control statement data set.

**ANALYZE control statements**

The following example of an ANALYZE control statement contains all valid keywords.

For each keyword, if a default value is provided, this value is shown; if no default value is provided, a generic value is shown. Some keywords are shown as comments because they cannot be used at the same time as other keywords.

**Figure 18: ANALYZE control statement example**

```
ANALYZE
ACTIVEOLDS=NO
DB2LOG=db2LogDataSet | DB2BSDS=db2BootstrapDataSet
```

---

Chapter 3  Working with Log Analyzer jobs    63
More examples are located in “Allocating log record sources” on page 249.

The following table summarizes information about the keywords that you can specify on an ANALYZE control statement.

### Table 7: ANALYZE control statement keywords

<table>
<thead>
<tr>
<th>Keyword (abbreviation)</th>
<th>Valid values</th>
<th>Default</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTIVEOLDS (AOLDS)</td>
<td>NO, YES</td>
<td>NO</td>
<td>Specify whether Log Analyzer should select an active OLDS during RECON processing.</td>
<td>“ACTIVEOLDS keyword” on page 67</td>
</tr>
<tr>
<td>DB2BSDS</td>
<td>db2BstdsDsn</td>
<td>None</td>
<td>Identify a DB2 bootstrap data set to use as input. You can specify multiple instances of this keyword.</td>
<td>“DB2BSDS keyword” on page 68</td>
</tr>
<tr>
<td>DB2LOG</td>
<td>db2LogDsn</td>
<td>None</td>
<td>Identify a DB2 log data set to use as input. You can specify multiple instances of this keyword.</td>
<td>“DB2LOG keyword” on page 68</td>
</tr>
<tr>
<td>ELAPSEDTIME</td>
<td>YES, NO</td>
<td>YES</td>
<td>Specify whether Log Analyzer should report elapsed times for LUOWs.</td>
<td>“ELAPSEDTIME keyword” on page 69</td>
</tr>
</tbody>
</table>
### Keyword (abbreviation)  | Valid values  | Default  | Description  | Topic
---|---|---|---|---
**IMSID** | ALL  
*imsid*  
(*ims1, ims2, ...*) | ALL | Identify the IMS system (or systems) that created log records to be processed. Specify ALL to process records from all IMS systems. | “IMSID keyword” on page 70
**INPEXT**  
**INPE** | *(extractDsn)* | None | Identify a previously created Log Analyzer extract data set to use as input. You can specify multiple instances of this keyword. | “INPEXT keyword” on page 70
**IPRJRNL** | *(journalDsn)* | None | Identify an Energizer for IMS Connect journal to use as input. You can specify multiple instances of this keyword. | “IPRJRNL keyword” on page 71
**IPRPREFIX** | *(journalPrefix)* | None | Include Energizer for IMS Connect journals automatically by identifying the journal prefix. You can specify multiple instances of this keyword. | “IPRPREFIX keyword” on page 72
**LIMIT**  
**LIM** | *nnnnnnnnnn*  
*(No limit)* | 9999999999 | Specify a maximum number of log records to process (after skipping log records, if SKIP is specified). | “LIMIT keyword” on page 73
**LOGRECORDCODEPDS**  
**LRCPDS** | *dataSetName* | None | Specify the name of a data set that contains predefined lists of log record codes to be filtered. | “LOGRECORDCO EPDS keyword” on page 73
<table>
<thead>
<tr>
<th>Keyword (abbreviation)</th>
<th>Valid values</th>
<th>Default</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAXLOGS (MAXL)</td>
<td>$\text{nnn}$&lt;br&gt;($\text{nnn, abendCode}$)&lt;br&gt;($\text{nnn,RreturnCode}$)&lt;br&gt;$\text{999}$</td>
<td>100</td>
<td>Limit the number of input log data sets. Or issue the specified user abend code or return code if Log Analyzer determines that the number of input log data sets is greater than the specified number. Specify 999 to set no limit on the number of input log data sets.</td>
<td>“MAXLOGS keyword” on page 74</td>
</tr>
<tr>
<td>MQEXT</td>
<td>$\text{mqExtractDataSet}$</td>
<td>None</td>
<td>Identify a WebSphere for MQ extract data set to use as input. You can specify multiple instances of this keyword.</td>
<td>“MQEXT keyword” on page 75</td>
</tr>
<tr>
<td>RECON</td>
<td>($\text{reconDsn1}$, $\text{reconDsn2}$, $\text{reconDsn3}$)</td>
<td>None</td>
<td>Explicitly identify a set of RECON data sets to examine for log record sources. You can specify multiple instances of this keyword.</td>
<td>“RECON keyword” on page 76</td>
</tr>
<tr>
<td>SKIP</td>
<td>$\text{nnnnnnnnnn}$</td>
<td>0</td>
<td>Specify a number of log records to skip before beginning to process log records.</td>
<td>“SKIP keyword” on page 76</td>
</tr>
<tr>
<td>SLDS</td>
<td>($\text{sldsName}$, $\text{R vr}$, $\text{unit}$, $\text{volser}$)</td>
<td>None</td>
<td>Explicitly identify a log data set to read and the version of IMS that created the log data set. If the data set is not cataloged, specify the unit and the name of the volume that contains the data set. You can specify multiple instances of this keyword.</td>
<td>“SLDS keyword” on page 77</td>
</tr>
</tbody>
</table>
### ACTIVEOLDS keyword

You can use the ACTIVEOLDS keyword (or the AOLDS abbreviation) on the ANALYZE control statement to specify whether Log Analyzer is allowed to select an active OLDS during RECON processing.

Log Analyzer automatically can identify log data sets to use as input (as explained in “Implicitly specifying IMS log data sets to allocate” on page 252). To prevent IMS performance problems, by default Log Analyzer does not try to select an OLDS that is in active use by IMS. If you want to allow Log Analyzer to select an active OLDS, you can specify the data set name of the active OLDS by using the SLDS keyword or you can specify the ACTIVEOLDS keyword.

Specify one of the following values:

- **NO**
  
  Do not select an OLDS that is designated as an active OLDS in the RECON. This value is ignored if you have specified the data set name of the OLDS by using the SLDS keyword.

- **YES**
  
  Allow Log Analyzer to select an OLDS for input to the analysis process even if it is designated as an active OLDS in the RECON. (Log Analyzer selects an active OLDS only if it is included in the specified interval.)

The default value is NO.

**Example**

To allow Log Analyzer to select an active OLDS for the IMSA system, specify the following keywords on the ANALYZE control statement:

```
IMSID=IMSA
ACTIVEOLDS=YES
```
For more information about the ACTIVEOLDS keyword and examples of its use, see “Using an active OLDS through implicit specification” on page 256.

**DB2BSDS keyword**

You can use the DB2BSDS keyword on the ANALYZE control statement to identify a DB2 bootstrap data set (BSDS).

Log Analyzer uses this data set to obtain the names of DB2 log data sets to use as input to the analysis job, based on the selected logical units of work (LUOWs) that Log Analyzer has created from input IMS records.

If you want Log Analyzer to use multiple DB2 BSDSs, specify multiple instances of the DB2BSDS keyword.

Specify the fully qualified data set name. This name should refer to a VSAM key-sequenced data set (KSDS). Do not specify the name of a BSDS archive copy, which is not a VSAM KSDS.

No default value is provided. If you omit the DB2BSDS keyword, Log Analyzer does not read any DB2 BSDSs. If you specify the DB2BSDS keyword, do not specify the DB2LOG keyword.

**Example**

You want to include two DB2 BSDSs in the Log Analyzer analysis job. Specify the following keywords on the ANALYZE control statement:

```plaintext
DB2BSDS=DB2P.BSDSA
DB2BSDS=DB2P.BSDSB
```

For more information about the DB2BSDS keyword, see “Using DB2 logs as input” on page 262.

**DB2LOG keyword**

You can use the DB2LOG keyword on the ANALYZE control statement to identify a DB2 log data set that was created by a DB2 subsystem.

Log Analyzer uses this data set as input to the analysis job. Log Analyzer can associate certain information about DB2-related events with LUOWs. If you want Log Analyzer to read multiple DB2 log data sets, specify multiple instances of the DB2LOG keyword.
Specify the fully qualified data set name. No default value is provided. If you omit the DB2LOG keyword, Log Analyzer does not read any DB2 logs. If you specify the DB2LOG keyword, do not specify the DB2BSDS keyword.

**Example**

You want to include two DB2 logs in the Log Analyzer analysis job. Specify the following keywords on the ANALYZE control statement:

```plaintext
DB2LOG=DB2P.PRODLOG.D2009237.T1251037
DB2LOG=DB2P.PRODLOG.D2009237.T1254421
```

For more information about the DB2LOG keyword, see “Using DB2 logs as input” on page 262.

**ELAPSEDTIME keyword**

You can use the ELAPSEDTIME keyword on the ANALYZE control statement to control whether Log Analyzer reports elapsed time values for LUOWs.

Log Analyzer calculates the elapsed time for an LUOW by subtracting the timestamp of the first record that is included in the LUOW from the timestamp of the last record that is included in the LUOW. Log Analyzer can report the calculated elapsed time on the Analysis Summary report and on the LUOW Summary report. You can use elapsed time as a filter on FILTER SELECT statements.

**WARNING**

For Log Analyzer to report valid values for calculated elapsed times, all log records that are included in the analysis must have been produced on systems that are using the same sysplex timer. If the systems are using different sysplex timers, specify ELAPSEDTIME=NO to prevent Log Analyzer from reporting misleading values.

You can specify one of the following values:

- **YES**
  
  Report elapsed time values.

- **NO**
  
  Do not report elapsed time values. Specify this value if log records have been produced on systems that are not using the same sysplex timer.

The default is ELAPSEDTIME=YES.


**Example**

You know that your analysis job includes logs from systems which use different sysplex timers. To prevent Log Analyzer from reporting elapsed time values, specify the following keyword on the ANALYZE control statement:

```plaintext
ELAPSEDTIME=NO
```

---

**IMSID keyword**

You can use the IMSID keyword on the ANALYZE control statement to specify one or more IMSIDs.

Each IMSID (one to eight characters) identifies an IMS system that created log records to be analyzed. Log Analyzer uses the IMSID when determining which log data sets to use as input, as explained in “Implicitly specifying IMS log data sets to allocate” on page 252.

To process log records for all IMS systems that are associated with the input RECON data set (or data sets), specify IMSID=ALL.

The default is IMSID=ALL.

If you specify the SLDS keyword, you cannot specify the IMSID keyword. The IMSID keyword applies only if Log Analyzer is reading RECON data sets to determine which log data sets to process.

**Example**

A set of RECON data sets is shared by three IMS systems: one that has been defined with IMSID IMSA, one with IMSB, and one with IMSC. To process only the log records that were created by the IMSA system or by the IMSB system, specify the following keyword on the ANALYZE control statement:

```plaintext
IMSID=(IMSA, IMSB)
```

For more information about the IMSID keyword, see “Specifying IMSIDs” on page 255.

---

**INPEXT keyword**

You can use the INPEXT keyword (or the INPE abbreviation) on the ANALYZE control statement to identify an input extract data set (which contains extracted log records) that was created during a previous execution of Log Analyzer.
If you want Log Analyzer to read multiple extract data sets, specify multiple instances of the INPEXT keyword.

**Note**
If you specify the INPEXT keyword, you must also specify the WORKFILE control statement to define a sort work file. Log Analyzer uses this work file to sort the extracted log records. For more information, see “Specifying EXTRACT, INDEXFILE, and WORKFILE control statements” on page 128.

No default value is provided for the INPEXT keyword. If you omit the INPEXT keyword, Log Analyzer does not read any extract data sets.

**Example**
Log Analyzer created an extract data set at each of your remote sites. These extract data set were sent to your central site for further analysis. To read these data sets during a subsequent Log Analyzer job, specify the following keywords on the ANALYZE control statement:

INPEXT=(BMCLUI.DENVER.D2007102.EXTRACT)
INPEXT=(BMCLUI.LOSANGEL.D2007102.EXTRACT)
INPEXT=(BMCLUI.PHOENIX.D2007102.EXTRACT)
INPEXT=(BMCLUI.SEATTLE.D2007102.EXTRACT)

For more information about the INPEXT keyword, see “Using extract files as input” on page 257.

**IPRJRNL keyword**

You can use the IPRJRNL keyword on the ANALYZE control statement to identify journal files that the BMC Energizer for IMS Connect product created.

Log Analyzer will use these files as input to the analysis job. The journal files contain information about IMS Connect events such as Read Socket, Send to IMS, Send to Client, and IMS Send to IMS Connect.

If you want Log Analyzer to read multiple Energizer journal data sets, specify multiple instances of the IPRJRNL keyword. If you specify the IPRJRNL keyword, do not specify the IPRPREFIX keyword.

No default value is provided. If you omit the IPRJRNL keyword (and you do not specify the IPRPREFIX keyword), Log Analyzer does not read any Energizer for IMS Connect journal files.
Example

Energizer created a set of journal files that you want to include in the Log Analyzer analysis job. Specify the following keywords on the ANALYZE control statement:

IPRJRNL=(BMCIPR.IPAHWS8.JOURNAL3.JAN2209)
IPRJRNL=(BMCIPR.IPAHWS8.JOURNAL4.JAN2209)
IPRJRNL=(BMCIPR.IPAHWS8.JOURNAL5.JAN2209)

For more information about the IPRJRNL keyword, see “Using Energizer journals as input” on page 260.

IPRPREFIX keyword

You can use the IPRPREFIX keyword (or the IPRP abbreviation) on the ANALYZE control statement to automatically include appropriate journal files that the BMC Energizer for IMS Connect product created.

The journal files contain information about IMS Connect events such as Read Socket, Send to IMS, Send to Client, and IMS Send to IMS Connect.

The data set name of an Energizer journal begins with a site-specified prefix and contains a timestamp, which indicates the Universal Time Code (UTC) creation time. If you use the IPRPREFIX keyword to provide the journal prefix, Log Analyzer can search the system catalog for all data sets that match the journal prefix and that have timestamps within the analysis time range. Log Analyzer determines the time range automatically from the IMS records that are included in the analysis.

For the value of the IPRPREFIX keyword, specify the Energizer journal prefix that matches the data set name prefix that was specified in the Energizer options. If you want Log Analyzer to read journals from multiple IMS Connect systems, specify multiple instances of the IPRPREFIX keyword. (To avoid unpredictable results, specify a unique prefix for each of these systems.) If you specify the IPRPREFIX keyword, do not specify the IPRJRNL keyword.

No default value is provided. If you omit the IPRPREFIX and IPRJRNL keywords, Log Analyzer does not read any Energizer for IMS Connect journal files.

Example

You want to include all relevant Energizer journals from two IMS Connect systems in the Log Analyzer analysis job. Specify the following keywords on the ANALYZE control statement:

IPRPREFIX=(IMS1.IPRJRNL1)
IPRPREFIX=(IMS2.IPRJRNL2)
For more information about the IPRPREFIX keyword, see “Using Energizer journals as input” on page 260.

LIMIT keyword

You can use the LIMIT keyword (or the LIM abbreviation) on the ANALYZE control statement to set a limit on the number (1 through 999999999) of log records to process from an input log data set.

The count of records to be limited begins after Log Analyzer has skipped the number of records that was specified with the SKIP keyword. After Log Analyzer processes the specified number of records, the process terminates as though it has reached the end of the data set. Specify LIMIT=999999999 to process an unlimited number of records.

The default is LIMIT=999999999.

If you specify the LIMIT keyword, you must specify one SLDS keyword. The SLDS keyword is required, and multiple SLDS keywords are not allowed.

**Example**

You are analyzing the log records in a single log data set named IMSP.LOG00001. If you do not want to process more than 10,000 records from the log data set, specify the following keywords on the ANALYZE control statement:

SLDS=(IMSP.LOG00001,R91)
LIMIT=10000

For more information about the LIMIT keyword, see “Skipping and limiting log records” on page 270.

LOGRECORDCODEPDS keyword

You can use the LOGRECORDCODEPDS keyword (or the LRCPDS abbreviation) on the ANALYZE control statement to specify the name of a PDS.

Each member of this PDS contains a predefined list of log record codes. You can then use the name of a member as a value of the LOGRECORDCODES keyword on the FILTER control statement, as explained in “LOGRECORDCODES keyword on the FILTER control statement” on page 85.

No default is provided. If you omit this keyword, you cannot specify a list of log record codes as a value of the LOGRECORDCODES keyword on the FILTER control statement.
If the name of your log record code PDS is BMCLUI.LRCPDS, specify the following keyword on the ANALYZE control statement:

```
LOGRECORDCODEPDS=BMCLUI.LRCPDS
```

For more information about creating the log record codes PDS and defining lists of log record codes, see “Using a log record code PDS” on page 275.

## MAXLOGS keyword

You can use the MAXLOGS keyword (or the MAXL abbreviation) on the ANALYZE control statement to control the number of log data sets that Log Analyzer processes.

Use this keyword when you are using implicit specification of log data sets and Log Analyzer is reading RECON data sets to determine which log data sets to process. In this case, the number of log data sets to be processed is unknown. The MAXLOGS keyword can help you prevent a problem that could result from processing an excessive number of log data sets.

The MAXLOGS keyword works differently, depending on how you specify the value:

- If you specify a value from 1 through 999, Log Analyzer stops reading log data sets after it reaches the specified number of log data sets. If Log Analyzer does not read other log data sets because the specified limit was reached, a warning message indicates the number of log data sets that were not read.

- If you specify a value from 1 through 999 followed by an abend code (1 through 4095), Log Analyzer issues a user abend code and terminates the job step if more than the specified number of log data sets would be read. Log Analyzer issues the user abend at the conclusion of the RECON analysis phase without reading any input log data sets.

- If you specify a value from 1 through 999 followed by literal R and a return code (08 through 99), Log Analyzer issues the specified return code and terminates the job step if more than the specified number of log data sets would be read. Log Analyzer issues the return code at the conclusion of the RECON analysis phase without reading any input log data sets.

If you want Log Analyzer to read all log data sets, specify MAXLOGS=999.

The default is MAXLOGS=100 (Log Analyzer reads a maximum of 100 log data sets).
Example
If you want Log Analyzer to read no more than 20 log data sets, specify the following keyword on the ANALYZE control statement:
MAXLOGS=20

Example
If you want Log Analyzer to issue a user abend, with abend code U4000, if more than 120 log data sets would be read, specify the following keyword on the ANALYZE control statement:
MAXLOGS=(120,4000)

Example
If you want Log Analyzer to issue return code 16 and terminate the job step if more than 10 log data sets would be read, specify the following keyword on the ANALYZE control statement:
MAXLOGS=(10,R16)

For more information about the MAXLOGS keyword, see “Specifying a maximum number of logs to read” on page 255.

MQEXT keyword

You can use the MQEXT keyword on the ANALYZE control statement to identify a WebSphere for MQ extract file.

Log Analyzer uses this file as input to the analysis job. The value of MQEXT is the data set name of the extract file. If you want to include multiple extract files, specify one MQEXT keyword for each extract file, and specify the extract files in chronological order.

No default value is provided. If you omit the MQEXT keyword, Log Analyzer does not read any WebSphere for MQ extract files.

Example
You have created two extract files to include in the analysis.
MQTEST.EXTRACT.T0000 contains records with the earliest time stamps.
MQTEST.EXTRACT.T0800 contains records with the latest time stamps. In this example, you have provided the required IMS log input by specifying the SLDS keyword.

```
ANALYZE ...
SLDS=IMSTEST.SLDS.T0000
MQEXT=MQTEST.EXTRACT.T0000
MQEXT=MQTEST.EXTRACT.T0800
...
```
For more information about the MQEXT keyword, see “Using WebSphere for MQ extract data sets” on page 265.

**RECON keyword**

You can use the RECON keyword on the ANALYZE control statement to specify the names of a set of RECON data sets to examine during the Log Analyzer job.

If you want Log Analyzer to examine multiple sets of RECON data sets, you must specify each set with a separate RECON keyword.

If you specify the RECON keyword, your user ID must have Security Access Facility (SAF) READ access to the RECON data sets.

You can specify from one to three data set names. One of the specified data set names must be designated as COPY1 by IMS.

No default value is provided for the RECON keyword. If you omit the RECON keyword, the SLDS keyword, and the INPEXT keyword, Log Analyzer allocates one set of RECON data sets by using IMS dynamic allocation (MDA) members, as explained in “Implicitly specifying IMS log data sets to allocate” on page 252.

If you specify the RECON keyword, you must specify the INTERVAL control statement with a START time value and a STOP time value.

**Example**

The RECON data sets for the IMSA system are named IMSA.RECON1 and IMSA.RECON2, and the RECON data sets for the IMSB system are named IMSB.RECON1 and IMSB.RECON2. To examine IMSA and IMSB RECON data sets in the Log Analyzer job, specify the following keywords on the ANALYZE control statement:

```plaintext
RECON=(IMSA.RECON1, IMSA.RECON2)
RECON=(IMSB.RECON1, IMSB.RECON2)
```

For more information about the RECON keyword, see “Specifying RECON data set names” on page 253.

**SKIP keyword**

You can use the SKIP keyword on the ANALYZE control statement to specify a number (1 through 999999999) of log records in an input log data set to skip before Log Analyzer begins to process records. The default is SKIP=0, which skips no records.
If you specify the SKIP keyword, you must specify one SLDS keyword. The SLDS keyword is required, and multiple SLDS keywords are not allowed.

**Example**

You are analyzing the log records in a single log data set named IMSP.LOG00001. To skip processing of the first 9,000 records in the log data set, specify the following keywords on the ANALYZE control statement:

```
SLDS=(IMSP.LOG00001,R91)
SKIP=9000
```

For more information about the SKIP keyword, see “Skipping and limiting log records” on page 270.

**SLDS keyword**

You can use the SLDS keyword on the ANALYZE control statement to identify explicitly an input log data set to use as input to the analysis process.

Specify the keyword values as follows:

```
SLDS=(dataSetName,Rvr, unit, volser)
```

- For the *dataSetName* value, specify the fully qualified data set name. The data set can be an SLDS, an OLDS, or a log extract file that was produced by the IMS File Select and Formatting Print utility (program DFSERA10), the IEBGENER utility, or a similar utility which produces a file that contains nothing except IMS log records in their original format.

- For the *Rvr* value, specify the literal R followed by the IMS version and release number of the IMS system that created the data set (for example, R91 for IMS version 9.1 and R101 for IMS version 10.1).

  **WARNING**

  Ensure that the correct release of IMS is specified. The internal layout of IMS log records depends on the version of IMS. If the wrong release is specified, Log Analyzer might not be able to detect the error, and results are unpredictable.

- If the log data set is not cataloged, specify the *unit* value to identify the unit and the *volser* value to identify the volume serial (VOLSER) of the volume that contains the log data set. Omit the *unit* value and the *volser* value to obtain volume information from the RECONs or the system catalog.

If you want Log Analyzer to read multiple log data sets, specify multiple instances of the SLDS keyword.
No default values are provided. If you omit the SLDS keyword and the INPEXT keyword, Log Analyzer automatically identifies log data sets to use as input, as explained in “Implicitly specifying IMS log data sets to allocate” on page 252.

If you specify the SLDS keyword, you cannot specify the IMSID keyword. If you specify more than one SLDS keyword, you cannot specify the LIMIT keyword or the SKIP keyword, and you cannot specify a log sequence number in the INTERVAL control statement.

**Example**

IMSA is an IMS version 10.1 system and creates generations of SLDSs that use the naming convention imsid.SLDS.Ddate.GENnnn. The SLDSs are cataloged. To analyze log records that IMSA wrote to generations 10 through 12 on March 29, 2008, specify the following keywords on the ANALYZE control statement:

SLDS=(IMSA.SLDS.D2008088.GEN010, R101)
SLDS=(IMSA.SLDS.D2008088.GEN011, R101)
SLDS=(IMSA.SLDS.D2008088.GEN012, R101)

For more information about the SLDS keyword, see “Specifying SLDS and OLDS data set names” on page 251.

**TIMEZONE keyword**

You can use the TIMEZONE keyword (or the TIMEZ abbreviation) on the ANALYZE control statement to specify how time values are interpreted when you are specifying time input values and when Log Analyzer is displaying time values in reports and on panels.

You can specify one of the following values:

- **LOCAL SUBPARAGRAPH**
  Use the offset that is specified in the z/OS system that is executing the Log Analyzer job. Using this offset, Log Analyzer automatically converts time values to the equivalent coordinated universal time (UTC), also referred to as Greenwich Mean Time (GMT).

- **GMT**
  Use UTC times. Log Analyzer interprets and displays all time values as UTC times.
Note
Using TIMEZONE=GMT is the safest option because IMS always writes log records with GMT values. With GMT, no confusion can result from time shifts (standard time to daylight savings time, or daylight savings time to standard time) or analysis of records from multiple time zones.

Even if you usually specify TIMEZONE=LOCAL, if you are analyzing logs across a time shift you might want to specify TIMEZONE=GMT. With this specification, Log Analyzer automatically handles the time shift for you.

- \((-hh:mm\) or \(+hh:mm\))
  Use the specified offset instead of the offset that is specified in the z/OS system that is executing the Log Analyzer job. Using this offset, Log Analyzer interprets and displays time values as GMT plus or minus the specified number of hours and minutes. If you want to work in local time but Log Analyzer is processing log records that were created in a different time zone than your own, you might find it easiest to specify TIMEZONE=\((-hh:mm\) or \(+hh:mm\).

Note
You must use 15-minute increments for the \(mm\) units. If you use any other increment, you will encounter an error message.

- ORIGINAL
  Automatically determine and display the local time offset from GMT by using the UTC or IMS timestamps in log records within the SLDS.

  You cannot specify TIMEZONE=ORIGINAL if you specify START and STOP timestamps on the INTERVAL control statement.

  The default is TIMEZONE=LOCAL.

Example
To work with log records in a local time zone that is 6 hours earlier than GMT, specify the following keyword on the ANALYZE control statement:
TIMEZONE=\((-0600)\)

Specifying INTERVAL control statements

The INTERVAL control statement (or the INT abbreviation) specifies the beginning and end of a range of log records to process.

Log Analyzer also uses this statement to determine which log data sets to allocate automatically from RECON information.
If log data sets are allocated from RECON information, the INTERVAL statement is required. If you specify the SLDS keyword, the INPEXT keyword, or both keywords on the ANALYZE control statement, the INTERVAL statement is optional. For more information, see “Implicitly specifying IMS log data sets to allocate” on page 252.

To process multiple noncontiguous ranges of log records, specify multiple INTERVAL statements (each with a START and STOP keyword). An INTERVAL statement must follow the ANALYZE statement and precede the END control statement, but you can specify it in any position relative to other control statements.

**INTERVAL control statement**

The following example of an INTERVAL control statement contains all valid keywords. For each keyword, the default value is shown.

**Figure 19: INTERVAL control statement example**

```
INTERVAL
  START=FIRST
  STOP=LAST
```

More examples are located in “Specifying intervals” on page 277.

The following table summarizes information about the keywords that you can specify on an INTERVAL control statement.

**Table 8: INTERVAL control statement keywords**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Valid values</th>
<th>Default</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>START</td>
<td>FIRST <code>yyyyddd/ hhmmsssth logSequenceNumber</code></td>
<td>FIRST</td>
<td>Specify the beginning of a range of log records to process.</td>
<td>“START keyword” on page 80</td>
</tr>
<tr>
<td>STOP</td>
<td>LAST <code>yyyyddd/ hhmmsssth logSequenceNumber</code></td>
<td>LAST</td>
<td>Specify the end of a range of log records to process.</td>
<td>“STOP keyword” on page 82</td>
</tr>
</tbody>
</table>

**START keyword**

You can use the START keyword on the INTERVAL control statement for the following purposes:

- Specify the beginning a range of log records to identify as candidates for the record selection process.
Control the beginning of range of log data sets to allocate from RECON information

Reduce the overall number of log records that are selected from input SLDS or extract files

Specify one of the following values:

- **FIRST**
  To start processing a range of records at the first record in the input log data set, specify START=FIRST. This value is valid only if you have specified one or more SLDS keywords.

- **yyyyddd/hhmmssth**
  To start processing a range of records that were created on or after a specified date and time, specify the date and time in the format yyyyddd/hhmmssth (yyyy is the year, ddd is the Julian day of the year, hh is the hour, mm is the minute, ss is the second, t is the tenth of a second, and h is the hundredth of a second). To use the default value of 0 for the time, omit any trailing portion of the time value (but retain the literal /).
  
  This time value should match the type that the TIMEZONE keyword on the ANALYZE control statement specifies. For example, if you specify TIMEZONE=LOCAL, specify a local time value for the START keyword.

- **logSequenceNumber**
  To start processing a range of records at a specified record, specify the log sequence number (in hexadecimal format). This value is valid only if you have specified one SLDS. The SLDS keyword is required, and you cannot specify multiple SLDS keywords.

If you have specified the SLDS keyword, the default is START=FIRST. Otherwise, no default value is provided.

**Example**
To process a range of log records that were created at any time on or after April 12, 2007, specify the following keyword on the INTERVAL control statement:

```
START=2007102/
```

For more information about the START keyword, see “Specifying intervals” on page 277.
STOP keyword

You can use the STOP keyword on the INTERVAL control statement for the following purposes:

- Specify the end of a range of log records to identify as candidates for the record selection process
- Control the end of range of log data sets to allocate from RECON information
- Reduce the overall number of log records that are selected from input SLDS or extract files

Specify one of the following values:

- **LAST**
  
  To end processing a range of records at the last record in the input log data set, specify STOP=LAST. This value is valid only if you have specified one or more SLDS keywords.

- **yyyyddd/hhmmssth**
  
  To end processing a range of records that were created on or before a specified date and time, specify the date and time in the format `yyyyddd/hhmmssth` (where `yyyy` is the year, `ddd` is the Julian day of the year, `hh` is the hour, `mm` is the minute, `ss` is the second, `t` is the tenth of a second, and `h` is the hundredth of a second). To use the default value of 0 for the time, omit any trailing portion of the time value (but retain the literal `/`).

  This time value should match the type that the TIMEZONE keyword on the ANALYZE control statement specifies. For example, if you specify TIMEZONE=LOCAL, specify a local time value for the STOP keyword.

- **logSequenceNumber**
  
  To end processing a range of records at a specified record, specify the log sequence number (in hexadecimal format). This value is valid only if you have specified one SLDS. The SLDS keyword is required, and you cannot specify multiple SLDS keywords.

If you have specified the SLDS keyword, the default is STOP=LAST. Otherwise, no default value is provided.

---

**Example**

To process a range of log records that were created at any time before April 13, 2007, specify the following keyword on the INTERVAL control statement:

STOP=2007102/23595999
For more information about the STOP keyword, see “Specifying intervals” on page 277.

Specifying FILTER control statements

You can use the FILTER control statement (or the FIL abbreviation) to include and exclude LUOWs from consideration by Log Analyzer, according to the criteria that you specify.

If you omit the FILTER statement, Log Analyzer considers all LUOWs from all input sources of log records (except for records that are not processed because of SKIP, LIMIT, START, or STOP keyword values on the ANALYZE control statement).

You can specify only one FILTER statement in each command set. The FILTER statement must follow the ANALYZE statement and precede the END control statement, but you can specify it in any position relative to other control statements.

Note
If you request a specialty report by specifying the DEADLOCK keyword or the APPCHECK keyword on the REPORTS control statement, you cannot specify the FILTER control statement. Log Analyzer produces specialty reports by using filtering techniques that are different from the techniques for other reports.

FILTER control statement

The following example of a FILTER control statement contains all valid keywords. For each keyword, if a default value is provided, this value is shown; if no default value is provided, a generic value is shown.

Figure 20: FILTER control statement example

```
FILTER
  CONTENT0103=DATA
  LOGRECORDCODES=value
  SELECT=((subkeyword=value) AND (subkeyword operand> value))
  OR ((subkeyword <operand> value))
```

More examples are located in “Selecting log records and LUOWs” on page 283.

The following table summarizes information about the keywords that you can specify on a FILTER control statement.
Table 9: FILTER control statement keywords

<table>
<thead>
<tr>
<th>Keyword (abbreviation)</th>
<th>Valid values</th>
<th>Default</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTENT0103 (CONT)</td>
<td>DATA ALL</td>
<td>DATA</td>
<td>Control how Log Analyzer handles content filtering for type 01 log records and type 03 log records.</td>
<td>“CONTENT0103 keyword” on page 84</td>
</tr>
<tr>
<td>LOGRECORDCODES (LRC)</td>
<td>ALL, nn or n*, nnnn or nnn*, ¬nn or ¬n*, ¬nnnn or ¬nnn*, memberName (nn, n*, ¬nn, ¬n*, nnnn, nnn*, ¬nnnn, ¬nnn*, memberName, ...)</td>
<td>Default value depends on other keywords</td>
<td>Specify the types and subtypes of log records that are eligible to be included in LUOWs.</td>
<td>“LOGRECORDCODES keyword on the FILTER control statement” on page 85</td>
</tr>
<tr>
<td>SELECT (SEL)</td>
<td>subkeywords AND OR operands ( )</td>
<td>None</td>
<td>Specify which LUOWs to include or exclude for analysis.</td>
<td>“SELECT keyword” on page 89</td>
</tr>
</tbody>
</table>

**CONTENT0103 keyword**

You can use the CONTENT0103 keyword (or the CONT abbreviation) on the FILTER control statement to control how Log Analyzer handles content filtering for type 01 (IMS input message) log records and type 03 (IMS output message) log records.

Specify one of the following values:

- **DATA**
  
  This value causes Log Analyzer to check for the filtering-target content in only the user data portion of 01 and 03 log records.
ALL

This value causes Log Analyzer to check for the filtering-target content in all portions of 01 and 03 log records. (Log Analyzer always checks the entire log record for types other than 01 and 03.)

Scratch-pad area (SPA) segments are always excluded from a content search, regardless of the value of the CONTENT0103 keyword.

The default is CONTENT0103=DATA.

Example

You want to search all portions of all eligible log records, including 01 and 03 records, for the string 012345. Specify the following keywords on the FILTER control statement:

CONTENT0103=ALL
SELECT=CONTENT=(C'012345',**)

For more information about the CONTENT0103 keyword, see “Searching for content in 01 and 03 records” on page 287.

LOGRECORDCODES keyword on the FILTER control statement

You can use the LOGRECORDCODES keyword (or the LRC abbreviation) on the FILTER control statement to specify the types and subtypes of log records that are eligible to be included in LUOWs and in Log Analyzer output files.

The LOGRECORDCODES keyword is optional. If you omit it and you specify the SELECT keyword on the FILTER control statement, Log Analyzer automatically determines the types of log records to include.

Tip

If you specify the SELECT keyword, in most cases you should omit the LOGRECORDCODES keyword and allow Log Analyzer to determine the types of log records to include automatically. Use the LOGRECORDCODES keyword when you want to include other record types in addition to the automatically selected types.

WARNING

If you specify a value other than ALL for the LOGRECORDCODES keyword and you also specify the SELECT keyword, Log Analyzer uses the specified set of log record codes. If the specified set does not contain all of the log record types that Log Analyzer would have included automatically, Log Analyzer issues a message and sets return code 4 to indicate that LUOWs might be missing or incomplete.
If you omit the LOGRECORDCODES keyword and you do not specify the SELECT keyword, the default is LOGRECORDCODES=ALL (all log records are eligible).

**Note**
Under certain conditions, Log Analyzer adds log record codes to the explicit list of log record codes that you have specified with the LOGRECORDCODES keyword. If you do not want to view information about these added types of records in Log Analyzer reports, you can specify the LOGRECORDCODES keyword on the REPORTS control statement to exclude these records.

For more information, see “Identifying log record codes” on page 271.

**Specifying explicit log record code values**

If you specify the LOGRECORDCODES keyword, use one or more of the following values to specify log record codes to include or exclude:

- Two-character log record type code (*nn*) or four-character log record type and subtype code (*nnnn*)
  
  Log Analyzer recognizes all standard log record types that are defined by IMS and most log record types that are defined by BMC. If you are interested in only a specified subtype of a log record type, you can specify the subtype code. If you omit the subtype code, the specification applies to all subtypes of the specified log record type. For information about supported log record types, see “IMS log record types” on page 345.

- Masked two-character log record type code (*n*) or masked four-character log record type and subtype code (*nnn*)
  
  You can use an asterisk (*) as a mask for the second character position of a log record code or for the fourth character position of a log record type and subtype code. Log Analyzer selects all log record codes that have the specified character in the masked position. For example, specify LOGRECORDCODES=3* to select record types 30, 31, 32, 33, and so on.

- ALL
  
  You can specify LOGRECORDCODES=ALL to include all log record codes. This value is useful if you want to include all log record codes except the ones that you have specifically excluded or if you are running an analysis job to obtain high-level summary information.
Note

Log Analyzer overrides the LOGRECORDCODES=ALL specification if all of the following conditions are true:

- You specify the LOGRECORDCODES=ALL keyword with no included or excluded log record codes.
- You specify the SELECT keyword on the FILTER control statement.
- You request one or more reports other than the Analysis Summary report (by specifying a keyword other than the SUMMARY keyword on the REPORTS control statement).

In this case, Log Analyzer uses a default list of log record codes to match the SELECT keyword values. A warning message and return code 4 are issued.

- Name of a PDS member that contains a predefined list of log record codes (memberName)

Lists of log record codes are defined in members of the Log Analyzer log record code PDS. You specify the name of the PDS to Log Analyzer by using the LOGRECORDCODEPDS keyword on the ANALYZE control statement, as explained in “LOGRECORDCODEPDS keyword” on page 73. For more information about creating the log record codes PDS and defining lists of log record codes, see “Using a log record code PDS” on page 275.

Although some log record codes would otherwise be valid member names (A0 through FF), Log Analyzer always interprets a two-character value within the range of 00 through FF as a log record code.

Including log record codes

To include a specified log record code, specify the value without the logical-not symbol (~).

Log Analyzer can include log record codes that you did not specifically include. For more information, see “Automatically included log record types” on page 274.

Example

You want to analyze LUOWs that are related to input messages, output messages, and queue management, and you also want to include all log records that are created by BMC products. You have defined a list of log record codes to include BMC records. This list is located in member BMCRECS in a log record codes PDS named BMCLUI.LRCPDS. Specify the following control statements and keywords:

ANALYZE LOGRECORDCODEPDS=BMCLUI.LRCPDS
FILTER LRC=(01, 03, 3*, BMCRECS)
Excluding log record codes

To exclude a specified log record code, precede the value with the logical-not symbol (~).

The logical-not symbol is assigned to EBCDIC code X'5F' and is typed by pressing **Shift - 6** on most keyboards. **Shift - 6** is labeled as a circumflex (^) on some keyboards. You can also use the logical-not symbol in defined lists of log record codes.

To include all log record codes except specified codes, specify the value ALL and then specify each log record code to exclude by preceding the log record code (or masked code) with the logical-not symbol.

If an identical log record type (and subtype) is excluded and included, exclusion overrides inclusion. (This specification is valid and might occur, for example, if you are using multiple LOGRECORDCODESPDS members.) For example, if you specify LOGRECORDCODES=(67,~67), type 67 records are excluded.

If you specify a masked value and a specific value, inclusion or exclusion of the specific value overrides the masked value. For example, if you specify LOGRECORDCODES=(~6*,67), all type 6 x records are excluded except type 67 records.

Inclusion or exclusion of a log record subtype overrides inclusion or exclusion of the corresponding log record type. For example, if you specify LOGRECORDCODES=(67,~6701), all type 67 records are included except type 6701 records. If you specify LOGRECORDCODES=(~67,6701), all type 67 records are excluded except type 6701 records.

You can use the logical-not symbol to exclude some log record types that Log Analyzer has included automatically because of specified filter criteria. Some log record types that Log Analyzer has included automatically cannot be excluded from the log record selection process. (For more information, see “Automatically included log record types” on page 274.) You can exclude these automatically selected types from Log Analyzer reports by specifying the LOGRECORDCODES keyword on the REPORTS control statement.
Example
You want to analyze all log records except for padding records. Specify the following control statement:
FILTER LOGRECORDCODES=(ALL, ¬48)
You want to include all type 40 log records and exclude subtype 4012 records. Specify the following control statement:
FILTER LOGRECORDCODES=(40, ¬4012)
You want to exclude all type 40 log records and include only subtype 4012 records. Specify the following control statement:
FILTER LOGRECORDCODES=(¬40, 4012)
You want to include all type 4 x log records and exclude subtype 4012 records. Specify the following control statement:
FILTER LOGRECORDCODES=(4*, ¬4012)
You want to exclude all type 4 x log records and include only subtype 4012 records. Specify the following control statement:
FILTER LOGRECORDCODES=(¬4*, 4012)

SELECT keyword

You can use the SELECT keyword (or the SEL abbreviation) on the FILTER control statement to specify which LUOWs to include (keep for analysis) or exclude (throw away).

Specify only one instance of the SELECT keyword on a FILTER statement (multiple instances cause a syntax error).

For more information, view the Quick Course "Log Analyzer for IMS - Using SELECT Filters."

SELECT keyword syntax

Specify the SELECT keyword with one or more valid clauses.

A clause consists of a SELECT subkeyword, an operand that is valid for the keyword, and a value. You can specify multiple instances of subkeyword clauses. All subkeyword clauses are optional, but you must specify at least one clause.

Logical operands

You must use one of the following logical operands between the subkeyword and the value:
Use an equal sign (=) to select the record if the value in the record matches the specified value. This operator is valid for all subkeywords except ETIME.

Use a literal NOT and an equal sign (NOT=) to select the record if the value in the record does not match the specified value. This operator is valid for all subkeywords except ETIME.

Use a greater-than sign (>) to select the LUOW if the calculated value is greater than the specified value. This operator is valid for the ETIME keyword.

Use a less-than sign (<) to select the LUOW if the calculated value is less than the specified value. This operator is valid for the ETIME keyword.

Use a combination of a greater-than sign and an equal sign (>= or =>) to select the LUOW if the calculated value is greater than or equal to the specified value. This operator is valid for the ETIME keyword.

Use a combination of a less-than sign and an equal sign (<= or =<) to select the LUOW if the calculated value is less than the specified value. This operator is valid for the ETIME keyword.

**Boolean operators**

To construct complex criteria, you can use Boolean operators (AND and OR) between one expression (a subkeyword and its value) and another expression.

You can group expressions by nesting them within parentheses.

---

**Note**

If you use the AND operator between subkeywords, Log Analyzer selects the LUOW for inclusion only if *both* conditions that the subkeywords specify are true. If you use the OR operator, LUOW selects the LUOW for inclusion if *either* condition is true.

---

**Masked values**

Many SELECT subkeywords support masked values through the use of wildcard characters.

To specify a masked value, specify an asterisk (*) to match any number of characters, starting from that position. Specify a percent sign (%) to match a single character at that position.

**SELECT keyword example**

The following example of a SELECT keyword contains all valid subkeywords.
For each subkeyword, a generic value is shown (except for ABENDS=YES). The operator OR is shown between each subkeyword, meaning that if any of the values is matched, Log Analyzer selects the LUOW for inclusion.

**Figure 21: FILTER control statement example**

```plaintext
FILTER
SELECT=(
    ABENDS=YES
    OR APPC_LTERM=lterm
    OR CICSTRAN=transaction
    OR CLIENTID=clientId
    OR CONTENT=(C'string', logrec, offset, length)
    OR DBD=(dbdName, dsId | dbdName, area | dbdName, area, rba)
    OR DEST=destination
    OR ETIME>seconds
    OR LTERM=lterm
    OR LUNAME=luName
    OR MPR=jobName
    OR NODE=nodeName
    OR OIMS=originatingIms
    OR ORIGIN=source
    OR PORT#=nnnn
    OR PSB=psbName
    OR PSI=pstId
    OR RDBD=(dbdName)
    OR TMEMBER=tmemberName
    OR TPIPE=tpipeName
    OR TPNAME=tpName
    OR TRAN=transaction
    OR UDBD=(dbdName, dsId | dbdName, area | dbdName, area, rba)
    OR UOW1='characterImsID16characterUowID'
    OR USERID=userId
)
```

For more examples, see “Selecting log records and LUOWs” on page 283.

**SELECT subkeywords**

The following table summarizes information about the keywords that you can specify on a FILTER control statement.

<table>
<thead>
<tr>
<th>Subkeyword and variations (abbreviation)</th>
<th>Valid values</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABENDS (ABE)</td>
<td>YES</td>
<td>Search for units of work that have terminated abnormally.</td>
<td>“ABENDS subkeyword” on page 94</td>
</tr>
<tr>
<td>APPC_LTERM 01APPC_LTERM 03APPC_LTERM</td>
<td>lterm</td>
<td>Specify an APPC logical terminal name to find in 01 records, 03 records, or both types of records.</td>
<td>“APPC_LTERM subkeyword” on page 95</td>
</tr>
<tr>
<td>Subkeyword and variations (abbreviation)</td>
<td>Valid values</td>
<td>Description</td>
<td>Topic</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>--------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>CICSTRAN (CTRAN)</td>
<td>transaction</td>
<td>Specify a CICS transaction name to find in log records.</td>
<td>“CICSTRAN subkeyword” on page 95</td>
</tr>
<tr>
<td>CLIENTID</td>
<td>clientId</td>
<td>Specify a client ID to find in 01 records, 03 records, or both types of records.</td>
<td>“CLIENTID subkeyword” on page 95</td>
</tr>
<tr>
<td>CONTENT</td>
<td>(C'string', logrec) (X'string', logrec) (C'string', logrec, offset, length) (X'string', logrec, offset, length)</td>
<td>Specify a search string to find in the specified type of log record. You can also include an offset within the record at which to begin the search and the number of bytes to search.</td>
<td>“CONTENT subkeyword” on page 96</td>
</tr>
<tr>
<td>DBD</td>
<td>dbdName (dbdName, dsid) (dbdName,area) (dbdName,area,rba)</td>
<td>Specify a database description (DBD) name and (as an option) data set ID (DSID) or area to find. In a search for a Fast Path area, you can also specify a relative byte address (RBA). This filter can find DBDs that have been either updated or read.</td>
<td>“DBD subkeyword” on page 97</td>
</tr>
<tr>
<td>DEST 01DEST 03DEST</td>
<td>destination</td>
<td>Specify a destination name to find in 01 records, 03 records, or both types of records.</td>
<td>“DEST subkeyword” on page 98</td>
</tr>
<tr>
<td>ETIME</td>
<td>ssss hh:mm:ss.nnnnnn mm:ss.nnnnnn ss.nnnnnn</td>
<td>Filter logical units of work by a calculated elapsed time.</td>
<td>“ETIME subkeyword” on page 98</td>
</tr>
<tr>
<td>LTERM 01LTERM 03LTERM</td>
<td>lterm</td>
<td>Specify a logical terminal name to find in 01 records, 03 records, or both types of records.</td>
<td>“LTERM subkeyword” on page 99</td>
</tr>
<tr>
<td>LUNAME 01LUNAME 03LUNAME (LUN)</td>
<td>luName</td>
<td>Specify an advanced program-to-program communication (APPC) logical unit (LUNAME) to find in 01 records, 03 records, or both types of records.</td>
<td>“LUNAME subkeyword” on page 100</td>
</tr>
<tr>
<td>Subkeyword and variations (abbreviation)</td>
<td>Valid values</td>
<td>Description</td>
<td>Topic</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>--------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>MPR</td>
<td>jobName</td>
<td>Specify a message processing region (MPR) to find.</td>
<td>“MPR subkeyword” on page 100</td>
</tr>
<tr>
<td>NODE 01NODE 03NODE (NODE)</td>
<td>node</td>
<td>Specify a node name to find in 01 records, 03 records, or both types of records.</td>
<td>“NODE subkeyword” on page 101</td>
</tr>
<tr>
<td>OIMS</td>
<td>originatingIms</td>
<td>Specify the IMSID of an originating IMS system.</td>
<td>“OIMS subkeyword” on page 101</td>
</tr>
<tr>
<td>ORIGIN 01ORIGIN 03ORIGIN (ORIG)</td>
<td>source</td>
<td>Specify a source name to find in 01 records, 03 records, or both types of records.</td>
<td>“ORIGIN subkeyword” on page 102</td>
</tr>
<tr>
<td>PORT# 01PORT# 03PORT# (PORT)</td>
<td>nnnn</td>
<td>Specify an Open Transaction Manager Access (OTMA) port number to find in 01 records, 03 records, or both types of records. OTMA sources can include IMS Connect, WebSphere for MQ, and other types of activity.</td>
<td>“PORT# subkeyword” on page 102</td>
</tr>
<tr>
<td>PSB</td>
<td>psbName</td>
<td>Specify a program specification block (PSB) name to find.</td>
<td>“PSB subkeyword on the FILTER control statement” on page 103</td>
</tr>
<tr>
<td>PST</td>
<td>pstId</td>
<td>Specify a partition specification table (PST) identifier to find.</td>
<td>“PST subkeyword” on page 103</td>
</tr>
<tr>
<td>RDBD</td>
<td>dbdName</td>
<td>Specify a database description (DBD) name to find. This filter can find DBDs that have been read but not updated.</td>
<td>“RDBD subkeyword” on page 103</td>
</tr>
<tr>
<td>TMEMBER 01TMEMBER 03TMEMBER (TMEM)</td>
<td>tmemberName</td>
<td>Specify an OTMA transaction member (TMEMBER) name to find in 01 records, 03 records, or both types of records.</td>
<td>“TMEMBER subkeyword” on page 104</td>
</tr>
<tr>
<td>Subkeyword and variations (abbreviation)</td>
<td>Valid values</td>
<td>Description</td>
<td>Topic</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>--------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>TPIPE 01TPIPE 03TPIPE</td>
<td>tpipeName</td>
<td>Specify an OTMA transaction pipe (TPIPE) name to find in 01 records, 03 records, or both types of records.</td>
<td>&quot;TPIPE subkeyword&quot; on page 104</td>
</tr>
<tr>
<td>TPNAME 01TPNAME 03TPNAME (TPN)</td>
<td>tpName</td>
<td>Specify an APPC transaction program name (TPNAME) to find in 01 records, 03 records, or both types of records.</td>
<td>&quot;TPNAME subkeyword&quot; on page 105</td>
</tr>
<tr>
<td>TRAN 01TRAN 03TRAN</td>
<td>transaction</td>
<td>Specify a transaction name to find in 01 records, 03 records, or both types of records.</td>
<td>&quot;TRAN subkeyword&quot; on page 105</td>
</tr>
<tr>
<td>UDBD</td>
<td>dbdName</td>
<td>Specify a database description (DBD) name and (as an option) data set ID (DSID) or area to find. In a search for a Fast Path area, you can also specify a relative byte address (RBA). This filter can find DBDs that have been updated but not read.</td>
<td>&quot;UDBD subkeyword&quot; on page 106</td>
</tr>
<tr>
<td></td>
<td>(dbdName, dsId)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(dbdName,area)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(dbdName,area,rba)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>8charImslsID16charUowID</td>
<td>Specify a unit of work to find.</td>
<td>&quot;UOW1 subkeyword&quot; on page 106</td>
</tr>
<tr>
<td>USERID 01USERID 03USERID (USER)</td>
<td>userID</td>
<td>Specify a user ID to find.</td>
<td>&quot;USERID subkeyword&quot; on page 107</td>
</tr>
</tbody>
</table>

**ABENDS subkeyword**

To filter LUOWs by whether the LUOW has terminated abnormally, you can use the ABENDS subkeyword (or the ABE abbreviation) on the SELECT keyword on the FILTER control statement.

If you specify this subkeyword, you must specify the value YES. Log Analyzer checks a bit setting in 07 records to determine whether a transaction abend has occurred.
APPC_LTERM subkeyword

To filter LUOWs by APPC logical terminal name, you can use the APPC_LTERM subkeyword (or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified APPC LTERM in the origin field and the destination field in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01APPC_LTERM subkeyword.
- To search for the value in 03 records, specify the 03APPC_LTERM subkeyword.
- To search for the value in 01 records and 03 records, specify the APPC_LTERM subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

The value that you specify must reflect a name that is a printable EBCDIC value.

CICSTRAN subkeyword

To filter LUOWs by CICS transaction name, you can use the CICSTRAN subkeyword (or the CTRAN abbreviation or its variations) on the SELECT keyword on the FILTER control statement.

You can specify a masked value (using * and % masking characters) for this subkeyword.

CLIENTID subkeyword

To filter LUOWs by OTMA client identifier, you can use the CLIENTID subkeyword (or its variations) on the SELECT keyword on the FILTER control statement.

If the message contains a client ID and the message passes to IMS through IMS Connect, Log Analyzer searches for the specified ID in the destination field in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01CLIENTID subkeyword.
- To search for the value in 03 records, specify the 03CLIENTID subkeyword.
To search for the value in 01 records and 03 records, specify the CLIENTID subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

**CONTENT subkeyword**

To filter LUOWs by content, you can use the CONTENT subkeyword on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified target string in the specified log record type (or types). You can specify an offset within the record at which to begin the search and the number of bytes to search.

Values of the CONTENT subkeyword specify how to search the content. You must enclose all values for the CONTENT subkeyword in parentheses. You must specify the target string for the search and the type of log record to search:

SELECT=CONTENT=(C'string',logrec) or SELECT=CONTENT=(X'string',logrec)

To specify the target string, use a literal C to indicate that the string is specified in EBCDIC character format. Use a literal X to indicate that the string is specified in hexadecimal format. Enclose the string in single quote marks (').

To specify the type of log record to search, specify the two-character log record code. You can use an asterisk to mask the second character of the log record code. For example, specify 3* to search for the target string in log record types 30, 31, 32, and so on. To search for the target string in all types of eligible log records, specify asterisks in both positions (**).

You can specify the offset (position) within the record to start searching for the target string and specify the length of data to search, beginning at the specified offset:

SELECT=CONTENT=(C'string',logrec,offset,length)

The offset is relative to 0; if you specify 1, the search starts at the second byte of the record.

If you omit the offset value, for all log records except 01 and 03 records, Log Analyzer searches the entire log record, starting at the beginning of the record. For 01 and 03 records, the CONTENT0103 keyword controls how Log Analyzer handles the search. For more information, see “CONTENT0103 keyword” on page 84.
DBD subkeyword

You can use the DBD subkeyword on the SELECT keyword on the FILTER control statement to filter LUOWs by database description (DBD) name and (as an option) by database data set ID (DSID), which is also known as the data set group number (DSG) or by Fast Path area name.

If you specify an area name, you can also specify a relative byte address (RBA) as an option.

- SELECT=DBD=(dbdName,dsID) or
- SELECT=DBD=(dbdName,areaName) or
- SELECT=DBD=(dbdName,areaName,rba)

Use the DBD subkeyword to find databases that have been either updated or read during the time span of the analysis. Use the RDBD subkeyword to find only the databases that have been read; use the UDBD subkeyword to find only the databases that have been updated.

Log Analyzer searches for the specified values as follows:

- DBD and DSID in type 50 records
- DBD and area name (and RBA) in type 59 records
- DBD in MainView type FA records
- DBD in type 20, 21, 24, 25, 26, and 27 records
- DSID in type 20, 21, and 27 records

These types of log records are placed in LUOWs that are separate from transaction-type LUOWs.

To report information about read-only databases, Log Analyzer uses type FA records that the BMC MainView for IMS Offline product creates. To obtain information about database read activity in an analysis, ensure that MainView has produced the required records during the analysis time span. Also, ensure that FA records are included in any log record code filters that you explicitly specify. Log Analyzer automatically includes FA records in transactional LUOWs if you do not specify a log record code filter.

Omit the DSID or area name value to search for the DBD name but not a DSID or area name:

SELECT=DBD=dbdName

You can specify masked values (using * and % masking characters) for the DBD name and area name values of this subkeyword.
If you specify an RBA for a Fast Path area, the RBA must follow the area name. Specify an even number of characters to represent the hexadecimal RBA. You can specify a maximum of 8 characters. If you specify fewer than 8 characters, Log Analyzer pads the value to the left with zeros. For example, if you specify 1111 as the value, Log Analyzer uses the value 00001111.

**DEST subkeyword**

To filter LUOWs by destination name, you can use the DEST subkeyword (or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified name in the destination field in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01DEST subkeyword.
- To search for the value in 03 records, specify the 03DEST subkeyword.
- To search for the value in 01 records and 03 records, specify the DEST subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

The value that you specify must reflect a destination name that is a printable, EBCDIC value. If you must specify an APPC or OTMA destination name, use the APPC_LTERM, PORT, TPIPE, TMEMBER, LUNAME, or TPNAME subkeyword as appropriate.

**ETIME subkeyword**

To filter LUOWs by a calculated elapsed time, you can use the ETIME subkeyword on the SELECT keyword on the FILTER control statement.

This filter is not associated with a specific log record type.

Log Analyzer calculates the elapsed time for an LUOW by subtracting the timestamp of the first record that is included in the LUOW from the timestamp of the last record that is included in the LUOW.
WARNING
All logs that are included in the analysis must have been produced on systems that are using the same sysplex timer. If the systems are using different sysplex timers, do not use the ETIME subkeyword.

Specify one of the following operands for the ETIME subkeyword:

- > (greater than)
- < (less than)
- >= or => (greater than or equal to)
- <= or =< (less than or equal to)

The = and NOT= operands are not valid for the ETIME keyword.

Specify a value for the ETIME keyword in one of the following formats:

- ssss
- hh:mm:ss.nnnnnn
- mm:ss.nnnnnn
- ss.nnnnnn

In these values, the variables represent the following elements:

- ssss is a number of seconds (0 through 3600; 3600 is one hour).
- hh is the number hours (0 through 24).
- mm is the number of minutes (0 through 60).
- ss is the number of seconds (0 through 60).
- nnnnnn is the number of bits of a second (0 through 999999).

For example, specify SELECT=ETIME>10 to find all LUOWs that have a calculated elapsed time which is greater than 10 seconds.

**LTERM subkeyword**

To filter LUOWs by logical terminal name, you can use the LTERM subkeyword (or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified LTERM in the origin field and the destination field in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01LTERM subkeyword.
- To search for the value in 03 records, specify the 03LTERM subkeyword.
To search for the value in 01 records and 03 records, specify the LTERM subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

The value that you specify must reflect a name that is a printable EBCDIC value. If you must specify an APPC or OTMA name, use the APPC_LTERM, PORT, TPIPE, TMEMBER, LUNAME, or TPNAME subkeyword as appropriate.

**LUNAME subkeyword**

To filter LUOWs by advanced program-to-program communication (APPC) logical unit (LUNAME) name, you can use the LUNAME subkeyword (or the LUN abbreviation or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified LUNAME name in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01LUNAME subkeyword.
- To search for the value in 03 records, specify the 03LUNAME subkeyword.
- To search for the value in 01 records and 03 records, specify the LUNAME subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

**MPR subkeyword**

To filter LUOWs by the message processing region (MPR) that is associated with transactions of interest, you can use the MPR subkeyword on the SELECT keyword on the FILTER control statement.

Log Analyzer searches all 07 records for this MPR identifier.

Specify the one- to eight-character jobname of the MPR. You can specify a masked value (using * and % masking characters) for this subkeyword.
NODE subkeyword

To filter LUOWs by VTAM node name, at the logical unit (LU) or communications line block (CLB) level, you can use the NODE subkeyword (or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified source name in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01NODE subkeyword.
- To search for the value in 03 records, specify the 03NODE subkeyword.
- To search for the value in 01 records and 03 records, specify the NODE subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

Log Analyzer searches for the specified node name in 01 records and 03 records (as requested) and places these records in transaction-type LUOWs. If you specify the NODE keyword, Log Analyzer always searches for the specified node name in type 02, 10, 11, 12, 13, 16, 28, 63, 66, DE, and EA log records (unless you exclude them by using the LOGRECORDCODES keyword). Log Analyzer also associates a type 72 record with a node. These types of log records are placed in LUOWs that are separate from transaction-type LUOWs.

OIMS subkeyword

To filter LUOWs by the IMS ID of an originating IMS system that is associated with transactions of interest, you can use the OIMS subkeyword on the SELECT keyword on the FILTER control statement.

Log Analyzer might identify originating IMS systems as "other" IMS systems in the Analysis Summary report. In this report, the Other IMSIDs Encountered field lists IMS systems for which Log Analyzer detected activity but for which logs spanning the same time range were not processed during this analysis job. You can use the OIMS subkeyword to select transactions in which one of these "other" IMS systems was involved.

Log Analyzer searches for the specified IMSID in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01OIMS subkeyword.
- To search for the value in 03 records, specify the 03OIMS subkeyword.
To search for the value in 01 records and 03 records, specify the OIMS subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

**ORIGIN subkeyword**

To filter LUOWs by source name, you can use the ORIGIN subkeyword (or the ORIG abbreviation or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified source name in the origin field in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01ORIGIN subkeyword.
- To search for the value in 03 records, specify the 03ORIGIN subkeyword.
- To search for the value in 01 records and 03 records, specify the ORIGIN subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

The value that you specify must reflect a source name that is a printable EBCDIC value. If you must specify an APPC or OTMA source name, use the APPC_LTERM, PORT, TPIPE, TMEMBER, LUNAME, or TPNAME subkeyword as appropriate.

**PORT# subkeyword**

This subkeyword is applicable to IMS Connect environments only.

To filter LUOWs by OTMA port number, you can use the PORT# subkeyword (or the PORT abbreviation or its variations) on the SELECT keyword on the FILTER control statement. Log Analyzer searches for the specified user ID in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01PORT# subkeyword.
- To search for the value in 03 records, specify the 03PORT# subkeyword.
- To search for the value in 01 records and 03 records, specify the PORT# subkeyword.
You can specify * as a masking character for this subkeyword.

**PSB subkeyword on the FILTER control statement**

To filter LUOWs by program specification block (PSB) name, you can use the PSB subkeyword on the SELECT keyword on the FILTER control statement.

Log Analyzer searches all 07 and 08 records for this PSB name.

You can specify a masked value (using * and % masking characters) for this subkeyword.

**PST subkeyword**

To filter LUOWs by the partition specification table (PST) identifier that is assigned to an IMS region of interest, you can use the PST subkeyword on the SELECT keyword on the FILTER control statement.

Log Analyzer searches all 07 records for this PST identifier.

Specify a one- to five-digit number. You can specify * as a masking character for this subkeyword.

**RDBD subkeyword**

You can use the RDBD subkeyword on the SELECT keyword on the FILTER control statement to filter LUOWs by the database description (DBD) name:

```
SELECT=RDBD=(dbdName)
```

Use the RDBD subkeyword to find only the databases that have been read (but not updated) during the time span of the analysis. Use the UDBD subkeyword to find only the databases that have been updated; use the DBD subkeyword to find the databases that have been either updated or read.

To report information about read-only databases, Log Analyzer uses type FA records that the BMC MainView for IMS Offline product creates. To obtain information about database read activity in an analysis, ensure that MainView has produced the required records during the analysis time span. Also, ensure that FA records are included in any log record code filters that you explicitly specify. Log Analyzer automatically includes FA records in transactional LUOWs if you do not specify a log record code filter.
You can specify masked values (using * and % masking characters) for the value of this subkeyword.

**TMEMBER subkeyword**

To filter LUOWs by OTMA transaction member (TMEMBER) name, you can use the TMBER subkeyword (or the TMEM abbreviation or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified TPIPE name in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01TMEMBER subkeyword.
- To search for the value in 03 records, specify the 03TMEMBER subkeyword.
- To search for the value in 01 records and 03 records, specify the TMEMBER subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

**TPIPE subkeyword**

To filter LUOWs by OTMA transaction pipe (TPIPE) name, you can use the TPIPE subkeyword (or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified TPIPE name in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the 01TPIPE subkeyword.
- To search for the value in 03 records, specify the 03TPIPE subkeyword.
- To search for the value in 01 records and 03 records, specify the TPIPE subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.
**TPNAME subkeyword**

To filter LUOWs by APPC TPNAME name, you can use the TPNAME subkeyword (or the TPN abbreviation or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified TPNAME name in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the **01TPNAME** subkeyword.
- To search for the value in 03 records, specify the **03TPNAME** subkeyword.
- To search for the value in 01 records and 03 records, specify the **TPNAME** subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

**TRAN subkeyword**

To filter LUOWs by transaction name, you can use the TRAN subkeyword (or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified transaction name in the destination field in 01 records, in 03 records, or in both types of records:

- To search for the value in 01 records, specify the **01TRAN** subkeyword.
- To search for the value in 03 records, specify the **03TRAN** subkeyword.
- To search for the value in 01 records and 03 records, specify the **TRAN** subkeyword.

You can specify a masked value (using * and % masking characters) for this subkeyword.

The value that you specify must reflect a destination name that is a printable EBCDIC value. If you must specify an APPC or OTMA destination name, use the APPC_LTERM, PORT, TPIPE, TMEMBER, LUNAME, or TPNAME subkeyword as appropriate.
**UDBD subkeyword**

You can use the UDBD subkeyword on the SELECT keyword on the FILTER control statement to filter LUOWs by database description (DBD) name and (as an option) by database data set ID (DSID) or by Fast Path area name. (The DSID is also known as the data set group number, or DSG.) If you specify an area name, you can also specify a relative byte address (RBA) as an option.

```
SELECT=UDBD=(dbdName,dsId) or
SELECT=UDBD=(dbdName,areaName) or
SELECT=UDBD=(dbdName,areaName,rba)
```

Use the UDBD subkeyword to find only the databases that have been updated during the time span of the analysis. Use the RDBD subkeyword to find only the databases that have been read; use the DBD subkeyword to find the databases that have been either updated or read.

The syntax of the UDBD keyword is identical to the syntax of the DBD keyword. For more information, see “DBD subkeyword” on page 97.

**UOW1 subkeyword**

You can use the UOW1 subkeyword on the SELECT keyword on the FILTER control statement to focus your analysis on a specified unit of work (UOW).

In reports, Log Analyzer displays UOW information in the **UOW-1** field. Specify this subkeyword as follows:

```
SELECT=UOW1=’8characterImsID16characterUowID’
```

- Enclose the entire value in single quotes.
- Specify the value as follows (the same way that Log Analyzer displays **UOW-1** field values in reports):
  - The first eight characters are the IMS ID. Log Analyzer interprets them as alphanumeric characters. If your IMS ID is less than eight characters long, insert blanks to pad the remaining positions.
  - The last 16 characters are the UOW identifier. Log Analyzer interprets the specified characters as hexadecimal digits.

In the following example, the IMS ID is IML6, so the statement includes four blanks after IML6:
SELECT=UOW1='IML6 C361E35F71A2D206'

The UOW1 filter is valid with other filters.

Log Analyzer checks the following log record types for the UOW value:

- 01
- 03
- 31
- 35
- Certain 59 subcodes for Fast Path

USERID subkeyword

To filter LUOWs by user ID, you can use the USERID subkeyword (or the USER abbreviation or its variations) on the SELECT keyword on the FILTER control statement.

Log Analyzer searches for the specified user ID in the security segment of 01 records, 03 records, or both types of records:

- To search for the value in 01 records, specify the 01USERID subkeyword.
- To search for the value in 03 records, specify the 03USERID subkeyword.
- To search for the value in 01 records and 03 records, specify the USERID subkeyword.

Log Analyzer searches for the specified user ID in 01 records and 03 records (as requested) and places these records in transaction-type LUOWs. If you specify the USERID keyword, Log Analyzer always searches for the specified user ID in type 02, 10, 11, 12, 13, 16, 62, 72, DE, and EA log records (unless you exclude them by using the LOGRECORDCODES keyword). These types of log records are placed in LUOWs that are separate from transaction-type LUOWs.

You can specify a masked value (using * and % masking characters) for this subkeyword.

Specifying REPORTS control statements

The REPORTS control statement (or the REP abbreviation) controls production of the batch reports that Log Analyzer writes to SYSOUT-type data sets.
The REPORTS control statement has no effect on the interactive analysis of data in the ISPF interface. The INDEXFILE control statement and the EXTRACT control statement control the availability of data for interactive analysis.

The REPORTS statement is optional. If you omit it, Log Analyzer writes the Analysis Summary report to the SUMMARY data set by default.

A REPORTS statement must follow the ANALYZE statement and precede the END control statement, but you can specify it in any position relative to other control statements.

**REPORTS control statement**

The following example of a REPORTS control statement contains all valid keywords.

For each keyword, if a default value is provided, this value is shown; if no default value is provided, a generic value is shown. Some keywords are shown as comments because they cannot be used at the same time as other keywords.

**Figure 22: REPORTS control statement example**

```plaintext
REPORTS
  *  ABEND=limit
  AUDIT=limit
  APPCHECK=subkeywords
  *  DEADLOCK=ALL
  *  DUMPFORM=BOTH
  DUMPREC=(limit,offset,length)
  *  DUMPVERT=(limit,offset,length)
  LOGRECORDCODES=ALL
  LUOWDETAIL=limit
  LUOWDETAILDATA=(offset,length)
  LUOWSUMM=limit
  LUOWTSEQ=limit
  OFFSET=HEX
  ORPHANS=limit
  *  RBASTATS=ALL
  SUMMARY=limit
```

For more examples, see “Producing batch reports” on page 291.

The following table summarizes information about the keywords that you can specify on a REPORTS control statement.
<table>
<thead>
<tr>
<th>Keyword (abbreviation)</th>
<th>Valid values</th>
<th>Default</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABEND</td>
<td>ALL</td>
<td>None</td>
<td>Control production of the ABEND report.</td>
<td>“ABEND keyword” on page 110</td>
</tr>
<tr>
<td>APPCHECK</td>
<td>ALL expressions</td>
<td>None</td>
<td>Control production of the Application Checkpoint report.</td>
<td>“APPCHECK keyword” on page 111</td>
</tr>
<tr>
<td>AUDIT</td>
<td>ALL expressions</td>
<td>None</td>
<td>Control production of the Auditing report.</td>
<td>“AUDIT keyword” on page 115</td>
</tr>
<tr>
<td>DEADLOCK</td>
<td>ALL</td>
<td>None</td>
<td>Control production of the Deadlocks report.</td>
<td>“DEADLOCK keyword” on page 118</td>
</tr>
<tr>
<td>DUMPFORM (DUMPF)</td>
<td>BOTH CHAR HEX</td>
<td>BOTH</td>
<td>Specify whether to show character data, hexadecimal data, or both kinds of data in the Record Dumps report in vertical format.</td>
<td>“DUMPFORM keyword” on page 119</td>
</tr>
<tr>
<td>DUMPREC (DUMPR)</td>
<td>(limit, offset, length)</td>
<td>None</td>
<td>Control production of the Record Dumps report in standard format.</td>
<td>“DUMPREC keyword” on page 119</td>
</tr>
<tr>
<td>DUMPVERT (DUMPV)</td>
<td>(limit, offset, length)</td>
<td>None</td>
<td>Control production of the Record Dumps report in vertical format.</td>
<td>“DUMPVERT keyword” on page 120</td>
</tr>
<tr>
<td>LOGRECORDCODES (LRC)</td>
<td>ALL</td>
<td>ALL</td>
<td>Report information about the specified types of log records from within the set of log records that were included LUOWs that Log Analyzer has created.</td>
<td>“LOGRECORDCODES keyword on the REPORTS control statement” on page 122</td>
</tr>
<tr>
<td>LUOWDETAIL (LDET)</td>
<td>ALL</td>
<td>None</td>
<td>Control production of the LUOW Detail report</td>
<td>“LUOWDETAIL keyword” on page 123</td>
</tr>
<tr>
<td>Keyword (abbreviation)</td>
<td>Valid values</td>
<td>Default</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------</td>
<td>---------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>LUOWDETAILDATA (LDETDATA)</td>
<td>(offset, length)</td>
<td>None</td>
<td>Specify the location (within log records) and amount of data to include in the LUOW Detail report.</td>
<td></td>
</tr>
<tr>
<td>LUOWSUMM (LSUMM)</td>
<td>ALL nnnnnnnnnn</td>
<td>None</td>
<td>Control production of the LUOW Summary report.</td>
<td></td>
</tr>
<tr>
<td>LUOWTSEQ (LTS)</td>
<td>ALL nnnnnnnnnn</td>
<td>None</td>
<td>Control production of the Time Sequence report.</td>
<td></td>
</tr>
<tr>
<td>OFFSET</td>
<td>HEX DECIMAL or DEC</td>
<td>HEX</td>
<td>Specify the format of offset fields in the Record Dumps report.</td>
<td></td>
</tr>
<tr>
<td>ORPHANS</td>
<td>ALL nnnnnnnnnn</td>
<td>None</td>
<td>Control production of the Orphans report.</td>
<td></td>
</tr>
<tr>
<td>RBASTATS</td>
<td>ALL TOTALS</td>
<td>None</td>
<td>Control production of the RBA Buffer Statistics report.</td>
<td></td>
</tr>
<tr>
<td>SUMMARY (SUMM)</td>
<td>ALL nnnnnnnnnn</td>
<td>None</td>
<td>Control production of the Analysis Summary report.</td>
<td></td>
</tr>
</tbody>
</table>

**ABEND keyword**

You can use the ABEND keyword on the REPORTS control statement to control production of the ABEND report.

Log Analyzer writes this report to the SYSOUT-type ABEND data set.

The ABEND report is a single-record report. To produce it, Log Analyzer focuses exclusively on relevant records and does not create LUOWs. If you specify the ABEND keyword, the following specifications are invalid:

- Other keywords on the REPORTS control statement
- Filters specified with the SELECT keyword on the FILTER control statement
- INDEXFILE control statement
- EXTRACT control statement
Specify one of the following values for the ABEND keyword:

- **ALL or 999999999**
  
  If you do not want to limit the number of records that are written to the output data set, specify ALL or specify 999999999.

- **nnnnnnnnn**
  
  To limit the number of output records, enter the number of records (1 through 999999999) to write to the data set.

No default value is provided. If you omit this keyword (but you specify the REPORTS control statement), Log Analyzer does not produce this report.

For more information, see “Using the ABEND report” on page 202.

**APPCHECK keyword**

You can use the APPCHECK keyword on the REPORTS control statement to control production of the Application Checkpoint report.

Log Analyzer writes this report to the SYSOUT-type APPCHECK data set.

**Note**

Log Analyzer produces the Application Checkpoint report by using filtering techniques that are different from the techniques for producing most other reports. The FILTER control statement is invalid if you specify the APPCHECK keyword.

To produce the Application Checkpoint report, specify APPCHECK=ALL or APPCHECK=expressions as explained in this topic.

No default value is provided for the APPCHECK keyword. If you omit this keyword (but you specify the REPORTS control statement), Log Analyzer does not produce this report.

**APPCHECK=ALL**

To produce the Application Checkpoint report with default filters, specify APPCHECK=ALL. With this keyword, Log Analyzer includes all batch message processing (BMP) jobs that are found in the input logs, and does not filter the information by checkpoint frequency.
**APPCHECK**=expressions

To filter the Application Checkpoint report by job name, program specification block (PSB) name, checkpoint frequency, or any combination of these filters, specify APPCHECK=expressions.

The following syntax rules apply to these expressions:

- An expression consists of a subkeyword, an operator, and a value (or values).
- Enter at least one space between the subkeyword and the operator, and between the operator and the value.
- In expressions that permit multiple values, use a comma (not a space) to separate values.
  
  If you use the = (equal to) operator in an expression that contains multiple values, Log Analyzer processes the values with OR logic. (An item that matches any one of the specified values results in a match for the expression.)
  
  If you use the NOT= (not equal to) operator in an expression that contains multiple values, Log Analyzer processes the values with AND logic. (Any item that matches every specified value results in a match for the expression.)

- Some expressions support wildcard characters:
  
  — Specify an asterisk (*) to match any number of characters, starting from that position.
  
  — Specify a percent sign (%) to match a single character at that position.

- Specify only one instance of each expression (as identified by the subkeyword). You can specify expressions in any order.

- Use at least one space to separate an expression from the next expression. Log Analyzer processes multiple expressions with AND logic. (Any item that matches every specified expression results in a match for the entire set of filters; the item is included in the report.)

- You can enclose all expressions within a single set of optional parentheses.

**JOB subkeyword**

You can use the JOB subkeyword in an APPCHECK expression to include or exclude BMP jobs, based on their job names.

Use the following syntax:

```
JOB operator jobName,jobName,....
```
For *operator*, specify one of the following operators:

- **=**
  
  This operator includes information for BMP jobs with the specified job names and omits information for all other jobs that Log Analyzer finds in the input logs.

- **NOT=**
  
  This operator omits information for BMP jobs with the specified job names and includes information for all other jobs that Log Analyzer finds in the input logs.

For *jobName*, specify the name of each job to include or exclude. You can specify multiple names (separated by commas). You can use wildcards (* and %).

### PSB subkeyword in an APPCHECK expression

You can use the PSB subkeyword in an APPCHECK expression to include or exclude BMP jobs, based on their program specification block (PSB) names.

Use the following syntax:

```
PSB operator psbName,psbName,...
```

For *operator*, specify one of the following operators:

- **=**
  
  This operator includes information for BMP jobs with the specified PSB names and omits information for all other BMP jobs that Log Analyzer finds in the input logs.

- **NOT=**
  
  This operator omits information for BMP jobs with the specified PSB names and includes information for all other BMP jobs that Log Analyzer finds in the input logs.

For *psbName*, specify the name of each PSB to include or exclude. You can specify multiple names (separated by commas). You can use wildcards (* and %).

### CHKFREQ subkeyword

You can use the CHKFREQ subkeyword in an APPCHECK expression to filter the information in the Application Checkpoint report, based on checkpoint frequency.

To establish a checkpoint frequency filter, specify a comparison operator and the threshold number of checkpoints per time unit. (You use the THRESH subkeyword to specify the time unit.) Log Analyzer calculates the checkpoint frequency for an included BMP job by dividing the number of checkpoint records by the elapsed time.
of the job (as reflected in the input log records). Then, Log Analyzer compares this calculated checkpoint frequency with the threshold number of checkpoints per time unit. If the comparison results in a true statement, the report displays information about the BMP job.

Use the following syntax:

```
CHKFREQ operator nnnn
```

For `operator`, specify one of the following comparison operators:

- `<`
  - This operator displays data for this BMP job if the calculated checkpoint frequency is less than the specified threshold value.

- `<=` or `=<`
  - This operator displays data for this BMP job if the calculated checkpoint frequency is less than or equal to the specified threshold value.

- `>`
  - This operator displays data for this BMP job if the calculated checkpoint frequency is greater than the specified threshold value.

- `>=` or `=>`
  - This operator displays data for this BMP job if the calculated checkpoint frequency is greater than or equal to the specified threshold value.

For `nnnn`, specify the threshold number of checkpoints to compare with the calculated frequency of checkpoints. Valid values are 0 through 9999.

### THRESH subkeyword

You can use the THRESH subkeyword in an APPCHECK expression to specify the time unit to use for the threshold value in a checkpoint frequency filter.

Use the following syntax:

```
THRESH = value
```

Valid values are MIN (minutes) and SEC (seconds). If you omit the THRESH subkeyword, Log Analyzer uses the default value MIN.

### APPCHECK keyword examples

The following examples illustrate the use of the APPCHECK keyword.
Example
To find all BMPs that have job names starting with ABC or XYZ and that issued more than 200 checkpoints each minute, specify the following keyword and subkeywords on the REPORTS control statement:

```plaintext
APPCHECK=(JOB = ABC*,XYZ* CHKFREQ > 200)
```

Example
To find all BMPs that issued more than 10 checkpoints each second, specify the following keyword and subkeywords on the REPORTS control statement:

```plaintext
APPCHECK=(CHKFREQ > 10 THRESH = SEC)
```

For more information, see “Obtaining information about application checkpoints” on page 323 and “Using the Application Checkpoint report” on page 206.

**AUDIT keyword**

You can use the AUDIT keyword on the REPORTS control statement to control production of the Auditing report. Log Analyzer writes this report to the SYSOUT-type AUDIT data set.

Log Analyzer produces the AUDIT report using different filtering techniques than those used for producing other reports. The AUDIT report filters are determined by the FILTER control statement. When used together with the APPCHECK keyword, Log Analyzer will respond to the APPCHECK filters.

**Note**

The AUDIT keyword has no default. If you omit this keyword while specifying the REPORTS control statement, Log Analyzer does not produce this report.

**AUDIT=ALL or 999999999**

You can delimit the number of records that are written to the output data set, by specifying **ALL** or **999999999**. When you specify this keyword, Log Analyzer includes all of the following types of event captured from the IMS Logs:

- Commands
- Signons
- Signoffs
- Security errors
- Transactions
- Databases updated and read by transactions

**AUDIT=nnnnnnnnnn**

You can limit the number of output records by specifying `nnnnnnnnnn`.

**AUDIT=expressions**

You can filter the AUDIT report by `type` or `dbd` subkeyword filters by specifying `AUDIT=expressions`. The following syntax rules apply to `AUDIT=expressions`:

- An expression consists of a subkeyword, an operator, and one or more values.
- Enter at least one space between the subkeyword and the operator, and between the operator and the value.
- In expressions that permit multiple values, separate values *with a comma* (not a space).
  
  If you use the `=` (equal to) operator in an expression that contains multiple values, Log Analyzer processes the values with **OR** logic. An item that matches any *one* of the specified values results in a match for the expression.
  
  If you use the `NOT=` (not equal to) operator in an expression that contains multiple values, Log Analyzer processes the values with **AND** logic. Any item that matches *every* specified value results in a match for the expression.

- Some expressions support wildcard characters:
  
  - Specify an asterisk (`*`) to match any number of characters, starting from that position.
  
  - Specify a percent sign (`%`) to match a single character at that position.

- Specify only one instance of each expression (as identified by the subkeyword). You can specify expressions in any order.

- Use at least one space to separate an expression from the next expression. Log Analyzer processes multiple expressions with **AND** logic. Any item that matches *every* specified expression results in a match for the entire set of filters; the item is included in the report.

- You can enclose all expressions within a single set of optional parentheses.

You can apply one of the following subkeywords:
TYPE

Use the TYPE subkeyword in an AUDIT expression to include or exclude types of event captured from the IMS Logs. Use the following syntax:

\texttt{TYPE operator VALUE}

\textit{operator}

Specify one of the following operators:

- \texttt{=}  
  This operator includes all information for TYPE of events that are defined by the \textit{value} that you specify.

- \texttt{NOT=}  
  This operator omits information for TYPE of events defined by the \textit{value} that you specify, but includes information for all other TYPE of events.

\textit{VALUE}

Specify one of the following values:

- \texttt{TRAN}
- \texttt{COMMAND}
- \texttt{SECVIO}
- \texttt{SIGNON}
- \texttt{SIGNOFF}

DBD

You can use the DBD subkeyword in an AUDIT expression to include or exclude DBDs, based on their DBD names. Use the following syntax:

\texttt{DBD operator dbdName, dbdName, ...}

\textit{operator}

Specify one of the following qualifiers:

- \texttt{=}  
  Includes information for DBDs with the specified DBD names, and omits information for all other DBDs in the input logs.
- **NOT =**
  Omits information for DBDs of with the specified DBD names and includes information for all other DBDs in the input logs.

  **Note**
  For **dbd.Name**, Specify the name of each DBD to include or exclude. You can specify multiple names (separated by commas), and you can use wildcards (* and %).
  If you use **TYPE NOT=TRAN** or **TYPE=COMMAND | SINON | SIGNOFF | SECVIO**, the DBD will be ignored.

---

**AUDIT keyword examples**

The following examples illustrate the use of the AUDIT keyword.

**Example**
To find all TRANs that have dbd names starting with **ABC** or **XYZ**, specify the following keyword and subkeywords on the REPORTS control statement:

```
AUDIT=(TYPE = TRAN DBD = ABC*,XYZ*)
```

**Example**
To find all command events, specify the following keyword and subkeywords on the **REPORTS** control statement:

```
AUDIT=(TYPE = COMMAND )
```

---

**DEADLOCK keyword**

You can use the DEADLOCK keyword on the REPORTS control statement to control production of the Deadlocks report.

Log Analyzer writes this report to the SYSOUT-type DEADLOCK data set.

The Deadlocks report is a single-record report. To produce it, Log Analyzer focuses exclusively on relevant records and does not create LUOWs. If you specify the DEADLOCK keyword, the following specifications are invalid:

- Other keywords on the REPORTS control statement
- FILTER control statement
- INDEXFILE control statement
- EXTRACT control statement

Specify the value ALL to produce the Deadlocks report with all available data.
No default value is provided. If you omit this keyword (but you specify the REPORTS control statement), Log Analyzer does not produce this report.

For more information, see “Using the Deadlocks report” on page 203.

**DUMPFORM keyword**

If you use the DUMPVERT keyword, you can also use the DUMPFORM keyword (or the DUMPF keyword) on the REPORTS control statement to control the kind of data to display in the Record Dumps report (in vertical format).

Specify one of the following values:

- **BOTH**
  Display character data and hexadecimal data in the report.

- **CHAR**
  Display only character data in the report.

- **HEX**
  Display only hexadecimal data in the report.

The default is DUMPFORM=BOTH.

**DUMPREC keyword**

You can use the DUMPREC keyword (or the DUMPR abbreviation) on the REPORTS control statement to control production of the Record Dumps report (in standard dump format).

Log Analyzer writes this report to the SYSOUT-type LUOWDMPS data set.

Specify the keyword as follows:

```
DUMPREC=(limit,offset,length)
```

- Specify one of the following required `limit` values:
  - **ALL** or **999999999**
    Do not limit the number of lines that are written to the output report data set.
Limit the number of output lines in the report to the specified number.

**Note**
The Record Dumps report can be large. You might want to limit the number of output records until you are certain that specified selection criteria are correct.

- You can specify optional *offset* and *length* values to have Log Analyzer report only a portion of the record content.

  If you omit the offset and length, Log Analyzer reports the entire record content.

- To indicate the starting position of the content to report, use the *offset* value. The offset is relative to 0 (if you specify 1 as the value, the starting position is the second byte of the record).

- To indicate the number of bytes of content to report, use the *length* value. If any portion of the target content (which is identified by the specified offset and length) falls beyond the end of the record, Log Analyzer ignores the specified offset and length.

No default values are provided. If you omit this keyword, Log Analyzer does not produce this report.

If you specify the DUMPREC keyword, you must also specify the LUOWDETAIL keyword (see “LUOWDETAIL keyword” on page 123). Log Analyzer must collect the information for the LUOW Detail report in order to produce the Record Dumps report.

You can specify the DUMPREC keyword or the DUMPVERT keyword, but not both.

For more information, see “Using the Record Dumps report” on page 195.

**DUMPVERT keyword**

You can use the DUMPVERT keyword (or the DUMPV abbreviation) on the REPORTS control statement to control production of the Record Dumps report (in vertical dump format).

Log Analyzer writes this report to the SYSOUT-type LUOWDMPS data set. For more information about the report, see “Using the Record Dumps report” on page 195.

Specify the keyword as follows:

DUMPVERT=(limit,offset,length)
Specify one of the following required *limit* values:

- **ALL** or **999999999**
  Do not limit the number of lines that are written to the output report data set.

- **nnnnnnnnnn**
  Limit the number of output lines in the report to the specified number.

**Note**
The Record Dumps report can be large. You might want to limit the number of output records until you are certain that specified selection criteria are correct.

You can specify optional *offset* and *length* values to have Log Analyzer report only a portion of the record content.

- To indicate the starting position of the content to report, use the *offset* value. The offset is relative to 0 (if you specify 1 as the value, the starting position is the second byte of the record).

- To indicate the number of bytes of content to report, use the *length* value.

- If any portion of the target content (which is identified by the specified offset and length) falls beyond the end of the record, Log Analyzer ignores the specified offset and length.

If you omit the offset and length, Log Analyzer reports the entire record content.

No default value is provided. If you omit this keyword, Log Analyzer does not produce this report.

If you specify the DUMPVERT keyword, you must also specify the LUOWDETAIL keyword (see “LUOWDETAIL keyword” on page 123). Log Analyzer must collect the information for the LUOW Detail report in order to produce the Record Dumps report.

You can specify the DUMPVERT keyword or the DUMPREC keyword, but not both.

If you specify the DUMPVERT keyword, you can also specify the DUMPFORM keyword, as explained in “DUMPFORM keyword” on page 119.
LOGRECORDCODES keyword on the REPORTS control statement

You can use the LOGRECORDCODES keyword (or the LRC abbreviation) on the REPORTS control statement to report information about specified types of log records from within the set of log records that were included in LUOWs that Log Analyzer has created.

By using this keyword together with keywords on the FILTER control statement, you can select a large set of records to build LUOWs and then report on a smaller subset of LUOWs.

**Note**

You cannot specify a record type for display on reports if that record type is not included in the list of record types that Log Analyzer uses to determine whether a record is a candidate for the selection process. You can specify candidate record types by using the LOGRECORDCODES keyword on the FILTER control statement. In addition, Log Analyzer can include certain record types automatically.

To report information about specified types of log records in the set, you can specify one or more of the following values:

- Two-character log record type code (nn) or four-character log record type and subtype code (nnnn)

  Log Analyzer recognizes all standard log record types that are defined by IMS and most log record types that are defined by BMC. If you are interested in only a specified subtype of a log record type, you can specify the subtype code. If you omit the subtype code, the specification applies to all subtypes of the specified log record type. For information about supported log record types, see “IMS log record types” on page 345.

- Masked two-character log record type code (n*) or masked four-character log record type and subtype code (nnn*)

  You can use an asterisk (*) as a mask for the second character position of a log record code or for the fourth character position of a log record type and subtype code. Log Analyzer selects all log record codes that have the specified character in the masked position. For example, specify LOGRECORDCODES=3* to select record types 30, 31, 32, 33, and so on.

- Name of a PDS member that contains a predefined list of log record codes (memberName)

  Lists of log record codes are defined in members of the Log Analyzer log record code PDS. You specify the name of the PDS to Log Analyzer by using the LOGRECORDCODEPDS keyword on the ANALYZE control statement, as explained in “LOGRECORDCODEPDS keyword” on page 73 on page 81. For
more information about creating the log record codes PDS and defining lists of log record codes, see “Using a log record code PDS” on page 275.

If you want Log Analyzer to report information about all types of log records in the set, specify LOGRECORDCODES=ALL.

If you want Log Analyzer to report information about all types of log records except specified types, specify LOGRECORDCODES=ALL and then specify each type of log record to exclude by preceding the two-character log record code (or masked code) with the logical-not symbol (¬). This symbol is assigned to EBCDIC code X'5F' and is typed by pressing Shift - 6 on most keyboards. Shift - 6 is labeled as a circumflex (^) on some keyboards. You can also use the logical-not symbol in defined lists of log record codes.

The default is LOGRECORDCODES=ALL.

**LUOWDETAIL keyword**

You can use the LUOWDETAIL keyword (or the LDET abbreviation) on the REPORTS control statement to control production of the LUOW Detail report. Log Analyzer writes this report to the SYSOUT-type LUOWDET data set.

*Note*

The LUOW Detail report can be large. You might want to limit the number of output records until you are certain that specified selection criteria are correct.

If you specify the EXTRACT control statement but not the INDEXFILE control statement, you must specify the LUOWDETAIL keyword on the REPORTS control statement. Otherwise, Log Analyzer cannot populate the extract file.

Specify one of the following values:

- **ALL or 999999999**
  Do not limit the number of lines that are written to the output report data set.

- **nnnnnnnnnn**
  Limit the number of output lines in the report to the specified number.

- **(LUOWluowId,LUOWluowId,...)**
  Report detailed information for one or more specified LUOWs only. You identify an LUOW to include in the report by using its previously obtained LUOW ID (you can omit leading zeroes from the ID). To obtain the LUOW ID, run an analysis job that requests the LUOW Summary (LUOWSUMM) report, note the ID of the LUOW of interest, and then run the analysis job again with the
LUOWDETAIL=(LUOWluowid ...) keyword and without changing the selection criteria. (The selection criteria control LUOW creation and the ID that is assigned to each LUOW.)

No default value is provided. If you omit this keyword, Log Analyzer does not produce the LUOW Detail report.

If you specify the LUOWDETAIL keyword, you can also specify one of the following keywords or keyword combinations:

- LUOWDETAILDATA keyword (see “LUOWDETAILDATA keyword” on page 124), DUMPRREC keyword (see “DUMPRREC keyword” on page 119), or both
- DUMPVERT keyword (see “DUMPVERT keyword” on page 120)

Example
To produce the LUOW Detail report and display information for LUOWs 934 and 1265 only, specify the following keyword on the REPORTS control statement:

LUOWDETAIL=(LUOW934,LUOW1265)

For more information, see “Using the LUOW Detail report” on page 192.

LUOWDETAILDATA keyword

If you use the LUOWDETAIL keyword, you can also use the LUOWDETAILDATA keyword (or the LDETDATA abbreviation) on the REPORTS control statement to control whether Log Analyzer includes a portion of data from 01 records and 03 records in the LUOW Detail report.

This data is reported in a separate field labeled DATA ==> on the report and can help you to locate LUOWs of interest. The data is obtained from the first data segment that is encountered.

Specify the keyword as follows:

LUOWDETAILDATA=(offset,length)

To indicate the starting position of the data within the log record, use the offset value. The offset is relative to 0 (if you specify 1 as the value, the starting position is the second byte of the segment). To indicate the number (1 through 80) of bytes to include, use the length value.

Unprintable characters cannot be displayed in the report. For useful results, the data should contain at least some printable characters.
No default value is provided. If you omit the LUOWDETAILDATA keyword, the DATA field is not included in the report.

**Example**

To display the first 40 characters of 01 records and 03 records in the LUOW Detail report, specify the following keyword on the REPORTS control statement:

```
LUOWDETAILDATA=(0,40)
```

---

**LUOWSUMM keyword**

You can use the LUOWSUMM keyword (or the LSUMM abbreviation) on the REPORTS control statement to control production of the LUOW Summary report.

Log Analyzer writes this report to the SYSOUT-type LUOWSUMM data set.

Specify one of the following values:

- **ALL or 999999999**
  - Do not limit the number of lines that are written to the output report data set.

- **nnnnnnnnnn**
  - Limit the number of output lines in the report to the specified number.

No default value is provided. If you omit this keyword, Log Analyzer does not produce this report.

For more information, see “Using the LUOW Summary report” on page 188.

---

**LUOWTSEQ keyword**

You can use the LUOWTSEQ keyword (or the LTS abbreviation) on the REPORTS control statement to control production of the Time Sequence report.

Log Analyzer writes this report to the SYSOUT-type LUOWTSEQ data set.

Specify one of the following values:

- **ALL or 999999999**
  - Do not limit the number of lines that are written to the output report data set.
Limit the number of output lines in the report to the specified number.

**Note**
The Time Sequence report can be large. You might want to limit the number of output records until you are certain that specified selection criteria are correct.

No default value is provided. If you omit this keyword, Log Analyzer does not produce this report.

For more information, see “Using the Time Sequence report” on page 198.

**OFFSET keyword**

If you use the DUMPVERT keyword or the DUMPREC keyword, you can also use the OFFSET keyword on the REPORTS control statement to control the format of the location counter for hexadecimal data on the Record Dumps report.

Specify one of the following values:

- **HEX**
  
  Display the location counter in hexadecimal format, starting at relative location 0 and counting by multiples of four.

- **DECIMAL or DEC**
  
  Display the location counter in decimal format, starting at relative location 1 and counting by multiples of five.

The default is OFFSET=HEX.

**ORPHANS keyword**

You can use the ORPHANS keyword on the REPORTS control statement to control production of the Unresolved Orphans report.

Log Analyzer writes this report to the SYSOUT-type ORPHAN data set.

Specify one of the following values:

- **ALL or 999999999**
  
  Do not limit the number of lines that are written to the output report data set.
RBASTATS keyword

You can use the RBASTATS keyword on the REPORTS control statement to control production of the RBA Buffer Statistics report.

This report shows the distribution of updated relative byte addresses (RBAs) or block numbers in full-function IMS databases. (The report does not provide information about Fast Path databases.) Log Analyzer writes this report to the SYSOUT-type RBASTATS data set.

The RBA Buffer Statistics report is a single-record report. To produce it, Log Analyzer focuses exclusively on relevant records and does not create LUOWs. If you specify the RBASTATS keyword, the following specifications are invalid:

- Other keywords on the REPORTS control statement
- INDEXFILE control statement
- EXTRACT control statement

Although you can specify any SELECT subkeywords to narrow the focus of the report, specifying the UDBD subkeyword is the most helpful because Log Analyzer derives this report solely from type 5050 log records.

Specify one of the following values:

- **ALL**
  - Write detail and total fields to the output report data set.

- **TOTALS**
  - Write the total fields to the output report data set, and omit the detail fields.

No default value is provided. If you omit this keyword, Log Analyzer does not produce this report.

For more information, see “Using the RBA Buffer Statistics report” on page 209.
SUMMARY keyword

You can use the SUMMARY keyword (or the SUMM abbreviation) on the REPORTS control statement to control production of the Analysis Summary report.

Log Analyzer writes this report to the SYSOUT-type SUMMARY data set.

If you omit the REPORTS control statement and the EXTRACT control statement, Log Analyzer produces the Analysis Summary report by default.

_Btip_

When you approach a new problem to diagnose by using Log Analyzer, you might want to start by obtaining the Analysis Summary report only (without other reports, an extract data set, or an index data set). Log Analyzer can produce the Analysis Summary report quickly, with relatively low resource consumption, to let you know whether relevant records have been selected and whether the volume of data is small enough for interactive or manual analysis. If not, you can adjust control statement specifications and rerun the analysis job. When the Analysis Summary report shows the desired results, you can run the analysis job a final time to produce other reports and data sets.

BMC recommends that you do not limit the size of the Analysis Summary report.

Specify one of the following values:

- **ALL** or **999999999**
  Do not limit the number of records that are written to the output data set.

- **nnnnnnnnnn**
  Limit the number of output records to the specified number.

No default value is provided. If you omit this keyword (but you specify the REPORTS control statement with other keywords), Log Analyzer does not produce this report.

For more information, see “Using the Analysis Summary report” on page 178.

Specifying EXTRACT, INDEXFILE, and WORKFILE control statements

EXTRACT, INDEXFILE, and WORKFILE control statements control allocation of optional external data sets that are used for the following purposes:
The optional EXTRACT statement (or the EXTR abbreviation) allocates a Log Analyzer extract file. Log Analyzer writes all selected log records to this file. You can use this file as input to another execution of the Log Analyzer analysis job and for certain functions (such as viewing dumps of log records) during interactive analysis of LUOWs through the ISPF interface. If you omit the EXTRACT statement, Log Analyzer does not allocate an extract file and functions that require an extract file are unavailable. If you specify the EXTRACT statement but omit the INDEXFILE statement, you must specify the LUOWDETAIL keyword on the REPORTS control statement. Otherwise, Log Analyzer cannot populate the extract file.

For more information, see “Working with extract files” on page 296.

The optional INDEXFILE statement (or the INDEXF abbreviation) allocates a Log Analyzer index file. The Log Analyzer ISPF interface uses this file during interactive analysis of LUOWs through the ISPF interface. If you include the INDEXFILE statement, you must also include the EXTRACT statement. If you omit the INDEXFILE statement, Log Analyzer does not allocate an index file and interactive analysis is unavailable.

For more information, see “Working with index files” on page 300.

The WORKFILE statement (or the WORKF abbreviation) allocates a sort work file. Log Analyzer uses this file during the analysis job for sorting log records that are read from an input extract file. The WORKFILE statement is required if you specify the INPEXT keyword on the ANALYZE control statement.

For more information, see “Using extract files as input” on page 257.

EXTRACT, INDEXFILE, and WORKFILE control statements must follow the ANALYZE statement and precede the END control statement. However, you can specify them in any position relative to other control statements. Within a single command set, you can specify only one instance of each type of keyword.

### EXTRACT, INDEXFILE, and WORKFILE control statements

The following examples of EXTRACT, INDEXFILE, and WORKFILE control statements contain all valid keywords.

For each keyword, the figure shows a generic value. Some keywords are shown as comments because they typically are not used at the same time as other keywords.

#### Figure 23: EXTRACT, INDEXFILE, and WORKFILE control statement example

```plaintext
EXTRACT
  *  DATACLASS=className
  *  DCB=dataSetName
  DSN=dataSetName
  *  LIKE=dataSetName
  *  MGMTCLASS=className
  PRISPACE=nnnnnnnn
  SECSPACE=nnnnnnnn
```
For more examples, see “Working with analysis data” on page 291.

The following table summarizes information about the keywords that you can specify on EXTRACT, INDEXFILE, and WORKFILE control statements.

Table 12: EXTRACT, INDEXFILE, and WORKFILE control statement keywords

<table>
<thead>
<tr>
<th>Keyword (abbreviation)</th>
<th>Valid values</th>
<th>Default</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATACLASS (DATAC)</td>
<td>className</td>
<td>None</td>
<td>Specify the SMS data class to use for data set allocation on a system-managed storage device.</td>
<td>“DATACLASS keyword” on page 132</td>
</tr>
<tr>
<td>DCB</td>
<td>dataSetName</td>
<td>None</td>
<td>Specify the data set name of a data set to use as a model for allocation of a non-SMS data set.</td>
<td>“DCB keyword” on page 132</td>
</tr>
<tr>
<td>DSN</td>
<td>dataSetName</td>
<td>None</td>
<td>Specify the data set name to use for data set allocation. On the EXTRACT control statement, this keyword can specify a generation data group.</td>
<td>“DSN keyword” on page 132</td>
</tr>
<tr>
<td>Keyword (abbreviation)</td>
<td>Valid values</td>
<td>Default</td>
<td>Description</td>
<td>Topic</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------</td>
<td>---------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>LIKE</td>
<td>dataSetName</td>
<td>None</td>
<td>Specify the data set name of a data set to use as a model for allocation of an SMS data set.</td>
<td>“LIKE keyword” on page 133</td>
</tr>
<tr>
<td>MGMTCLASS (MGMTC)</td>
<td>className</td>
<td>None</td>
<td>Specify the SMS management class to use for data set allocation on a system-managed storage device.</td>
<td>“MGMTCLASS keyword” on page 133</td>
</tr>
<tr>
<td>PRISPACE (PRISP)</td>
<td>nnnnn</td>
<td>None</td>
<td>Specify the primary amount of space to use for data set allocation.</td>
<td>“PRISPACE keyword” on page 133</td>
</tr>
<tr>
<td>SECSPACE (SECSP)</td>
<td>nnnnn</td>
<td>None</td>
<td>Specify the secondary amount of space to use for data set allocation.</td>
<td>“SECSPACE keyword” on page 134</td>
</tr>
<tr>
<td>SPACEUNITS (SPU)</td>
<td>CYL, TRK</td>
<td>None</td>
<td>Specify whether to allocate space in cylinders or tracks.</td>
<td>“SPACEUNITS keyword” on page 134</td>
</tr>
<tr>
<td>STORCLASS (STORC)</td>
<td>className</td>
<td>None</td>
<td>Specify the SMS storage class to use for data set allocation on a system-managed storage device.</td>
<td>“STORCLASS keyword” on page 134</td>
</tr>
<tr>
<td>UNIT</td>
<td>unitName</td>
<td>None</td>
<td>Specify the unit type to use for data set allocation on a manually managed storage device. This keyword is not valid on the INDEXFILE statement.</td>
<td>“UNIT keyword” on page 134</td>
</tr>
<tr>
<td>VOLSER (VOLS)</td>
<td>volser</td>
<td>None</td>
<td>Specify the volume to use for data set allocation on a manually managed storage device.</td>
<td>“VOLSER keyword” on page 135</td>
</tr>
</tbody>
</table>
Requirements and valid values for most of these keywords depend on standards for storage usage that have been defined (by a storage administrator) in your environment.

**Note**
Ensure that the extract file is allocated to a storage pool that does not allow multi-volume data sets. Log Analyzer uses NOTE and POINT logic to access data in this file, and multi-volume data sets result in failures.

### DATACLASS keyword

You can use the DATACLASS keyword (or the DATAC abbreviation) on the EXTRACT, INDEXFILE, or WORKFILE control statement to specify the SMS data class to use for the data set to be allocated.

Specify a valid data class name as the value. No default value is provided.

### DCB keyword

You can use the DCB keyword on the EXTRACT control statement to specify a model data set control block (DSCB) for allocation of the data set as a generation data group (GDG).

Specify a valid DSCB name as the value. No default value is provided.

### DSN keyword

You can use the DSN keyword on the EXTRACT and INDEXFILE control statement to specify the name of the data set to be allocated.

**Note**
If the specified data set name does not exist in the system catalog, Log Analyzer allocates the data sets with the DISP=NEW parameter. If the data set name exists, Log Analyzer reuses the existing data set and overwrites the data; the data set is not reallocated. If you do not want to reuse existing data sets, you must rename them or specify data set names that are different from names of previously allocated data sets. To ensure unique data set names, you can implement simple naming conventions, such as including the current date in qualifiers.

Specify a valid data set name (1 through 44 characters) as the value. This data set name must not already exist in the system catalog.
On the EXTRACT statement, you can specify a GDG data set name:

\[ \text{dataSetName(+nn)} \]

**Note**
The data set name that you specify by using the DSN keyword is required for the following purposes when the Log Analyzer job is complete:

- You must enter the data set name of the index file on an ISPF panel when you want to analyze LUOWs interactively.
- You must specify the data set name of the extract file as the value of the INPEXT keyword when you use the file as input to another Log Analyzer job.

The DSN keyword is required on the EXTRACT statement and the INDEXFILE statement, and no default value is provided.

The work file is a temporary file that is used only for the duration of the analysis job. The DSN keyword is invalid on the WORKFILE control statement.

**LIKE keyword**

You can use the LIKE keyword on the EXTRACT control statement to specify a model for allocation of the data set as an SMS-managed data set.

Specify a valid data set name as the value. No default value is provided.

**MGMTCLASS keyword**

You can use the MGMTCLASS keyword (or the MGMTC abbreviation) on the EXTRACT, INDEXFILE, or WORKFILE control statement to specify the Storage Management Subsystem (SMS) management class to use for the data set to be allocated.

Specify a valid management class name as the value. No default value is provided.

**PRISPACE keyword**

You can use the PRISPACE keyword (or the PRISP abbreviation) on the EXTRACT, INDEXFILE, or WORKFILE control statement to specify the amount of primary space to use for the data set to be allocated.
Specify a number (1 through 16777215) as the value. No default value is provided.

**SECSPACE keyword**

You can use the SECSPACE keyword (or the SECSP abbreviation) on the EXTRACT, INDEXFILE, or WORKFILE control statement to specify the amount of primary space to use for the data set to be allocated.

Specify a number (1 through 16777215) as the value. No default value is provided.

**SPACEUNITS keyword**

You can use the SPACEUNITS keyword (or the SPU abbreviation) on the EXTRACT, INDEXFILE, or WORKFILE control statement to specify whether to allocate space for the data set in cylinders or tracks.

Specify one of the following values:

- **CYL**
  Allocate space in cylinders.

- **TRK**
  Allocate space in tracks.

No default value is provided.

**STORCLASS keyword**

You can use the STORCLASS keyword (or the STORC abbreviation) on the EXTRACT, INDEXFILE, or WORKFILE control statement to specify the SMS storage class to use for the data set to be allocated.

Specify a valid storage class name as the value. No default value is provided.

**UNIT keyword**

You can use the UNIT keyword on the EXTRACT or WORKFILE control statement to specify the unit type to use for the data set to be allocated.
This keyword is not valid on the INDEXFILE control statement.

Specify a valid unit name as the value. No default value is provided.

**VOLSER keyword**

You can use the VOLSER keyword (or the VOLS abbreviation) on the EXTRACT, INDEXFILE, or WORKFILE control statement to specify the volume serial number to use for the data set to be allocated.

*Note*

On the EXTRACT control statement, specify only a single value for the volume serial number.

Specify a valid volume serial number as the value. No default value is provided.

**Specifying END control statements**

Use the END control statement to indicate the end of the commands in the control statement data set.

The control statement data set can contain one END statement. This statement must be the last statement (other than comments) in the control statement data set. No keywords are valid on the END statement.

**Checking results of Log Analyzer jobs**

When you submit a Log Analyzer job for execution and the job is complete, you should check the results to ensure that the job has been completed successfully and has produced the output that you wanted.

The following table describes the condition codes that Log Analyzer can issue.

<table>
<thead>
<tr>
<th>Table 13: Log Analyzer condition codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition code</td>
</tr>
<tr>
<td>00</td>
</tr>
<tr>
<td>04</td>
</tr>
<tr>
<td>Condition code</td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>08</td>
</tr>
<tr>
<td>16</td>
</tr>
</tbody>
</table>

Log Analyzer writes error and informational messages to the SYSOUT data set. For an explanation of each message and the action that you should take in response, see the product messages manual or the ISPF interface. (For information about how to use the ISPF interface, see “Obtaining information about Log Analyzer messages” on page 330.)

For general information about solving problems that you might experience while you are working with Log Analyzer jobs, see “Troubleshooting analysis jobs” on page 331.

If Log Analyzer has produced batch reports, they follow the SYSOUT data set in the output data stream. If you have requested a batch report by specifying a keyword on the REPORTS control statement (or if you have omitted the REPORTS control statement and the Analysis Summary report is produced by default), Log Analyzer writes the report to a dynamically allocated data set. The ddname of this data set is the same as the keyword. For more information, see “Working with batch reports” on page 175.

If Log Analyzer has produced an extract file (because you have specified the EXTRACT control statement), you can examine it to ensure that the expected contents are present or use the file as input to another analysis job.

If Log Analyzer has produced an index file (because you have specified the INDEXFILE control statement) in addition to the extract file, you can analyze LUOWs through the Log Analyzer ISPF interface. For more information, see “Working with reports in the ISPF interface” on page 211.
Building Log Analyzer JCL through the ISPF interface

This chapter explains how to prepare a simple job to execute an analysis job by using the BMC Log Analyzer ISPF interface, submit the job for execution, and check the results of the execution.

Overview of building JCL in the ISPF interface

To analyze log records automatically and create LUOWs (if applicable), you must execute an analysis batch job.

You control the execution of the analysis job by using standard JCL and by using control statements that are specific to Log Analyzer. Control statements specify:

- Sources of log records to read
- Log records to select for analysis
- Reports and files to produce
- Options to use during processing

You can use the Log Analyzer ISPF interface to build the JCL from information that you specify on a series of panels (as explained in this chapter), or you can prepare the JCL manually (as explained in “Working with Log Analyzer jobs” on page 55).

To work with a Log Analyzer job through the ISPF interface, perform the following tasks:

**Note**  
Help information is available for each panel and field in the ISPF interface. For more information, see “Using online Help” on page 53.

1. Access the JCL functions of the ISPF interface.  
   For more information, see “Accessing JCL functions” on page 138.
2 Specify information that is used to build JOB, STEPLIB, and SYSIN data sets and control statements.

For more information, see the following topics:

- “Providing SYSIN, JOB, and STEPLIB information” on page 139
- “Building the ANALYZE control statement” on page 141
- “Building the FILTER control statement” on page 151
- “Building the INTERVAL control statement” on page 160
- “Building the REPORTS control statement” on page 162
- “Building EXTRACT, INDEXFILE, and WORKFILE control statements” on page 168

3 Submit the prepared job for execution, and check the results of the job.

For more information, see “Submitting Log Analyzer jobs and checking results” on page 171.

### Accessing JCL functions

Use the JCL Menu panel to access JCL functions, which include building JCL, generating JCL and viewing it in ISPF View, and generating and submitting JCL for execution.

#### To access JCL functions

1 Access the Log Analyzer ISPF interface.

   For instructions, see “Accessing the ISPF interface” on page 46.

2 On the Log Analyzer Main Menu, enter 1 in the Log Analyzer option field.

   The JCL Menu panel is displayed.

#### Figure 24: JCL Menu panel

<table>
<thead>
<tr>
<th>Menu Options Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer      JCL Menu</td>
</tr>
</tbody>
</table>

  JCL actions.

  1. Build - Specify JCL control cards and SYSIN
  2. View - View generated JCL
  3. Submit - Submit generated JCL
3 In the **JCL Actions** field, enter one of the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The Build JCL panel is displayed. Use this panel (and dependent panels) to provide values that are used to build JCL for executing an analysis job. For more information, see “Providing SYSIN, JOB, and STEPLIB information” on page 139.</td>
</tr>
</tbody>
</table>
| 2     | The ISPF interface performs the following actions:  
  - Uses profile values that have been specified through Build JCL panels to generate JCL for executing an analysis job  
  - Displays the generated JCL in a temporary data set in an ISPF View session  
  After you view the JCL, you can submit it for execution. Changes you make to the displayed JCL do not change your profile values. For more information, see “Generating and viewing Log Analyzer JCL” on page 171. |
| 3     | The ISPF interface generates JCL from your profile values and submits the JCL for execution. For more information, see “Generating and submitting Log Analyzer jobs” on page 172. |

---

**Providing SYSIN, JOB, and STEPLIB information**

To provide values that the ISPF interface uses to generate SYSIN, JOB, and STEPLIB statements for the analysis job, and to select control statements to include in the SYSIN control statement data set, use the Build JCL panel and dependent panels.

**To access and work with the Build JCL panel**

1 Access the JCL Menu panel.  
   For instructions, see “Accessing JCL functions” on page 138.

2 In the **JCL Actions** field, enter 1.  
   The Build JCL panel is displayed.

**Figure 25: Build JCL panel**

```
File   Edit   Options   Help
Log Analyzer Build JCL
Command ➔
```
Control statements. Type an action code. S=Update, /=Include, blank=Exclude
Note: A / indicates the control statement will be included in the generated JCL

/ ANALYZE   - Specify which logs to read
/ FILTER    - Specify which LUOWs to keep
/ INTERVAL  - Specify the time frames within the logs
/ REPORTS   - Specify which reports to produce
/ EXTRACT   - Specify the output SLDS extract file
/ INDEXFILE - Specify the output report index file
/ WORKFILE  - Specify the sort work file allocation information

Job card.
//PAGTEST JOB (5711),'LOG ANALYZER',CLASS=Q,MSGCLASS=X,TIME=1440,
//  NOTIFY=&SYSUID

________________________________________________________________________

STEPLIB libraries.
'BMCLUI.LOADLIB'
'BMCLUI.BBLINK'
'BMCLUI.ICOLIB'

F1=Help    F3=Exit    F7=Up      F8=Down   F12=Cancel

3 In the Control statements fields to the left of each type of control statement, enter one of the following action codes:

<table>
<thead>
<tr>
<th>Code</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>The ISPF interface displays a panel that you can use to view and change keywords and values for the selected control statement. The panel that is displayed depends on the control statement that you select:</td>
</tr>
<tr>
<td></td>
<td>▪ ANALYZE (See “Building the ANALYZE control statement” on page 141.)</td>
</tr>
<tr>
<td></td>
<td>▪ FILTER (See “Building the FILTER control statement” on page 151.)</td>
</tr>
<tr>
<td></td>
<td>▪ INTERVAL (See “Building the INTERVAL control statement” on page 160.)</td>
</tr>
<tr>
<td></td>
<td>▪ REPORTS (See “Building the REPORTS control statement” on page 162.)</td>
</tr>
<tr>
<td></td>
<td>▪ EXTRACT (See “Building EXTRACT, INDEXFILE, and WORKFILE control statements” on page 168.)</td>
</tr>
<tr>
<td></td>
<td>▪ INDEXFILE (See “Building EXTRACT, INDEXFILE, and WORKFILE control statements” on page 168.)</td>
</tr>
<tr>
<td></td>
<td>▪ WORKFILE (See “Building EXTRACT, INDEXFILE, and WORKFILE control statements” on page 168.)</td>
</tr>
<tr>
<td>/</td>
<td>The ISPF interface includes the control statement when it generates the JCL for executing the analysis job.</td>
</tr>
<tr>
<td>blank</td>
<td>The ISPF interface omits the control statement from the generated JCL</td>
</tr>
</tbody>
</table>
When you return to the Build JCL panel after you have selected a control statement with S, the field for that control statement is set to / (so that the statement is included in the generated JCL).

The ANALYZE control statement is required in the generated JCL.

4 In the Job card field, enter the JOB statement to use for executing the analysis job.

The JOB statement that you specify in this field is included in the generated JCL exactly as you specify it. It must conform to JOB statement standards in your environment.

5 In the STEPLIB libraries field, enter the data set name of each data set to include in the STEPLIB concatenation.

For more information about the libraries to include, see “STEPLIB DD statement” on page 57.

Building the ANALYZE control statement

To generate keywords and values on the ANALYZE control statement, use the ANALYZE Control Statement panel and dependent panels.

The ANALYZE control statement specifies log record sources to use as input and general options to use for the analysis job. For more information, see “Specifying ANALYZE control statements” on page 63.

To access and work with the ANALYZE Control Statement panel

1 Access the Build JCL panel.

For instructions, see “Accessing JCL functions” on page 138.

2 In the Control Statements field to the left of the ANALYZE label, enter S.

The ANALYZE Control Statement panel is displayed.

Figure 26: ANALYZE Control Statement panel
Building the ANALYZE control statement

In the **Specify input** fields to the left of each type of log record source, enter one of the following action codes:

<table>
<thead>
<tr>
<th>Code</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>The ISPF interface displays a panel that you can use to view and change keywords and values that specify log record sources to use as input to the analysis job. The panel that is displayed depends on the type of input that you select:</td>
</tr>
<tr>
<td></td>
<td>■ RECON</td>
</tr>
<tr>
<td></td>
<td>See “Building keywords for RECON input” on page 143.</td>
</tr>
<tr>
<td></td>
<td>■ SLDS</td>
</tr>
<tr>
<td></td>
<td>See “Building keywords for SLDS input” on page 144.</td>
</tr>
<tr>
<td></td>
<td>■ Extract file</td>
</tr>
<tr>
<td></td>
<td>See “Building keywords for extract file input” on page 146.</td>
</tr>
<tr>
<td></td>
<td>■ Active OLDS</td>
</tr>
<tr>
<td></td>
<td>Select this field to cause the ISPF interface to generate the ACTIVEOLDS=YES keyword on the ANALYZE control statement. For more information, see “ACTIVEOLDS keyword” on page 67.</td>
</tr>
<tr>
<td></td>
<td>■ IMS Connect</td>
</tr>
<tr>
<td></td>
<td>See “Building keywords for IMS Connect input” on page 146.</td>
</tr>
<tr>
<td></td>
<td>■ DB2 log</td>
</tr>
<tr>
<td></td>
<td>See “Building keywords for DB2 log input” on page 148.</td>
</tr>
<tr>
<td></td>
<td>■ DB2 BSDS</td>
</tr>
<tr>
<td></td>
<td>See “Building keywords for DB2 bootstrap data set input” on page 149.</td>
</tr>
<tr>
<td></td>
<td>■ MQ extract</td>
</tr>
<tr>
<td></td>
<td>See “Building keywords for MQ extract data set input” on page 150.</td>
</tr>
<tr>
<td>blank</td>
<td>The ISPF interface omits the input type from the generated JCL</td>
</tr>
<tr>
<td>/</td>
<td>The ISPF interface includes the selected type of input when it generates the JCL for executing the analysis job.</td>
</tr>
</tbody>
</table>
When you return to the ANALYZE Control Statement panel after you have entered S in a Specify input field, the field is set to / (so that the keywords are included in the generated JCL).

At least one log record source must be included in the generated JCL.

4 (optional) In the Time zone field, enter a value that specifies how to interpret input time values and display output time values.

The ISPF interface uses this field to generate the TIMEZONE keyword. For more information, see “TIMEZONE keyword” on page 78.

5 (optional) In the Log record code PDS field, enter the data set name of your log record code PDS.

The ISPF interface uses this field to generate the LOGRECORDCODEPDS keyword. For more information, see “LOGRECORDCODEPDS keyword” on page 73.

6 (optional) In the Suppress elapsed time calculations field, enter a slash or S if you do not want Log Analyzer to report elapsed time values.

You should specify this value if input log records have been produced on systems that are not using the same sysplex timer. The ISPF interface uses this field to generate the ELAPSEDTIME=NO keyword. For more information, see “ELAPSEDTIME keyword” on page 69.

Building keywords for RECON input

To generate RECON-related keywords and values on the ANALYZE control statement, use the RECON Input panel. RECON-related keywords control how Log Analyzer allocates input SLDSs by using information that is recorded in RECON data sets.

To access and work with the RECON Input panel

1 Access the ANALYZE Control Statement panel.

For instructions, see “Building the ANALYZE control statement” on page 141.

2 In the Specify input field to the left of the RECON label, enter S.

The RECON Input panel is displayed.
3 (optional) In the IMSIDs field, enter the IMSID of each IMS system that created log records to be analyzed.

The ISPF interface uses this field to generate the IMSID keyword. For more information, see “IMSID keyword” on page 70.

4 (optional) In the Maximum logs options field, enter values that control the number of log data sets that Log Analyzer tries to process.

The ISPF interface uses this field to generate the MAXLOGS keyword. For more information, see “MAXLOGS keyword” on page 74.

5 (optional) In the RECON n field, enter the data set name of a RECON data set to examine during the analysis job.

The ISPF interface uses this field to generate the RECON keyword. For more information, see “RECON keyword” on page 76.

**Building keywords for SLDS input**

To generate SLDS-related keywords and values on the ANALYZE control statement, use the SLDS Input panel. SLDS-related keywords are used for explicit specification of input log data sets to be allocated.

**To access and work with the SLDS Input panel**

1 Access the ANALYZE Control Statement panel.

For instructions, see “Building the ANALYZE control statement” on page 141.
2 In the Specify input field to the left of the SLDS label, enter S.

The SLDS Input panel is displayed:

**Figure 28: SLDS Input panel**

Specify the SLDS data sets to use as input.

<table>
<thead>
<tr>
<th>SLDS data set name</th>
<th>IMS Release (7.1, 8.1, 9.1, 10.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>'LUI.LOG.MSGSWI.BMCIRM.QA.ZFIE0.IR8V.SLDSP05'</td>
<td>8.1</td>
</tr>
</tbody>
</table>

---

3 *(optional)* In the Log record limit field, enter a limit on the number of log records to process from an input log data set.

The ISPF interface uses this field to generate the LIMIT keyword. For more information, see “LIMIT keyword” on page 73.

4 *(optional)* In the Skip to log record field, enter a number of log records in an input log data set to skip before Log Analyzer begins to process records.

The ISPF interface uses this field to generate the SKIP keyword. For more information, see “SKIP keyword” on page 76.

5 *(optional)* In the SLDS data set name field, enter the data set name of a log data set to examine during the analysis job. Also enter the release number (such as 9.1 or 10.1) of the IMS system that created the log data set.

If a data set name prefix (such as your TSO user ID) is set in your TSO profile and you do not want this prefix to be added to the data set name that you enter, enclose the data set name in single quotes.

The ISPF interface uses this field to generate the SLDS keyword. For more information, see “SLDS keyword” on page 77.
Building keywords for extract file input

To generate INPEXT keywords and values on the ANALYZE control statement, use the Extract File Input panel. These keywords are used to allocate extract files that were created during a previous execution of the analysis job.

To access and work with the Extract File Input panel

1. Access the ANALYZE Control Statement panel.

   For instructions, see “Building the ANALYZE control statement” on page 141.

2. In the Specify input field to the left of the Extract File label, enter S.

   The Extract File Input panel is displayed:

   Figure 29: Extract File Input panel

   File Options Help
   Log Analyzer Extract File Input
   Command ===> ____________________________
   Specify the Log Analyzer extract files to use as input.
   Extract file ____________________________
   Extract file ____________________________
   Extract file ____________________________
   Extract file ____________________________
   Note: If you specify an extract file as input, you must also specify the WORKFILE control statement.

3. (optional) In the Extract file field, identify an input extract data set that was created during a previous execution of Log Analyzer.

   The ISPF interface uses this field to generate the INPEXT keyword value. For more information, see “INPEXT keyword” on page 70.

Building keywords for IMS Connect input

To generate IPRJRNRL or IPRPREFIX keywords and values on the ANALYZE control statement, use the IMS Connect Input panel.
These keywords allocate Energizer for IMS Connect journal files for Log Analyzer to use as input. The journal files contain information about IMS Connect events such as Read Socket, Send to IMS, Send to Client, and IMS Send to IMS Connect.

To access and work with the IMS Connect Input panel

1 Access the ANALYZE Control Statement panel.
   For instructions, see “Building the ANALYZE control statement” on page 141.

2 In the Specify Input field to the left of the IMS Connect label, enter S.
   The IMS Connect Input panel is displayed:

   Figure 30: IMS Connect Input panel

3 (optional) In the Type of Energizer for IMS Connect journal data sets to use as input field, specify one of the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Dynamic) Log Analyzer automatically identifies Energizer journal data sets to select, based on the value in the Data set name prefix field. This choice causes the Log Analyzer ISPF interface to build JCL that contains the IPRPREFIX keyword.</td>
</tr>
<tr>
<td>2</td>
<td>(Specific) Log Analyzer uses the Energizer journal data sets that you specify in the Data set names field. This choice causes the Log Analyzer ISPF interface to build JCL that contains the IPRJRNL keyword.</td>
</tr>
</tbody>
</table>
4 If you specified 1 in the **Type of Energizer for IMS Connect journal data sets to use as input** field, specify a value in the **Data set name prefix** field.

This value is the prefix that Log Analyzer uses to automatically determine which Energizer for IMS Connect journal data sets to include in the analysis job. This value should match the value that was specified in the Energizer for IMS Connect options. For more information, see “IPRPREFIX keyword” on page 72.

5 If you specified 2 in the **Type of Energizer for IMS Connect journal data sets to use as input** field, specify one or more values in the **Data set names** field.

Each value is the data set name of an Energizer for IMS Connect journal data set that Energizer uses to track IMS Connect events. For more information, see “IPRJRNL keyword” on page 71.

### Building keywords for DB2 log input

To generate DB2LOG keywords and values on the ANALYZE control statement, use the DB2 Input panel. These keywords allocate DB2 log data sets for Log Analyzer to use as input.

**To access and work with the DB2 Input panel**

1 Access the ANALYZE Control Statement panel.

   For instructions, see “Building the ANALYZE control statement” on page 141.

2 In the **Specify input** field to the left of the **DB2 log** label, enter **S**.

   The DB2 Input panel is displayed:

   **Figure 31: DB2 Input panel**

   ```markdown
   File  Edit  Options  Help
   Log Analyzer                        DB2 Input
   Command ===> _________________________________________________________________
   Specify the DB2 log data set names to use as input.
   'PAG.TEST.DB2.LOG'
   ```
3 (optional) In the data set names area, identify one or more DB2 log data sets.

The ISPF interface uses this field to generate the DB2LOG keyword value. For more information, see “DB2LOG keyword” on page 68.

Building keywords for DB2 bootstrap data set input

To generate DB2BSDS keywords and values on the ANALYZE control statement, use the DB2 BSDS Input panel. These keywords specify DB2 bootstrap data sets (BSDSs) for Log Analyzer to use to allocate DB2 logs implicitly.

To access and work with the DB2 BSDS Input panel

1 Access the ANALYZE Control Statement panel.

   For instructions, see “Building the ANALYZE control statement” on page 141.

2 In the Specify input field to the left of the DB2 BSDS label, enter S.

   The DB2 Input panel is displayed:

   **Figure 32: DB2 BSDS Input panel**

   ```
   File  Edit  Options  Help
   Log Analyzer                    DB2 BSDS Input
   Command ==> _________________________________________________________________
   Specify the DB2 bootstrap data set names to use as input.
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   _______________________________________________________________
   F1=Help    F3=Exit   F12=Cancel
   ```

3 (optional) In the data set names area, identify one or more DB2 BSDSs.
Building the ANALYZE control statement

The ISPF interface uses this field to generate the DB2BSDS keyword value. For more information, see “DB2BSDS keyword” on page 68.

Building keywords for MQ extract data set input

To generate MQEXT keywords and values on the ANALYZE control statement, use the MQ Extract Input panel. These keywords allocate WebSphere for MQ extract data sets for Log Analyzer to use as input.

To access and work with the MQ Extract Input panel

1. Access the ANALYZE Control Statement panel.

   For instructions, see “Building the ANALYZE control statement” on page 141.

2. In the Specify input field to the left of the MQ extract label, enter S.

   The MQ Extract Input panel is displayed:

   **Figure 33: MQ Extract Input panel**

   File Edit Options Help

   Log Analyzer MQ Extract Input

   Command ===> _________________________________________________________________

   Specify the WebSphere MQ extract data set names to use as input.

   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________
   ________________________________________________

   F1=Help F3=Exit F12=Cancel

3. *(optional)* In the data set names area, identify one or more MQ extract data sets.

   The ISPF interface uses this field to generate the MQEXT keyword value. For more information, see “MQEXT keyword” on page 75.
Building the FILTER control statement

To generate keywords and values on the FILTER control statement, use the FILTER Control Statement panel and dependent panels.

The FILTER control statement specifies types of log records that are candidates for the record selection process and filter criteria for selecting log records and LUOWs of interest. For more information, see “Specifying FILTER control statements” on page 83.

**Note**
You cannot specify the FILTER control statement if you specify certain keywords on the REPORTS control statement. For more information, see “Building the REPORTS control statement” on page 162.

To access and work with the FILTER Control Statement panel

1. Access the Build JCL panel.
   
   For instructions, see “Accessing JCL functions” on page 138.

2. In the **Control Statements** field to the left of the **FILTER** label, enter **S**.

   The FILTER Control Statement panel is displayed:

   **Figure 34: FILTER Control Statement panel**

<table>
<thead>
<tr>
<th>File</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>FILTER Control Statement</td>
<td>Command ==&gt;</td>
</tr>
</tbody>
</table>

   FILTER option. Select the type of FILTER keywords to update.
   - 1. Log record codes - Specify which log codes to include or exclude
   - 2. SELECT subkeywords - Specify destinations, userids, content, etc.

3. In the **FILTER option** field, enter one of the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The FILTER - Log Record Codes panel is displayed. You use this panel (and dependent panels) to specify types of log records that are candidates for the selection process. For more information, see “Specifying log record codes” on page 152.</td>
</tr>
<tr>
<td>2</td>
<td>The FILTER - SELECT Subkeywords panel is displayed. You use this panel (and dependent panels) to specify filter criteria for selecting log records and LUOWs of interest. For more information, see “Specifying filter criteria” on page 155.</td>
</tr>
</tbody>
</table>
Specifying log record codes

To generate the LOGRECORDCODES keyword and values on the FILTER control statement, use the FILTER - Log Record Code panel.

The LOGRECORDCODES keyword specifies how to handle different types of log records when identifying candidates for the log record selection process. For more information, see “LOGRECORDCODES keyword on the FILTER control statement” on page 85.

To access and work with the FILTER - Log Record Codes panel

1. Access the FILTER Control Statement panel.
   For instructions, see “Building the FILTER control statement” on page 151.

2. In the FILTER option field, enter 1.
   The FILTER - Log Record Codes panel is displayed.

Figure 35: FILTER - Log Record Codes panel

To clear all values from all fields on this panel, enter the CLEAR command.
Log Analyzer uses the default LOGRECORDCODES=ALL on the FILTER control statement only if all fields on this panel are empty.
If you want Log Analyzer to identify log record types that are relevant for your search criteria automatically, leave all fields on this panel empty and specify one or more SELECT subkeywords.
3 In the **Included log codes** field, enter one or more two-character or four-character log record codes that are associated with types and subtypes of log records to include as candidates for the log record selection process.

Separate each log code from another log code with a comma or a blank.

To enter log codes by selecting them from a list, place the cursor anywhere in the field and enter the PROMPT command (F4). The Select Log Codes pop-up window is displayed. For more information, see “Selecting log codes from a prompt” on page 154.

For a list of log record codes that Log Analyzer has mapped, see “IMS log record types” on page 345.

4 In the **Excluded log codes** field, enter one or more two-character or four-character log record codes that are associated with types and subtypes of log records to exclude as candidates for the log record selection process.

**Note**

A log code entry in this field makes sense only if the log code is included by one of the following methods:

- You enter a value in the **Included log codes** field.
- You leave the **Included log codes** field and the **PDS member names** field blank.
- The log code is included in a member of the log record code PDS and the name of this member is entered in the **PDS member name** field.

Separate each log code from another log code with a comma or a blank.

To enter log codes by selecting them from a list, place the cursor anywhere in the field and enter the PROMPT command (F4). The Select Log Codes pop-up window is displayed. For more information, see “Selecting log codes from a prompt” on page 154.

The ISPF interface generates the LOGRECORDCODES keyword with a logical-not sign (¬) preceding each code that you have specified.

For a list of log record codes that Log Analyzer has mapped, see “IMS log record types” on page 345.

5 In the **PDS member names** field, enter one or more names of members of the log record codes PDS.

Each member of the PDS lists log records to include or exclude as candidates for the log record selection process.
The log record codes PDS is identified by using the LOGRECORDCODEPDS keyword on the ANALYZE control statement, which is generated from the Log record code PDS field on the ANALYZE Control statement panel. For more information, see “Building the ANALYZE control statement” on page 141.

For more information about the LOGRECORDCODEPDS keyword, see “LOGRECORDCODEPDS keyword” on page 73.

Selecting log codes from a prompt

To enter log record codes by selecting them from a list, use the Select Log Code pop-up window.

To access and work with the Select Log Codes pop-up window

1. Access the FILTER - Log Record Codes panel.

For instructions, see “Building the FILTER control statement” on page 151.

2. Place the cursor in the Included log codes field or the Excluded log codes field, and enter the PROMPT command (which is assigned to F4).

The Select Log Codes pop-up window is displayed:

Figure 36: Select Log Codes pop-up window

Select log codes to be included in the log record filter. Then press Enter.

Commands: CLEAR, ALL

<table>
<thead>
<tr>
<th>Command</th>
<th>Select Log Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 MSGIN</td>
<td>02 CMMI</td>
</tr>
<tr>
<td>08 APPLC</td>
<td>09 BSTAT</td>
</tr>
<tr>
<td>12 CONVE</td>
<td>13 CONVC</td>
</tr>
<tr>
<td>20 DBOPN</td>
<td>21 DBCLS</td>
</tr>
<tr>
<td>27 XTEND</td>
<td>28 PH1DC</td>
</tr>
<tr>
<td>34 CANCEL</td>
<td>35 MGENQ</td>
</tr>
<tr>
<td>3A OFIXF</td>
<td>3B OFIXR</td>
</tr>
<tr>
<td>40 CHKPT</td>
<td>41 BATCH</td>
</tr>
<tr>
<td>48 PAD</td>
<td>4C DBPSB</td>
</tr>
<tr>
<td>57 DBUR</td>
<td>59 FP</td>
</tr>
<tr>
<td>66 3600</td>
<td>67 TRACE</td>
</tr>
<tr>
<td>70 OLC</td>
<td>72 USER</td>
</tr>
<tr>
<td>97 MVF9</td>
<td>9F MVFA</td>
</tr>
</tbody>
</table>

F1=Help   F3=Exit   F12=Cancel

3. Select log record codes as follows:

- To select a log code, enter any character in the selection field to the left of the log code.

The cursor advances to the next log code field so that you can easily select a range of codes by typing a character repeatedly.
- To select all fields on this panel, enter the ALL command.
- To clear all fields on this panel, enter the CLEAR command.

For more information about the log record codes that Log Analyzer has mapped, see “IMS log record types” on page 345.

4 Press Enter.

The FILTER - Log Record Codes panel is displayed. The selected log record codes are entered in the field from which you entered the PROMPT command.

**Specifying filter criteria**

To generate SELECT-related keywords and values on the FILTER control statement, use the FILTER - Select Subkeywords panel. SELECT-related keywords specify filter criteria that identify log records and LUOWs of interest.

**To access and work with the FILTER - SELECT Subkeywords panel**

1 Access the FILTER Control Statement panel.

For instructions, see “Building the FILTER control statement” on page 151.

2 In the FILTER option field, enter 2.

The FILTER - SELECT Subkeywords panel is displayed:

**Figure 37: FILTER - SELECT Subkeywords panel**

Specify SELECT subkeywords to indicate which LUOWs to keep. Use the PROMPT key to display a selection list of valid subkeywords.

3 In the SELECT subkeywords field, enter filter criteria that identify log records and LUOWs of interest.

To enter filter criteria manually, enter SELECT subkeywords, values, logical operators, and Boolean operators according to the rules that are described in “SELECT keyword” on page 89.
To enter subkeywords by selecting them from a list, place the cursor anywhere in the field and enter the PROMPT command (which is assigned to F4). The Insert SELECT Subkeyword pop-up window is displayed. For more information, see “Inserting SELECT subkeywords from a prompt” on page 156.

The ISPF interface uses this field to generate the SELECT keyword. For more information, see “SELECT keyword” on page 89.

4 In the CONTENT option field, enter one of the following values to specify how to filter content in 01 and 03 log records:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Search for specified content in the user data portion of the log record.</td>
</tr>
<tr>
<td>2</td>
<td>Search for specified content in all portions of the log record.</td>
</tr>
</tbody>
</table>

The ISPF interface uses this field to generate the CONTENT0103 keyword. For more information, see “CONTENT0103 keyword” on page 84.

**Inserting SELECT subkeywords from a prompt**

To insert one of the following elements by selecting it from a list, use the Insert SELECT Subkeyword pop-up window:
- SELECT subkeyword
- Relational operator
- Value
- Boolean operator (if another filter criterion has been specified)

Together, these elements specify a filter that identifies log records and LUOWs of interest.

**To access and work with the Insert SELECT Subkeyword pop-up window**

1 Access the FILTER - SELECT Subkeywords panel.

   For instructions, see “Specifying filter criteria” on page 155.

2 Place the cursor anywhere in the SELECT subkeywords field and enter the PROMPT command (which is assigned to F4).
The Insert SELECT Subkeyword pop-up window is displayed:

**Figure 38: Insert SELECT Subkeyword pop-up window**

<table>
<thead>
<tr>
<th>Boolean operator. (Optional)</th>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>And</td>
<td>Insert the AND Boolean operator between the criteria. For the log record to be selected, both criteria must be true.</td>
</tr>
<tr>
<td>2</td>
<td>Or</td>
<td>Insert the OR Boolean operator between the criteria. For the log record to be selected, either criterion must be true.</td>
</tr>
</tbody>
</table>

If you are inserting a second or subsequent criterion, enter one of the following values in the **Boolean operator** field:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Insert the CONTENT subkeyword, which specifies a search string, log record type, and optional offset and length to match in log records. The Insert SELECT CONTENT Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>2</td>
<td>Insert the DBD subkeyword, which specifies the name of a DBD and an optional data set ID or area name (and, if you specify an area name, a relative byte address) to match in log records. The Insert SELECT DBD Subkeyword pop-up window is displayed.</td>
</tr>
</tbody>
</table>

In the **Select a SELECT subkeyword to insert** field, enter one of the following values to choose the subkeyword to be inserted. For more information about these subkeywords, see “SELECT subkeywords” on page 91.
<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Insert the RDBD subkeyword, which specifies the name of a read-only DBD to match in log records. The Insert SELECT RDBD Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>4</td>
<td>Insert the UDBD subkeyword, which specifies the name of an update DBD and an optional data set ID or area name (and, if you specify an area name, a relative byte address) to match in log records. The Insert SELECT UDBD Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>5</td>
<td>Insert the PSB subkeyword, which specifies the name of a PSB to match in log records. The Insert SELECT PSB Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>6</td>
<td>Insert the DEST subkeyword, which specifies the name of a transaction destination to match in log records. The Insert SELECT DEST Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>7</td>
<td>Insert the ORIGIN subkeyword, which specifies the name of a transaction origin (such as an LTERM) to match in log records. The Insert SELECT ORIGIN Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>8</td>
<td>Insert the USERID subkeyword, which specifies the name of a user ID to match in log records. The Insert SELECT USERID Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>9</td>
<td>Insert the PORT# subkeyword, which specifies the name of an OTMA port number to match in log records. The Insert SELECT PORT# Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>10</td>
<td>Insert the TPIPE subkeyword, which specifies the name of an OTMA transaction pipe to match in log records. The Insert SELECT TPIPE Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>11</td>
<td>Insert the TMEMBER subkeyword, which specifies the name of an OTMA transaction member to match in log records. The Insert SELECT TMEMBER Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>12</td>
<td>Insert the LUNAME subkeyword, which specifies the name of an APPC logical unit to match in log records. The Insert SELECT LUNAME Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>13</td>
<td>Insert the TPNAME subkeyword, which specifies the name of an APPC transaction program to match in log records. The Insert SELECT TPNAME Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>14</td>
<td>Insert the ABENDS subkeyword, which specifies whether to search for units of work that have terminated abnormally. The Insert SELECT ABENDS Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>15</td>
<td>Insert the NODE subkeyword, which specifies the name of a node to match in log records. The Insert SELECT NODE Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>16</td>
<td>Insert the LTERM subkeyword, which specifies the name of a logical terminal to match in log records. The Insert SELECT LTERM Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>Value</td>
<td>Result</td>
</tr>
<tr>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>17</td>
<td>Insert the TRAN subkeyword, which specifies the name of a transaction to match in log records. The Insert SELECT TRAN Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>18</td>
<td>Insert the CICSTRAN subkeyword, which specifies the name of a CICS transaction to match in log records. The Insert SELECT CICS TRAN Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>19</td>
<td>Insert the ETIME subkeyword, which specifies whether to filter logical units of work by elapsed time. The Insert SELECT ETIME Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>20</td>
<td>Insert the PST subkeyword, which specifies the numeric identifier of the partition specification table (PST) of the IMS region that is associated with transactions of interest. The Insert SELECT PST Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>21</td>
<td>Insert the MPR subkeyword, which specifies the job name of the IMS message processing region (MPR) that is associated with transactions of interest. The Insert SELECT MPR Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>22</td>
<td>Insert the APPC_LTERM subkeyword, which specifies the APPC logical terminal name that is associated with transactions of interest. The Insert SELECT APPC_LTERM Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>23</td>
<td>Insert the CLIENTID subkeyword, which specifies the identifier of the client that is associated with transactions of interest. The Insert SELECT CLIENTID Subkeyword pop-up window is displayed.</td>
</tr>
<tr>
<td>24</td>
<td>Insert the OIMS subkeyword, which specifies the identifier of an originating IMS system that is associated with transactions of interest. The Insert SELECT OIMS Subkeyword pop-up window is displayed.</td>
</tr>
</tbody>
</table>

5 In the Relational Operator field on the Insert SELECT type Subkeyword pop-up window for all subkeywords except the ETIME subkeyword, enter one of the following values to choose the relational operator to be inserted:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Insert = (an equal sign) between the keyword and the value. For the comparison of the filter criterion with a log record to result in a True condition, the value in the record must match the subkeyword value.</td>
</tr>
<tr>
<td>2</td>
<td>Insert NOT= (a literal NOT followed by an equal sign) between the keyword and the value. For the comparison of the filter criterion with a log record to result in a True condition, the value in the record must not match the subkeyword value.</td>
</tr>
</tbody>
</table>

For the ETIME subkeyword, enter one of the following values to choose the relational operator to be inserted:
1. Insert \(>\) (a greater-than sign) between the keyword and the value. For the comparison of the filter criterion with an LUOW to result in a True condition, the value for the LUOW must be greater than the subkeyword value.

2. Insert \(\geq\) (a greater-than sign and an equal sign) between the keyword and the value. For the comparison of the filter criterion with an LUOW to result in a True condition, the value for the LUOW must be greater than or equal to the subkeyword value.

3. Insert \(<\) (a less-than sign) between the keyword and the value. For the comparison of the filter criterion with an LUOW to result in a True condition, the value for the LUOW must be less than the subkeyword value.

4. Insert \(\leq\) (a less-than sign and an equal sign) between the keyword and the value. For the comparison of the filter criterion with an LUOW to result in a True condition, the value for the LUOW must be less than or equal to the subkeyword value.

6. Fill in data-entry fields and selection fields (as applicable) on the Insert SELECT type Subkeyword pop-up window.

   The fields that are displayed and the valid values that you can enter depend on the SELECT subkeyword that you are specifying. For more information, see “SELECT subkeywords” on page 91.

7. Press Enter.

   The Boolean operator (if applicable), the subkeyword, the relational operator, and the subkeyword value (or values) are entered in the SELECT subkeywords field on the FILTER - SELECT Subkeywords panel.

8. As needed, continue to insert SELECT subkeywords by placing the cursor in the SELECT subkeywords field at the position where you want the subkeyword to be entered and entering the PROMPT command (F4).

**Building the INTERVAL control statement**

To build keywords and values on the INTERVAL control statement, use the INTERVAL Control Statement panel.

You can use the INTERVAL statement for the following purposes:

- Specify the beginning and end of a range of log records to identify as candidates for the record selection process
- Control the range of log data sets to allocate dynamically from RECON information
Reduce the overall number of log records that are selected from input SLDS or extract files

For more information, see “Specifying INTERVAL control statements” on page 79.

To access and work with the INTERVAL Control Statement panel

1. Access the Build JCL panel.
   For instructions, see “Accessing JCL functions” on page 138.

2. In the Control Statements field to the left of the INTERVAL label, enter S.
   The INTERVAL Control Statement panel is displayed:

   **Figure 39: INTERVAL Control Statement panel**

<table>
<thead>
<tr>
<th>File</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>INTERVAL Control Statement</td>
<td></td>
</tr>
<tr>
<td>Command ===&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Start type.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Timestamp</td>
<td>Date (YYYYDDD)</td>
<td>Time (hhmmssth)</td>
</tr>
<tr>
<td>2. First log record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Log sequence number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stop type.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Timestamp</td>
<td>Date (YYYYDDD)</td>
<td>Time (hhmmssth)</td>
</tr>
<tr>
<td>2. Last log record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Log sequence number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Additional intervals. Select if desired.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Start timestamp</td>
<td>Date (YYYYDDD)</td>
<td>Time (hhmmssth)</td>
</tr>
<tr>
<td>- Stop timestamp</td>
<td>Date (YYYYDDD)</td>
<td>Time (hhmmssth)</td>
</tr>
</tbody>
</table>

3. In the Start type field, enter the value that will be used to generate the START keyword on the INTERVAL control statement.
   This keyword identifies the beginning of a range of log records to be considered as candidates for the record selection process. For more information, see “START keyword” on page 80.

4. In the Stop type field, enter the value that will be used to generate the STOP keyword on the INTERVAL control statement.
   This keyword identifies the end of a range of log records to be considered as candidates for the record selection process. For more information, see “STOP keyword” on page 82.
5  \textit{(optional)} In the \textbf{Additional intervals} field, you can enter the timestamp values that will be used to generate the START keyword and the STOP keyword on another \textbf{INTERVAL} control statement.

For more information, see “START keyword” on page 80 and “STOP keyword” on page 82.

### Building the 	extbf{REPORTS} control statement

To build keywords and values on the \textbf{REPORTS} control statement, you use the \textbf{REPORTS Control Statement} panel.

The \textbf{REPORTS} control statement controls production of the batch reports that Log Analyzer writes to SYSOUT-type data sets. For more information, see “Specifying \textbf{REPORTS} control statements” on page 107.

#### To access and work with the \textbf{REPORTS} Control Statement panel

1  Access the Build JCL panel.

   For instructions, see “Accessing JCL functions” on page 138.

2  In the \textbf{Control Statements} field to the left of the \textbf{REPORTS} label, enter \textbf{S}.

   The \textbf{REPORTS} Control Statement panel is displayed:

\begin{verbatim}
Figure 40: REPORTS Control Statement panel

Log Analyzer               REPORTS Control Statement
Command ===> _________________________________________________________________
Reports. Select the reports to produce. (*=more options below for that report)
/ Analysis Summary         Limit SYSOUT to ________ lines
/ LUOW Summary             Limit SYSOUT to ________ lines
/ LUOW Detail *            Limit SYSOUT to ________ lines
/ LUOW Time Sequence       Limit SYSOUT to ________ lines
/ Log Record Dump *        Limit SYSOUT to ________ lines
/ Orphans                  Limit SYSOUT to ________ lines
   Audit                   Limit SYSOUT to ________ lines
   / Application Checkpoint *  The following reports are mutually exclusive with the above and each other.
   / Abends                  Limit SYSOUT to ________ lines
   / Deadlock
   / RBA Buffer Statistics *
   / LUOW Detail report options. Select if desired.
   / Display 01 and 03 record data. Offset ______ Length __
   / Limit to LUOW ID(s) ______________________________________________________
Log Record Dump report options.
   / Format.
   / 1. Standard 2. Over/under _ Character _ Hexadecimal
Audit report options.
   / Event type to include or exclude.
   / 1. Include 2. Exclude
\end{verbatim}

162  \textit{BMC Log Analyzer for IMS User Guide}
3 In the **Reports** selection field to the left of each type of batch report that you want Log Analyzer to produce, enter a slash (/).

These fields are used to produce the following keywords on the REPORTS control statement, as explained on the indicated page:

- “SUMMARY keyword” on page 128
- “LUOWSUMM keyword” on page 125
- “LUOWDETAIL keyword” on page 123
- “LUOWTSEQ keyword” on page 125
- “DUMPREC keyword” on page 119 or “DUMPVERT keyword” on page 120
- “ORPHANS keyword” on page 126
- “AUDIT keyword” on page 115
- “APPCHECK keyword” on page 111
- “ABENDS subkeyword” on page 94
- “DEADLOCK keyword” on page 118
- “RBASTATS keyword” on page 127

In the **Limit SYSOUT to n lines** field for many of the reports, you can also enter the maximum number of lines to write to the output data set for that report. If you omit this number, all lines that Log Analyzer produces for this report are written to the output data set.
If you select the Deadlock report or the Application Checkpoint report, you cannot select the FILTER control statement on the Build JCL panel. The ISPF interface issues a warning message if the FILTER control statement is selected and you select one of these reports. To correct this condition, remove any slashes from the Deadlock field and Application Checkpoint field, or remove the slash from the FILTER control statement field.

If you select the Abends report, you cannot specify SELECT fields for the FILTER control statement field.

4 *(optional)* Enter a slash (/) in the Display 01 and 03 record data field to generate the LUOWDETAILDATA keyword on the REPORTS control statement.

This keyword controls whether Log Analyzer includes a portion of data from 01 records and 03 records in the LUOW Detail report. If you select this field, you can also specify an offset and length. For more information, see “LUOWDETAILDATA keyword” on page 124.

5 *(optional)* Enter a list of specific LUOW IDs in the Limit to LUOW ID(s) field to limit the LUOW Detail report to the specified list.

This keyword modifies the generation of the LUOWDETAIL keyword on the REPORTS control statement. For more information, see “LUOWDETAIL keyword” on page 123.

6 *(optional)* In the Log Record Dump report options field, enter one of the following values to control the format of dumped log records:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The DUMPREC keyword is generated so that dumped records are formatted in standard format. For more information, see “DUMPREC keyword” on page 119.</td>
</tr>
</tbody>
</table>
| 2     | The DUMPVERT keyword is generated so that dumped records are formatted in vertical format. For more information, see “DUMPVERT keyword” on page 120.  
You can also enter a slash (/) in the Character field, in the Hexadecimal field, or in both fields to include character representations, hexadecimal representations, or both representations of the dumped records. The DUMPFORM keyword is generated. For more information, see “DUMPFORM keyword” on page 119. |

7 *(optional)* In the AUDIT report options fields, enter values to control the contents of the AUDIT report:

*Note*  
If you omit all of these values and you select the report, the ISPF interface generates the AUDIT=ALL keyword.
a  

(optional) In the **Event type to include or exclude** field, enter one of the following values to control generation of the TYPE subkeyword of the AUDIT keyword:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>In the AUDIT report, display information only for type that you specify in the <strong>Event type</strong> field. Omit information about all other types that Log Analyzer finds in the input logs.</td>
</tr>
<tr>
<td>2</td>
<td>In the AUDIT report, omit information about the type that you specify in the <strong>Event type</strong> field. Display information about all other types that Log Analyzer finds in the input logs.</td>
</tr>
</tbody>
</table>

b  

(optional) In the **DBDs to include or exclude** field, enter one of the following values to control generation of the DBD subkeyword of the AUDIT keyword:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>In the AUDIT report, display information only for DBDs that you specify in the <strong>DBD names</strong> field. Omit information about all other DBDs that Log Analyzer finds in the input logs.</td>
</tr>
<tr>
<td>2</td>
<td>In the AUDIT report, omit information about the DBDs that you specify in the <strong>DBD names</strong> field. Display information about all other DBDs that Log Analyzer finds in the input logs.</td>
</tr>
</tbody>
</table>

If you specify a value in the **DBDs to include or exclude field**, enter one or more DBD names in the **DBD names** field.

You can enter a masked value by using the following wildcard characters:

- Specify an asterisk (*) to match any number of characters, starting from that position.
- Specify a percent sign (%) to match a single character at that position.

Separate multiple names with a space, a comma, or both. The ISPF interface separates the names with a comma, as required by the keyword syntax.

8  

(optional) In the **Application Checkpoint report options** fields, enter values to control the contents of the Application Checkpoint report:

**Note**

If you omit all of these values and you select the report, the ISPF interface generates the APPCHECK=ALL keyword. This keyword causes Log Analyzer to include all BMPs in the analysis and bypass filtering by checkpoint frequency:

a  

(optional) In the **Jobs to include or exclude** field, enter one of the following values to control generation of the JOB subkeyword of the APPCHECK keyword:
In the Application Checkpoint report, display information only for jobs that you specify in the **Jobnames** field. Omit information about all other jobs that Log Analyzer finds in the input logs.

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>In the Application Checkpoint report, display information only for jobs that you specify in the <strong>Jobnames</strong> field. Omit information about all other jobs that Log Analyzer finds in the input logs.</td>
</tr>
</tbody>
</table>

In the Application Checkpoint report, omit information about the jobs that you specify in the **Jobnames** field. Display information about all other jobs that Log Analyzer finds in the input logs.

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>In the Application Checkpoint report, omit information about the jobs that you specify in the <strong>Jobnames</strong> field. Display information about all other jobs that Log Analyzer finds in the input logs.</td>
</tr>
</tbody>
</table>

If you specify a value in the **Jobs to include or exclude** field, enter one or more job names in the **Jobnames** field.

You can enter a masked value by using wildcard characters:

- Specify an asterisk (*) to match any number of characters, starting from that position.
- Specify a percent sign (%) to match a single character at that position.

Separate multiple names with a space, a comma, or both. The ISPF interface separates the names with a comma, as required by the keyword syntax.

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>(optional) In the <strong>PSBs to include or exclude</strong> field, enter one of the following values to control generation of the PSB subkeyword of the APPCHECK keyword:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>In the Application Checkpoint report, display information only for BMPs with PSBs that you specify in the <strong>PSB names</strong> field. Omit information about all other BMPs that Log Analyzer finds in the input logs.</td>
</tr>
</tbody>
</table>

In the Application Checkpoint report, omit information about the BMPs with PSBs that you specify in the **PSB names** field. Display information about all other BMPs that Log Analyzer finds in the input logs.

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>In the Application Checkpoint report, omit information about the BMPs with PSBs that you specify in the <strong>PSB names</strong> field. Display information about all other BMPs that Log Analyzer finds in the input logs.</td>
</tr>
</tbody>
</table>

If you specify a value in the **PSBs to include or exclude** field, enter one or more PSB names in the **PSB names** field.

You can enter a masked value by using wildcard characters:

- Specify an asterisk (*) to match any number of characters, starting from that position.
- Specify a percent sign (%) to match a single character at that position.

Separate multiple names with a space, a comma, or both. The ISPF interface separates the names with a comma, as required by the keyword syntax.
c (optional) Use the **Checkpoint frequency** fields to establish a filter for the Application Checkpoint report, based on the calculated frequency of checkpoints.

These fields control generation of the CHKFREQ subkeyword and the THRESH subkeyword of the APPCHECK keyword.

- In the **Comparison operator** field, enter one of the following values to define how to compare the calculated frequency of checkpoints with the threshold value that you specify in the **Number of checkpoints** field.

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>Display data for this job and PSB if the calculated checkpoint frequency is less than the specified threshold value.</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Display data for this job and PSB if the calculated checkpoint frequency is less than or equal to the specified threshold value.</td>
</tr>
<tr>
<td>&gt;</td>
<td>Display data for this job and PSB if the calculated checkpoint frequency is greater than the specified threshold value.</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Display data for this job and PSB if the calculated checkpoint frequency is greater than or equal to the specified threshold value.</td>
</tr>
</tbody>
</table>

- In the **Number of checkpoints** field, specify the threshold number of checkpoints to compare with the calculated frequency of checkpoints. Valid values are 0 through 9999.

- In the **Per threshold** field, specify the time unit for the threshold value. Valid values are SEC (seconds) and MIN (minutes).

For the filter, you specify a comparison operator, the threshold number of checkpoints per time unit, and the time unit. Log Analyzer calculates the checkpoint frequency for an included job and PSB by dividing the number of checkpoint records by the elapsed time of the job (as reflected in the input log records). Then, Log Analyzer compares this calculated checkpoint frequency with the threshold number of checkpoints per time unit. If the comparison results in a true statement, the report displays information about the job and PSB.

9 (optional) In the **RBA Buffer Statistics report options** field, enter a value to indicate the depth of information to report in the RBA Buffer Statistics report:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Report summarized (total) values.</td>
</tr>
<tr>
<td>2</td>
<td>Report information about all RBAs.</td>
</tr>
</tbody>
</table>
Building EXTRACT, INDEXFILE, and WORKFILE control statements

To build keywords and values on control statements that are used to allocate Log Analyzer external files, use the following panels:

- The EXTRACT Control Statement panel (Figure 41 on page 168) builds control statements to allocate an extract file. Log Analyzer writes all log records that are associated with LUOWs of interest to this file.

- The INDEXFILE Control Statement panel (which is similar to Figure 41 on page 168) to builds control statements to allocate an index file. The Log Analyzer ISPF interface uses this file during interactive analysis of LUOWs.

- The WORKFILE Control Statement panel (which is similar to Figure 41 on page 168) builds control statements to allocate a work file. Log Analyzer uses this file to sort log records that it reads from an input extract file.

For more information, see “Specifying EXTRACT, INDEXFILE, and WORKFILE control statements” on page 128

To access and work with fileType Control Statement panels

1. Access the Build JCL panel.

   For instructions, see “Accessing JCL functions” on page 138.

2. In the Control Statements field to the left of the EXTRACT, INDEXFILE, or WORKFILE label, enter S.

   One of the following panels is displayed, depending on which label you selected:

   - EXTRACT Control Statement panel (see Figure 41 on page 168)
   - INDEXFILE Control Statement panel
   - WORKFILE Control Statement panel

   These panels are similar to each other but contain different fields and generate different control statements (based on your entries).

Figure 41: EXTRACT Control Statement panel

File Options Help
Log Analyzer EXTRACT Control Statement
Command ===> _________________________________________________________________
Extract file data set name  . . _____________________________________________
Enter allocation information.
   SMS Management class . . . . ________ (For SMS managed data set)
 SMS Storage class . . . . . . ________ (For SMS managed data set)
 SMS Data class  . . . . . . . ________ (For SMS managed data set)
 Volume serial(s) . . . . . . _______________________________________________
 Generic unit . . . . . . . . ________
 Space units . . . . . . . . ___ (TRK or CYL)
 Primary quantity . . . . . . ___ (in above units)
 Secondary quantity . . . . . ___ (in above units)

The following fields may be specified to model the allocation after an existing data set. They are mutually exclusive.
 LIKE data set name (SMS) . _____________________________________________
 DCB data set name (non-SMS) . _____________________________________________

3 In the `fileType file data set name` field, enter the data set name of the data set to be allocated.

This information is used to generate the DSN keyword on the `fileType` control statement. For more information, see “DSN keyword” on page 132.

**Note**

This field is not displayed on the WORKFILE Control Statement panel.

4 Provide values in the following data entry fields as required:

<table>
<thead>
<tr>
<th>Field</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMS Management class</td>
<td>Specify the name of the SMS management class to use for the data set to be allocated. This keyword is used to produce the MGMTCLASS keyword on the <code>fileType</code> control statement. For more information, see “MGMTCLASS keyword” on page 133.</td>
</tr>
<tr>
<td>SMS Storage class</td>
<td>Specify the name of the SMS storage class to use for the data set to be allocated. This keyword is used to produce the STORCLASS keyword on the <code>fileType</code> control statement. For more information, see “STORCLASS keyword” on page 134.</td>
</tr>
<tr>
<td>SMS Data class</td>
<td>Specify the name of the SMS data class to use for the data set to be allocated. This keyword is used to produce the DATACLASS keyword on the <code>filetype</code> control statement. For more information, see “DATACLASS keyword” on page 132.</td>
</tr>
<tr>
<td>Volume serial(s)</td>
<td>Specify the name of the volume serial number to use for the data set to be allocated. This keyword is used to produce the VOLSER keyword on the <code>fileType</code> control statement. For more information, see “VOLSER keyword” on page 135.</td>
</tr>
<tr>
<td>Field</td>
<td>Specification</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Generic unit</td>
<td>Specify the unit type to use for the data set to be allocated. This keyword is used to produce the UNIT keyword on the EXTRACT control statement. For more information, see “UNIT keyword” on page 134. This field is not valid for the INDEXFILE control statement.</td>
</tr>
<tr>
<td>Space units</td>
<td>Specify whether to allocate space in cylinders or tracks. This keyword is used to produce the SPACEUNITS keyword on the fileType control statement. For more information, see “SPACEUNITS keyword” on page 134.</td>
</tr>
<tr>
<td>Primary quantity</td>
<td>Specify the amount of primary space to use for the data set to be allocated. This keyword is used to produce the PRISPACE keyword on the fileType control statement. For more information, see “PRISPACE keyword” on page 133.</td>
</tr>
<tr>
<td>Secondary quantity</td>
<td>Specify the amount of secondary space to use for the data set to be allocated. This keyword is used to produce the SECSPACE keyword on the fileType control statement. For more information, see “SECSPACE keyword” on page 134.</td>
</tr>
<tr>
<td>LIKE data set name (SMS)</td>
<td>Specify the data set name of a data set to use as a model for allocation of a new extract data set as an SMS-managed data set. This keyword is used to produce the LIKE keyword on the EXTRACT control statement. For more information, see “LIKE keyword” on page 133. This field is not valid for the INDEXFILE control statement.</td>
</tr>
<tr>
<td>DCB data set name (non-SMS)</td>
<td>Specify the data set name of a data set to use as a model for allocation of a new extract data set as a non-SMS-managed data set. This keyword is used to produce the DCB keyword on the EXTRACT control statement. For more information, see “DCB keyword” on page 132. This field is not valid for the INDEXFILE control statement.</td>
</tr>
</tbody>
</table>
Submitting Log Analyzer jobs and checking results

The values that you specify on the Build JCL panel and its dependent panels are saved in your ISPF profile data set.

The ISPF interface uses those profile values to generate JCL for executing an analysis job. The interface provides panels and commands that you can use to generate, view, and submit the analysis job.

Generating and viewing Log Analyzer JCL

At any time while you are specifying values for Log Analyzer control statements and keywords through the ISPF interface, you can see how the values will be used in generated JCL.

When you are ready to submit the job for execution, you can view the generated JCL first to ensure that it is complete and correct.

To generate and view JCL

1. Generate the JCL by using one of the following methods:
   - On the Build JCL panel or any of its dependent panels, enter the VIEW command in the Command area.
   - On the Build JCL panel or any of its dependent panels, select File from the action bar and enter 1 in the choice entry field.
   - On the JCL Menu panel, enter 2 in the JCL actions field.

The ISPF interface generates the JCL and displays it in a temporary data set in an ISPF Browse session, such as shown in the following example. You can use standard Browse commands (such as DOWN, UP, CREATE, and SUBMIT) to work with this data set.

Figure 42: View JCL (in a temporary data set) panel
You can work with the generated JCL as follows:

- Check the JCL to ensure that JOB, EXEC, STEPLIB, DD, and control statements have been generated as expected and required. For more information, see “Working with Log Analyzer jobs” on page 55.
- If all statements appear to be correct, you can enter the SUBMIT command to submit the analysis job for execution.
- If a statement appears to be incorrect or incomplete, enter the END command to return to the panel from which you generated the JCL. Change the incorrect value or provide a value, and view the JCL again.

Generating and submitting Log Analyzer jobs

You can submit a generated job for execution after viewing the JCL, or you can generate the job and submit it for execution without viewing the JCL first.

To generate and submit a job

1. Use one of the following methods:
   - On the Build JCL panel or any of its dependent panels, enter the SUBMIT command in the command area.
   - On the Build JCL panel or any of its dependent panels, select File from the action bar and enter 2 in the choice entry field.
   - On the JCL Menu panel, enter 3 in the JCL actions field.

2. Wait for the ISPF interface to generate the JCL and submit the JCL for execution.
Checking results

When you submit a Log Analyzer job for execution and the job is complete, you should check the results to ensure that the job has been completed successfully and has produced the output that you wanted.

For more information, see “Checking results of Log Analyzer jobs” on page 135.
Working with batch reports

Log Analyzer can produce batch reports that you can use to verify the success of an analysis job and to examine the results manually.

The Log Analyzer ISPF interface provides mostly the same data as the data that is available in batch reports, but the data might be presented in a different format or location. Certain reports are available in batch format but not in the ISPF interface.

Note
The report examples in this chapter might have been produced by different analysis jobs; therefore, data within the reports might not coincide with data in other reports.

Producing and accessing batch reports

To control whether and how Log Analyzer produces batch reports, use keywords on the REPORTS control statement.

For more information, see “Specifying REPORTS control statements” on page 107.

Log Analyzer writes each report to a separate SYSOUT-type data set. The ddname of the data set corresponds to the keyword that controls the production of the report.

The following table lists batch reports that Log Analyzer can produce.
### Table 14: Log Analyzer batch reports

<table>
<thead>
<tr>
<th>Keyword on <code>REPORTS</code> statement</th>
<th>ddname</th>
<th>Report title</th>
<th>Description</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUMMARY</td>
<td>SUMMARY</td>
<td>Analysis Summary</td>
<td>This report summarizes information about the log records that Log Analyzer handled during the analysis job, about how Log Analyzer has organized these records into LUOWs, and about the types of log records that were read.</td>
<td>“Using the Analysis Summary report” on page 178</td>
</tr>
<tr>
<td>AUDIT</td>
<td>AUDIT</td>
<td>Auditing Report</td>
<td>This report provides information that is relevant to an audit of events.</td>
<td>“Using the Auditing report” on page 185</td>
</tr>
<tr>
<td>LUOWSUMM</td>
<td>LUOWSUMM</td>
<td>LUOW Summary</td>
<td>This report summarizes information about the LUOWs that Log Analyzer created during the analysis job.</td>
<td>“Using the LUOW Summary report” on page 188</td>
</tr>
<tr>
<td>LUOWDETAIL</td>
<td>LUOWDET</td>
<td>LUOW Detail</td>
<td>This report lists key information about log records that Log Analyzer associated with an LUOW.</td>
<td>“Using the LUOW Detail report” on page 192</td>
</tr>
<tr>
<td>DUMPREC</td>
<td>DUMPREC</td>
<td>Record Dumps</td>
<td>This report displays the contents of log records in a standard format.</td>
<td>“Using the Record Dumps report” on page 195</td>
</tr>
<tr>
<td>DUMPVERT</td>
<td>DUMPVERT</td>
<td>Record Dumps</td>
<td>This report displays the contents of log records in a vertical format.</td>
<td>“Using the Record Dumps report” on page 195</td>
</tr>
<tr>
<td>Keyword on REPORTS statement</td>
<td>ddbname</td>
<td>Report title</td>
<td>Description</td>
<td>Topic</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------</td>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------------------------</td>
</tr>
<tr>
<td>LUOWTSEQ</td>
<td>LUOWTSEQ</td>
<td>Time Sequence</td>
<td>This report lists log records that have been selected for analysis in the order in which they were written to an IMS log.</td>
<td>“Using the Time Sequence report” on page 198</td>
</tr>
<tr>
<td>ORPHANS</td>
<td>ORPHAN</td>
<td>Orphans</td>
<td>This report provides information about records for which Log Analyzer was unable to resolve associations and relate the record to an LUOW.</td>
<td>“Using the Unresolved Orphans report” on page 200</td>
</tr>
<tr>
<td>ABEND</td>
<td>ABEND</td>
<td>Abend</td>
<td>This report focuses on type 07 records to identify transactions or programs that have terminated abnormally.</td>
<td>“Using the ABEND report” on page 202</td>
</tr>
<tr>
<td>DEADLOCK</td>
<td>DEADLOCK</td>
<td>Deadlocks</td>
<td>This report provides information about deadlocks.</td>
<td>“Using the Deadlocks report” on page 203</td>
</tr>
<tr>
<td>APPCHECK</td>
<td>APPCHECK</td>
<td>Application Checkpoint</td>
<td>This report provides information about the frequency of checkpoints issued by batch message processing programs.</td>
<td>“Using the Application Checkpoint report” on page 206</td>
</tr>
<tr>
<td>RBASTATS</td>
<td>RBASTATS</td>
<td>RBA Buffer Statistics</td>
<td>This report provides information about the distribution of updated relative byte addresses (RBAs) or block numbers in full-function IMS databases.</td>
<td>“Using the RBA Buffer Statistics report” on page 209</td>
</tr>
</tbody>
</table>
Using batch report headings

Log Analyzer creates all batch reports with a heading at the top of each page.

The following table lists and describes fields in the report heading.

Table 15: Headings in Log Analyzer batch reports

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>yyyy-ddd</td>
<td>The yyyy-ddd field contains the Julian date on which the analysis job was executed.</td>
</tr>
<tr>
<td>Log Analyzer for IMS V v.r.mm.nn</td>
<td>The Log Analyzer for IMS V v.r.mm.nn field indicates Log Analyzer version, release, maintenance, and (if applicable) level numbers.</td>
</tr>
<tr>
<td>Page nnn</td>
<td>The Page nnn field identifies the current page of the report.</td>
</tr>
<tr>
<td>report title (keyword)</td>
<td>The report title (keyword) field identifies the type of report and the keyword (on the REPORTS control statement) that was used to request the report.</td>
</tr>
<tr>
<td>Log time span</td>
<td>The Log time span field contains the time stamps of the oldest log record (reported in the FROM field) and the most recent log record (reported in the TO field) that Log Analyzer has considered as a candidate for selection. For information about how Log Analyzer selects these records, see “Identifying candidate records and creating LUOWs” on page 269.</td>
</tr>
<tr>
<td>DURATION</td>
<td>The DURATION field contains the calculated difference between the time stamps of the oldest log record and the most recent log record that Log Analyzer has considered as a candidate for selection.</td>
</tr>
</tbody>
</table>

Using the Analysis Summary report

The Analysis Summary report summarizes information about quantities and types of log records that Log Analyzer handled during the analysis job, how Log Analyzer organized these records into LUOWs, and characteristics that are associated with the LUOWs.

Figure 43: Analysis Summary report

```
2009-111                             Log Analyzer for IMS V1.2.01.00
Analysis Summary (SUMMARY)           Page 1
LDQ time span: FROM 2007-201 08:48:55.83 TO 2007-201 09:00:38.23 DURATION 00:11:42

********************
Record and LUOW Counts
********************
Log records read 008568390 Passed to select process 008216580 Selected log records 000000000
LUOWs 0000000179 Unresolved associations 0000899451 Extract file count 000000000

Other IMSIDs encountered: IMSBXRF IMSKXRF IMSJXRF IMSIXRF IMSDXRF IMSJXRF IMSKXRF

**********
LUOW Lists (* indicates abend occurred in some LUOWs)
**********
```
Overview of the Analysis Summary report

You can use the Analysis Summary report to obtain general information about an analysis job.

This information is useful for purposes such as these:

- If you do not have much information about the contents of the input log record sources (SLDSs, OLDSs and input extract data sets that were created by a previous analysis job), this report gives you a general idea about the IMS traffic that is represented and whether specified time ranges and log record sources are appropriate for your purposes. This information can help you if Log Analyzer has allocated log sources automatically from information in the RECONs and you are not familiar with the characteristics of the sources that were allocated.
If you are developing a set of filter criteria, this report shows you the types, quantities, and characteristics of records that Log Analyzer has included and excluded, based on the filter criteria. You can use the report to determine whether the appropriate records have been included for your purposes and whether enough records have been excluded so that you can analyze the included records interactively or manually with a reasonable amount of time and effort.

If you do not know the exact identity of an origin, destination, user ID, or other characteristic of a LUOW that might be useful for your purposes, the report provides a list of candidates for more study.

If you do not know where and how to start diagnosing a problem, the report gives you a broad view of conditions surrounding the problem.

Log Analyzer produces the Analysis Summary report and writes it to the SUMMARY data set when you specify the SUMMARY keyword on the REPORTS control statement or when you omit the REPORTS control statement. For more information, see “SUMMARY keyword” on page 128.

**Information in the Analysis Summary report**

The information in the Analysis Summary report is divided into the following sections:

- Record and LUOW Counts
- LUOW Lists
- Count of processed records by code
- Count of nonselected records by code

**Record and LUOW Counts section**

The Record and LUOW Counts section of the Analysis Summary report provides counts of records and LUOWs that were handled during the analysis job and other general information about the job.

The following table lists and describes the fields in this section.
### Table 16: Analysis Summary report—Record and LUOW Counts section fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log records read</td>
<td>The <strong>Log records read</strong> field contains a count of all log records (except header records) that are contained in all input log record sources. Log Analyzer has read all of these records to determine whether they are candidates for selection. This field can help you to gain a general understanding of the quantity of IMS traffic that is represented by the records in the input log record sources.</td>
</tr>
<tr>
<td>Passed to select process</td>
<td>The <strong>Passed to select process</strong> field contains a count of log records that Log Analyzer identified as candidates for selection. For information about how Log Analyzer identifies these candidates, see “Identifying candidate records and creating LUOWs” on page 269. This field can help you to evaluate whether control statement effectively influenced the identification of candidates. If too many records were passed to the selection process, you can rerun the analysis job with narrower specifications.</td>
</tr>
<tr>
<td>Selected log records</td>
<td>The <strong>Selected log records</strong> field contains a count of log records that Log Analyzer selected for inclusion in LUOWs. For information about how Log Analyzer selects these records, see “Selecting log records and LUOWs” on page 283. If you do not specify any filter criteria, the value in this field is identical to the value in the <strong>Passed to select process</strong> field. This field can help you to evaluate the effectiveness of control statements that influenced the selection process. If too many records were selected, you can rerun the analysis job with narrower specifications.</td>
</tr>
<tr>
<td>LUOWs</td>
<td>The <strong>LUOWs</strong> field contains the count of LUOWs that Log Analyzer created during the analysis job. The value in this field is the sum of the count of transaction-type LUOWs (LUOWs with an origin, a destination, and a user ID) and the count of &quot;other&quot; LUOWs. This field can help you to determine whether the volume of LUOWs is small enough for you to analyze effectively. For information about how Log Analyzer creates LUOWs, see “Creating LUOWs” on page 281.</td>
</tr>
<tr>
<td>Unresolved associations</td>
<td>The <strong>Unresolved associations</strong> field contains the count of records that Log Analyzer was unable to assign to a LUOW. For information about the cause of this condition, see “Creating LUOWs” on page 281.</td>
</tr>
<tr>
<td>Extract file count</td>
<td>The <strong>Extract file count</strong> field contains a count of log records that Log Analyzer wrote to an extract data set. If you provide an EXTRACT statement, this count reflects the count of selected log records minus the count of unresolved associations. If you do not include an EXTRACT control statement, this count is zero. This field can help you to verify the volume of extracted log records.</td>
</tr>
<tr>
<td>IMSIDs encountered</td>
<td>The <strong>IMSIDs encountered</strong> field contains a list of IMS systems, each of which has created one or more records that the analysis job has handled.</td>
</tr>
</tbody>
</table>
The Other IMSIDs encountered field (if displayed) contains a list of IMS systems for which Log Analyzer has detected activity but for which logs spanning the same time range were not processed during this analysis job. If the unprocessed logs are added to the analysis job, Log Analyzer can provide a more complete view of the transactions.

The Note: Remote IMS connections found field (if displayed) indicates that Log Analyzer has detected activity in logs for remote IMS systems, but the systems cannot be identified from the included logs.

### LUOW Lists section

The LUOW Lists section of the Analysis Summary report lists the counts of LUOWs that are associated with specific origins, destinations, user IDs, and PSBs, and lists the counts of other types of LUOWs.

For more information about types of LUOWs, see “Types of LUOWs” on page 302.

**Note**

The Analysis Summary report indicates that one or more abends have been detected in LUOWs for a specific element (origin name, destination name, user ID, or PSB) by placing an asterisk after the count of LUOWs for the element.

The following table lists and describes the fields in this section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Origin summary</strong></td>
<td><strong>Origin summary</strong> fields provide the following information about transaction origins (such as an LTERM) that were identified during the analysis job:</td>
</tr>
<tr>
<td></td>
<td>■ Count of unique origins</td>
</tr>
<tr>
<td></td>
<td>■ Total count of LUOWs for all origins</td>
</tr>
<tr>
<td></td>
<td>■ Name of each unique origin</td>
</tr>
<tr>
<td></td>
<td>■ For each unique origin, count of LUOWs with one or more records for that origin</td>
</tr>
<tr>
<td></td>
<td>Lowercase characters denote an element with no specifically identifiable origin. For example, <code>ims</code> denotes that the origin was an IMS system (rather than a specific LTERM).</td>
</tr>
<tr>
<td></td>
<td>For a non-message-driven BMP, the origin name field contains the label <code>NMSGBMP</code>.</td>
</tr>
<tr>
<td></td>
<td>An asterisk indicates that one or more abends were detected in LUOWs for the origin.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Destination summary</strong></td>
<td><strong>Destination summary</strong> fields provide the following information about transaction destinations (such as an LTERM) that were identified during the analysis job:</td>
</tr>
<tr>
<td>■ Count of unique destination</td>
<td></td>
</tr>
<tr>
<td>■ Total count of LUOWs for all destinations</td>
<td></td>
</tr>
<tr>
<td>■ Name of each unique destination</td>
<td></td>
</tr>
<tr>
<td>■ For each unique origin, count of LUOWs with one or more records for that destination</td>
<td></td>
</tr>
<tr>
<td>For a non-message-driven BMP, the destination name field contains the label <strong>NMSGBMP</strong>.</td>
<td></td>
</tr>
<tr>
<td>An asterisk indicates that one or more abends were detected in LUOWs for the destination.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>User ID summary</strong></th>
<th><strong>User ID summary</strong> fields provide the following information about the user IDs that were identified during the analysis job:</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ Count of unique user IDs</td>
<td></td>
</tr>
<tr>
<td>■ Total count of LUOWs for all user IDs</td>
<td></td>
</tr>
<tr>
<td>■ Each unique user ID</td>
<td></td>
</tr>
<tr>
<td>■ For each unique user ID, count of LUOWs with one or more records for that user ID</td>
<td></td>
</tr>
<tr>
<td>The USERID subkeyword of the SELECT keyword on the FILTER control statement applies to all types of log records that are associated with user IDs. If you specify a USERID filter, the count of LUOWs in this section is the sum of the count of transaction-type LUOWs and the count of other types of LUOWs that are associated with user IDs.</td>
<td></td>
</tr>
<tr>
<td>Lowercase characters denote an element with no specifically identifiable user ID. For example, <strong>nonusid</strong> denotes a LUOW with a security segment (indicating that it is a transaction LUOW), but the security segment does not contain a user ID. The string <strong>none</strong> denotes a transaction LUOW with no security segment.</td>
<td></td>
</tr>
<tr>
<td>For a non-message-driven BMP, the user ID field contains the PSB name.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>PSB summary</strong></th>
<th><strong>PSB summary</strong> fields provide the following information about the PSBs that were identified during the analysis job:</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ Count of unique PSBs</td>
<td></td>
</tr>
<tr>
<td>■ Total count of LUOWs for all PSBs</td>
<td></td>
</tr>
<tr>
<td>■ Each unique PSB</td>
<td></td>
</tr>
<tr>
<td>■ For each unique PSB, count of LUOWs with one or more records for that PSB</td>
<td></td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS Tran summary</td>
<td>If Log Analyzer has detected one or more CICS transactions, <strong>CICS Tran summary</strong> fields provide the following information about the CICS transactions that were identified during the analysis job:</td>
</tr>
</tbody>
</table>
|                           | ■ Count of unique CICS transactions  
|                           | ■ Total count of LUOWs for all CICS transactions  
|                           | ■ Each unique CICS transaction  
|                           | ■ For each unique CICS transaction, count of LUOWs with one or more records for that transaction  
| Other summary             | **Other summary** fields provide the following information about nontransaction LUOWs and miscellaneous LUOWs that were created during the analysis job: |
|                           | ■ Count of unique types of nontransaction and miscellaneous LUOWs  
|                           | ■ Total count of nontransaction and miscellaneous LUOWs  
|                           | ■ Name of each unique type of nontransaction or miscellaneous LUOWs  
|                           | ■ For each unique type, count of LUOWs of this type  
|                           | For more information, see “Types of LUOWs” on page 302. |

#### Count of processed records by code section

The **Count of processed records by code** section of the Analysis Summary report provides a count of every type of log record that Log Analyzer has selected for analysis.

Many factors, such as specified filter criteria and time ranges, influence which log record types are selected. For more information, see “Selecting log records and LUOWs” on page 283.

The counts are provided in a matrix. The junction of each row and each column is assigned to a log record type and provides the count of records of that type. The row represents the first character of the two-character log record type code. The column represents the second character of the log record type code. For example, the junction of row 5* and column *0 is assigned to log record type 50. A blank junction indicates that Log Analyzer did not read any records of that type.

**Note**
If subtypes of a specific log record type are included or excluded by using the LOGRECORDCODES keyword on the FILTER control statement, the **Count of processed records by code** section and the **Count of nonselected records by code** section can contain nonzero counts for that log record type.
Count of nonselected records by code section

The **Count of nonselected records by code** section of the Analysis Summary report section provides a count of every type of log record that Log Analyzer has not selected for analysis.

Many factors, such as specified filter criteria and time ranges, influence which log record types are selected. For more information, see “Selecting log records and LUOWs” on page 283.

The counts are provided in a matrix. The junction of each row and each column is assigned to a log record type and provides a count of records of that type. The row represents the first character of the two-character log record type code. The column represents the second character of the log record type code. For example, the junction of row 5* and column *0 is assigned to log record type 50. A blank junction indicates that Log Analyzer did not read any records of that type.

**Note**
If subtypes of a specific log record type are included or excluded by using the LOGRECORDCODES keyword on the FILTER control statement, the **Count of processed records by code** section and the **Count of nonselected records by code** section can contain nonzero counts for that log record type.

If all record types have been selected for analysis, this section of the report is not produced.

Using the Auditing report

The Auditing report provides information that is relevant to an audit of events that have occurred in the IMS environment. Reported events can include signon actions, signoff actions, transactions, commands, and security violations.

Following is an example of an AUDIT report:

```
2015-200                      Log Analyzer for IMS XL.5.00.03               Page 1
Auditing Report  (AUDIT)
No report filters were selected

<table>
<thead>
<tr>
<th>Type</th>
<th>Time</th>
<th>Target</th>
<th>LUOW#</th>
<th>Log Seq #</th>
<th>Additional data</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td>12:45:31.74</td>
<td>0056</td>
<td>00000000000000F6</td>
<td>STADC .</td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>12:45:48.51</td>
<td>0065</td>
<td>0000000000000106</td>
<td>STADC .</td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>12:45:48.52</td>
<td>0068</td>
<td>000000000000010B</td>
<td>STAC NOD <em>VD1I</em> .</td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>12:45:48.52</td>
<td>0072</td>
<td>0000000000000111</td>
<td>STA USR *PS35</td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>12:45:48.55</td>
<td>0074</td>
<td>0000000000000134</td>
<td>STA LTI* .</td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>12:45:48.55</td>
<td>0088</td>
<td>0000000000000135</td>
<td>STA MGR* .</td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>12:45:48.55</td>
<td>0092</td>
<td>0000000000000134</td>
<td>STA USR *PS35</td>
<td></td>
</tr>
<tr>
<td>TRAN</td>
<td>12:57:52.04</td>
<td>ADDPART</td>
<td>00181</td>
<td>00000000000002FF</td>
<td>source=otma_tib userid=nonusidimsid=VD1PXRF</td>
</tr>
<tr>
<td>DLI/Trace</td>
<td>000000000000030E</td>
<td>psb=DFSSAM04 pcb=DI21PART type=callfunction=ISRT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBUPD (full function)</td>
<td>DI21PART</td>
<td>dsid=00001 dl/i call=insert physicalfunction=insert</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLI/Trace</td>
<td>0000000000000312</td>
<td>pcb=DFSSAM04 pcb=DI21PART type=callfunction=ISRT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLI/Trace</td>
<td>0000000000000313</td>
<td>pcb=DFSSAM04 pcb=DI21PART type=callfunction=ISRT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRAN</td>
<td>12:57:54.24</td>
<td>DLETPART</td>
<td>00190</td>
<td>000000000000032B</td>
<td>psb=DFSSAM04 pcb=DI21PART type=callfunction=ISRT</td>
</tr>
<tr>
<td>DLI/Trace</td>
<td>000000000000032B</td>
<td>pcb=DFSSAM04 pcb=DI21PART type=callfunction=ISRT</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Overview of the Auditing report

You can use the Auditing report for purposes such as these:

- If you have been asked to provide information for an audit, this report
  summarizes the most commonly requested information in a format for auditors. It
  emphasizes the type of event that was found while providing information to
  support further research into the event.

- You can tailor the report to show information for specific users, databases,
  transactions, and so on. For this report, a transaction is any logical unit of work
  (LUOW) that contains a type-31 log record which is flagged as a DLI GET
  UNIQUE action.

Log Analyzer produces the Auditing report and writes it to the AUDIT data set
when you specify the AUDIT keyword on the REPORTS control statement. For more
information, see “AUDIT keyword” on page 115.

Information in the Auditing report

The Auditing report contains a single section that lists audit-related events.

The Auditing report provides AUDIT filtering criteria on the report header area:

- If you use AUDIT report filters
  The report header displays the FILTERING selection criteria used

- If you do not use AUDIT report filters
  The report header will display No report filters were selected
If you specify the following criteria:

\[ \text{AUDIT}=(\text{TYPE} = \text{TRAN} \; \text{DBD} = \text{ABC*},\text{XYZ*}) \]

The following report heading appears:

```
-----------------------------------------------------------------------
2015-323              Log Analyzer for IMS V1.5.00.03            Page 1
Auditing Report       (AUDIT)
FILTER=DBD=ABC*,XYZ*,TYPE=TRAN
-----------------------------------------------------------------------
```

The following table describes the fields in this report.

**Table 18: Auditing report fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>The <strong>Type</strong> field lists the type of event (such as signon, signoff, transaction, command, command type 2, or security violation). For each transaction that updated a database, each instance of a DBD and DSG combination that was updated is listed on a separate line. DLI/Trace records present are listed on a separate line.</td>
</tr>
<tr>
<td>Time</td>
<td>The <strong>Time</strong> field lists the time when the event occurred. You can use this field to find the event in other reports.</td>
</tr>
<tr>
<td>Target</td>
<td>The <strong>Target</strong> field contains event information that was captured from the log record:</td>
</tr>
<tr>
<td></td>
<td>■ For signon and signoff events, this field contains the user ID.</td>
</tr>
<tr>
<td></td>
<td>■ For transaction events, this field contains the destination (which is typically a transaction code).</td>
</tr>
<tr>
<td></td>
<td>■ For transactions that updated a database, this field contains the DBD that was updated.</td>
</tr>
<tr>
<td></td>
<td>■ For command type 2 related to the keyword value, this field contains the values of the following:</td>
</tr>
<tr>
<td></td>
<td>— Database</td>
</tr>
<tr>
<td></td>
<td>— Transaction</td>
</tr>
<tr>
<td></td>
<td>— mmlink</td>
</tr>
<tr>
<td></td>
<td>— msplink</td>
</tr>
<tr>
<td></td>
<td>— msname</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUOW#</td>
<td>The LUOW# field contains the LUOW number that Log Analyzer has assigned arbitrarily during the analysis job. You can use this field to find the event in other reports. For more information, see “Creating LUOWs” on page 281.</td>
</tr>
<tr>
<td>Log seq #</td>
<td>The Log seq # field contains the log sequence number (which is in hexadecimal format) that IMS has assigned to the log record. You can use this field to find the event in other reports or in the original log data set.</td>
</tr>
<tr>
<td>Additional data</td>
<td>The Additional data field contains event information that was captured from the log record:</td>
</tr>
<tr>
<td></td>
<td>■ For signon and signoff events, this field contains the LTERM from which the user signed on or signed off.</td>
</tr>
<tr>
<td></td>
<td>■ For transaction events, this field contains the source, the IMSID, and the user ID.</td>
</tr>
<tr>
<td></td>
<td>■ For transactions that updated a database, this field contains the DSID that was updated.</td>
</tr>
<tr>
<td></td>
<td>■ For command type 2, this field contains the verb, the keyword and, if present, the user ID.</td>
</tr>
</tbody>
</table>

### Using the LUOW Summary report

The LUOW Summary report summarizes information about the LUOWs that Log Analyzer created to organize input log records.

**Figure 44: LUOW Summary report**

<table>
<thead>
<tr>
<th>2009-110</th>
<th>Log Analyzer for IMS V1.2.01.00</th>
<th>Page 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUOW Summary (LUOWSUMM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOG time span: FROM 2007-201 08:48:55.83 TO 2007-201 09:00:38.23 DURATION 00:11:42</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUOW#</td>
<td>Origin</td>
<td>Rest.</td>
</tr>
<tr>
<td>0061079</td>
<td>NMSGBMP</td>
<td>NMSGBMP</td>
</tr>
<tr>
<td>0063631</td>
<td>——trace</td>
<td></td>
</tr>
<tr>
<td>0070737</td>
<td>ims</td>
<td>CVT144</td>
</tr>
<tr>
<td>0070971</td>
<td>——trace</td>
<td></td>
</tr>
<tr>
<td>0072800</td>
<td>appcotma</td>
<td>RLT728</td>
</tr>
<tr>
<td>0072803</td>
<td>appcotma</td>
<td>XST710</td>
</tr>
<tr>
<td>0072858</td>
<td>appcotma</td>
<td>RLT728</td>
</tr>
<tr>
<td>0072923</td>
<td>appcotma</td>
<td>XST710</td>
</tr>
<tr>
<td>0072977</td>
<td>appcotma</td>
<td>XST710</td>
</tr>
<tr>
<td>0073073</td>
<td>——trace</td>
<td></td>
</tr>
</tbody>
</table>

### Overview of the LUOW Summary report

You can use the LUOW Summary report to identify individual LUOWs for further analysis.
This information is useful for purposes such as these:

- If you are diagnosing a problem, this report presents a high-level view of the LUOWs that were active during the time interval when the problem became apparent. By selecting these LUOWs for more study, you can trace events that might have caused, contributed to, or been affected by the problem.

- If you want to examine individual LUOWs that are associated with a specific characteristic (such as an origin or a user ID), you can search the report to find each LUOW with that characteristic. Then you can note the LUOW number, which you can use to find the LUOW and its records on other reports.

Log Analyzer produces the LUOW Summary report and writes it to the LUOWSUMM data set when you specify the LUOWSUMM keyword on the REPORTS control statement. For more information, see “LUOWSUMM keyword” on page 125.

Information in the LUOW Summary report

The LUOW Summary report contains a single section listing all LUOWs that Log Analyzer created.

The entries in the LUOW Summary report are sorted by the timestamp in the 1st rec time field (even though the LUOW numbers might appear to be in sequence).

The following table lists and describes the fields on the LUOW Summary report.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUOW</td>
<td>The LUOW field contains the LUOW number that Log Analyzer has assigned arbitrarily during the analysis job. For more information, see “Creating LUOWs” on page 281.</td>
</tr>
<tr>
<td>Origin or Ori/CICSt</td>
<td>If Log Analyzer has detected CICS transaction activity, the label on this field is Ori/CICSt. If Log Analyzer has not detected CICS transaction activity, the label is Origin. For transaction-type LUOWs, the Origin field or the Ori/CICSt field contains the origin (the source, such as an LTERM) or CICS transaction name of the transaction. For a non-message-driven BMP, this field contains the value NMSGBMP. For other types of LUOWs, this field contains an eye-catcher (==&gt;&gt; ), followed by the type of the LUOW and relevant data (such as the database name for a database open or close event and user ID for sign-on and sign-off events). For more information, see “Types of LUOWs” on page 302.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| **Dest or DST/CICS** | If Log Analyzer has detected CICS transaction activity, the label on this field is **DST/CICS**. If Log Analyzer has not detected CICS transaction activity, the label is **Dest**. 
For transaction-type LUOWs, the **Dest** field or the **DST/CICS** field contains the destination or CICS application ID (applid) of the transaction. For a non-message-driven BMP, this field contains the value **NMSGBMP**. 
For a message-switching transaction, a second line is shown for the transaction to identify other destinations that were involved in the transaction. 
For other types of LUOWs, this field is blank. |
| **PSB** | For transaction-type LUOWs, the **PSB** field contains the PSB name, if applicable. 
For other types of LUOWs, this field is blank. |
| **User** | For transaction-type LUOWs, the **User** field contains the user ID of the transaction. Lowercase characters denote an element with no specifically identifiable user ID. For example, **nonusid** denotes a LUOW with a security segment (indicating that it is a transaction LUOW), but the security segment does not contain a user ID. The string **none** denotes a transaction LUOW with no security segment. For a non-message-driven BMP, this field contains the PSB. 
For other types of LUOWs, this field is blank. If the LUOW is associated with a user ID, the report displays the user ID as variable data after the label **userid**. |
| **Recs** | For transaction-type LUOWs, the **Recs** field contains a count of log records that Log Analyzer associated with the LUOW. 
For other types of LUOWs, this field is blank. Often only one log record is associated with other types of LUOW. |
| **1st rec time** | For transaction-type LUOWs, the **1st rec time** field contains the timestamp of the first log record that has been included in this LUOW. 
For other types of LUOWs, this field is blank. |
| **Elapsed** | For LUOWs with more than one record, this field contains the calculated elapsed time, which is the difference found by subtracting the timestamp of the first record in the LUOW from the timestamp of the last record in the LUOW. 
For LUOWs that contain only one record, this field is blank. 
If the **ELAPSETIME=NO** keyword was specified on the ANALYZE control statement, this field is replaced with a column that contains the TPIPE value or the LUNAME value for APPC or OTMA origins. |
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notes</td>
<td>If applicable, the Notes field contains more information about the LUOW:</td>
</tr>
<tr>
<td></td>
<td>■ <strong>ABEND</strong>=<code>code</code> indicates that one or more abnormal terminations have occurred during processing of the LUOW (<em>code</em> is the first abend code that is encountered).</td>
</tr>
<tr>
<td></td>
<td>■ <strong>DB-UPDT</strong> indicates that one or more IMS database updates were performed in the LUOW.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>DB2</strong> indicates that the LUOW contains DB2 log records.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>EXTSUB</strong> indicates that the LUOW was connected to an external subsystem. DB2 does not produce log records for read-only units of work (UOWs). Therefore, even if the LUOW was connected to a DB2 subsystem, the LUOW cannot report any DB2 log records for read-only work.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>IMS CONNECT</strong> indicates that the LUOW contains event trace records from Energizer for IMS Connect.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>log record=</strong><code>X'xxxx'</code> indicates that an origin is unknown, that an unmapped record was found, or that only one log record was associated with the LUOW. (The variable <code>xxxx</code> is the log record type and subtype.)</td>
</tr>
<tr>
<td></td>
<td>■ <strong>MULTI-IMS</strong> indicates that two or more IMS systems have logged records that are included in the LUOW.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>RO-DB</strong> indicates that read-only activity occurred against a database during the LUOW.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>RO-DB(O)</strong> indicates that read-only activity occurred against databases during the LUOW, but the activity was incompletely recorded because the region exceeded MainView’s defined maximum number of database trailers. MainView recorded the excess activity in an overflow trailer named OTHERS. To prevent this condition, increase the values of the MainView DBTS and DBTS4BMP parameters.</td>
</tr>
<tr>
<td>Additional DESTs</td>
<td>If multiple destinations are associated with an LUOW, the Additional DESTs field (shown on a separate line) identifies a maximum of six other transaction destinations. This field usually indicates that the transaction was involved with a message switch. If an LUOW has a single destination or is not a transaction LUOW, this field is not displayed.</td>
</tr>
<tr>
<td>Log Sequence Num</td>
<td>For certain LUOWs, such as those that contain unknown origins, the Log Sequence Num field (shown on a separate line) contains the log sequence number (which is in hexadecimal format) that IMS has assigned to the log record. For other LUOWs, this field is not displayed.</td>
</tr>
</tbody>
</table>
Using the LUOW Detail report

The LUOW Detail report lists key information about the individual log records that Log Analyzer has associated with an LUOW.

Figure 45: LUOW Detail report

Overview of the LUOW Detail report

You can use the LUOW Detail report to obtain in-depth information about the log records in an LUOW.

This information is useful for purposes such as these:

- If you are diagnosing a problem, you can identify a particular record for further study. By noting the log sequence number or the UOW-1 value of the record, you can find the record on other reports.

- If you have questions about whether or when a particular event or type of event has occurred for the LUOW, this report lists all events (as represented by log records) in the order in which the events occurred.
If you have questions about the flow of data within a LUOW, this report shows every event that occurred in sequence. For example, you can trace the progress of a message-switching or MSC transaction through various IMS systems.

Log Analyzer produces the LUOW Detail report and writes it to the LUOWDET data set when you specify the LUOWDETAIL keyword on the REPORTS control statement. For more information, see “LUOWDETAIL keyword” on page 123.

Information in the LUOW Detail report

The LUOW Detail report contains a section for each LUOW that Log Analyzer created.

These sections are listed in order by LUOW number:

- The first line of the section identifies the LUOW.
- The second line contains column headings for the information about log records that have been associated with the LUOW.
- The third line (and each subsequent line) provides information about a specific log record. Log records are listed in order by the record timestamp.

If input log sources contained log records from a single version of IMS, the heading of the LUOW Detail report notes the IMS version as $R = n$ (n is the version number). If input log sources contained log records from multiple versions of IMS, the LUOW Detail report contains a column indicating the IMS version that created each individual record.

The following table lists and describes the fields on the LUOW Detail report.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUOW</td>
<td>The LUOW field contains the LUOW number that Log Analyzer assigned arbitrarily during the analysis job. For more information, see “Creating LUOWs” on page 281. LUOWs on the report are listed in sequence by the value in the LUOW field.</td>
</tr>
<tr>
<td>Log recs</td>
<td>The Log recs field contains the count of log records that Log Analyzer associated with the LUOW.</td>
</tr>
<tr>
<td>Rec tkns</td>
<td>The Rec tkns field contains the count of recovery tokens that Log Analyzer found in log records that are associated with the LUOW.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UOW-1</td>
<td>The UOW-1 field contains the UOW-1 number that Log Analyzer obtained or created and assigned to this LUOW. For more information, see “LUOW numbers and UOW-1 numbers” on page 304.</td>
</tr>
<tr>
<td>Code</td>
<td>The Code field contains the two-character log record type code and (if applicable) the two-character log record subtype code that identify the type and subtype of the log record. For a list of log record type codes, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Type</td>
<td>The Type field describes the log record type. For a list of log record types, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>R</td>
<td>If input log sources contained log records from multiple versions of IMS, the R field contains the IMS version that created the log record. To make room for this column, the three high-order (left-most) digits of the Log Sequence Num field are omitted.</td>
</tr>
<tr>
<td>Log Seq Num</td>
<td>The Log Seq Num field contains the log sequence number (in hexadecimal format) that IMS assigned to the log record.</td>
</tr>
<tr>
<td>Time</td>
<td>The Time field contains the timestamp that IMS has assigned to the log record. Records for the LUOW are listed in sequence by the values in the Time field.</td>
</tr>
</tbody>
</table>
| Variable  | The Variable field can contain relevant information that Log Analyzer "snipped" (copied) from within the log record. This field can extend to multiple lines, depending on the type of record. For more information about variable data, see “Variable data snipped from log records” on page 304. If this field contains the following string, no variable data was considered to be important to snip for this type of log record: (no formatted data) Log Analyzer can include other notations in parentheses, such as (start unit-of-recovery). These notations indicate comments that Log Analyzer has added to help you interpret the flow of events. If Log Analyzer is reporting FA log record databases that were read during the LUOW, this field includes the following notations:  
  ■ readDBs (list of databases that were read)  
  ■ readDBs/upd (list of databases that were read and updated)  
  ■ insertDBs (list of databases with inserts)  
  ■ deleteDBs (list of databases with deletes)  
  ■ replaceDBs (list of databases with replaces)  
  ■ OTHERS in the readDBs list indicates that read-only activity was incompletely recorded because the region exceeded MainView’s defined maximum number of database trailers. MainView recorded the excess activity in an overflow trailer named OTHERS. To prevent this condition, increase the values of the MainView DBTS and DBTS4BMP parameters. |
### Field Description

**DATA ==>>**  
For type 01 records and type 03 records, the **DATA ==>>** field displays snipped data that Log Analyzer obtains if you have specified the LUOWDETAILDATA keyword. When you specify this keyword, Log Analyzer copies data from the first segment of the record, beginning at the specified offset within the record and continuing to the specified length. The offset is relative to 0. For more information, see “LUOWDETAILDATA keyword” on page 124.

If the specified offset is past the end of the record, the report indicates the error by displaying the following string instead of snipped data:

`*//* no data *//*`

If the specified offset is before the end of the record but the specified length is past the end of the record, the report displays as much of the snipped data as possible and indicates the condition by placing an asterisk within the field label as follows:

`DATA*==>>`

---

**Using the Record Dumps report**

The Record Dumps report displays the contents of log records in the requested format: standard or vertical.

**Figure 46: Record Dumps report in standard format**

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Log Analyzer for IMS V1.2.01.00</th>
<th>Page 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009-110</td>
<td></td>
<td>Record Dumps (DUMPDIC)</td>
<td></td>
</tr>
<tr>
<td>LOG time span: FROM 2007-201 08:48:55.83 TO 2007-201 09:00:38.23 DURATION 00:11:42</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUOW 0061079 record 0000001</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record Description=Application program start</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log sequence number=X'000000026602A2BF'</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log STCK time=20072010850555</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recovery token=IMSBXRF 0C6553F400000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Luow 0061079 record 0000002</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record Description=Fast Path database update</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log sequence number=X'000000026602C4D9'</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log STCK time=20072010850563</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recovery token=IMSBXRF 0C6553F400000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Linear dump</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Luow 0061079 record 0000003</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record Description=Fast Path database update</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log sequence number=X'000000026602C4D9'</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log STCK time=20072010850563</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recovery token=IMSBXRF 0C6553F400000000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

... data omitted ...

**Figure 47: Record Dumps report in vertical format**

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Log Analyzer for IMS V1.2.01.00</th>
<th>Page 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009-110</td>
<td></td>
<td>Record Dumps (DUMPDIC)</td>
<td></td>
</tr>
<tr>
<td>LOG time span: FROM 2007-201 08:48:55.83 TO 2007-201 09:00:38.23 DURATION 00:11:42</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUOW 0061079 record 0000001</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record Description=Application program start</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Chapter 5 Working with batch reports 195
**Overview of the Record Dumps report**

You can use the Record Dumps report to view the entire contents of log records that Log Analyzer associated with an LUOW.

This information is useful for purposes such as these:

- If you need to conduct in-depth research on log records, this report shows the data in a format that supports problem diagnosis.
- If Log Analyzer has not yet provided a map for a type of log record, this report provides a method for obtaining relevant information from this type of record.

The format of the Record Dumps report depends on the keyword that you specify on the REPORTS control statement:

- If you specify the DUMPREC keyword, Log Analyzer produces the Record Dumps report in standard format and writes it to the DUMPREC data set. For more information, see “DUMPREC keyword” on page 119.
- If you specify the DUMPVERT keyword, Log Analyzer produces the Record Dumps report in vertical format and writes it to the DUMPVERT data set. For more information, see “DUMPVERT keyword” on page 120.

**Note**

Log Analyzer must produce the LUOW Detail report to produce the Record Dumps report.

The numeric value that you can specify for the DUMPREC keyword or the DUMPVERT keyword specifies a limits on the number of lines that are produced on the Record Dumps report, not a limit on the number of log records that are dumped.
Information in the Record Dumps report

For each dumped log record, the Record Dumps report displays information in several sections:

- The heading interprets and labels selected fields from the record, such as the record type and record length.

- The Log record dump section displays each byte of the record. The format of the displayed data depends on the dump type and format options that you have selected.

- For many transaction-oriented types of records, the Application data section displays extracted application data.

The report lists the records for each LUOW in sequence according to their record field values, starting with record 1.

The following table lists and describes the fields on the Record Dumps report.

Table 21: Record Dumps report fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUOW</td>
<td>The LUOW field contains the LUOW number that Log Analyzer assigned arbitrarily during the analysis job. For more information, see “Creating LUOWs” on page 281. LUOWs on the report are listed in sequence by the value in the LUOW field.</td>
</tr>
<tr>
<td>record</td>
<td>The record field contains the sequential number that Log Analyzer assigned to indicate the sequence of this record within the LUOW.</td>
</tr>
<tr>
<td>Record description</td>
<td>The Record description field describes the type of the record. For information about record types, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Record code</td>
<td>The Record code field contains the two-byte code that identifies the type of the record. For information about record types, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Record sub-code</td>
<td>The Record sub-code field contains the two-byte subcode that identifies the subtype of the record. The report displays this field for records that have a recognizable subcode. For information about record types, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Length</td>
<td>The Length field reports the total number of bytes that are in the record. The length is reported in decimal format and hexadecimal format.</td>
</tr>
<tr>
<td>UOW-1</td>
<td>The UOW-1 field contains the UOW-1 number that Log Analyzer obtained (or created) and has assigned to this LUOW. For more information, see “LUOW numbers and UOW-1 numbers” on page 304.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Log sequence number</td>
<td>The <strong>Log sequence number</strong> field contains the log sequence number (which is in hexadecimal format) that IMS has assigned to the log record.</td>
</tr>
<tr>
<td>Log STCK time</td>
<td>The <strong>Log STCK time</strong> field contains the timestamp of the log record (as obtained from the STCK instruction).</td>
</tr>
<tr>
<td>record-specific fields</td>
<td>The report displays record-specific fields (such as <strong>Source</strong> and <strong>Dest</strong>) for certain record types. These fields contain formatted and labeled data from the record to help you with record analysis.</td>
</tr>
<tr>
<td>offset</td>
<td>The <strong>offset</strong> field, which is in the left-most column, indicates the offset of the dumped data within the log record.</td>
</tr>
<tr>
<td>data</td>
<td>Interpretation of the <strong>data</strong> field depends on the format of the report.</td>
</tr>
<tr>
<td></td>
<td>If you specified the standard format, each line of the data field shows 32 bytes of the dumped record, formatted in eight columns of eight characters with the two hexadecimal digits that represent a byte shown side by side. The character representation of the 32 bytes is shown at the end of each line.</td>
</tr>
<tr>
<td></td>
<td>If you specified the vertical format (also known as the &quot;over and under&quot; format), the left-most column indicates the offset within the record. Two, three, or four lines are used to show 64 bytes of the dumped record as follows:</td>
</tr>
<tr>
<td></td>
<td>■ The first line is a heading that shows column positions.</td>
</tr>
<tr>
<td></td>
<td>■ If you specified the DUMPFORM(BOTH) keyword, the second line shows the character representation of the byte. The third line shows the first digit of the two hexadecimal digits that represent each byte. The fourth line shows the second hexadecimal digit of each byte, directly under the first hexadecimal digit.</td>
</tr>
<tr>
<td></td>
<td>■ If you specified the DUMPFORM(CHAR) keyword, the second line shows the character representation of the byte. The hexadecimal representation is omitted.</td>
</tr>
<tr>
<td></td>
<td>■ If you specified the DUMPFORM(HEX) keyword, the second line shows the first digit of the two hexadecimal digits that represent each byte. The third line shows the second hexadecimal digit of each byte, directly under the first hexadecimal digit. The character representation is omitted.</td>
</tr>
</tbody>
</table>

**Using the Time Sequence report**

The Time Sequence report lists every log record that was selected for analysis in the order in which it was written to an IMS log.

**Figure 48: Time Sequence report**

2009-110   Log Analyzer for IMS V1.2.01.00
Time Sequence (LUWTSEQ) R=B
Log time span: FROM 2007-201 08:48:55.83 TO 2007-201 09:30:38.23 DURATION 00:11:42
Overview of the Time Sequence report

You can use the Time Sequence report to view a log record in context with other log records, which might be associated with the same or a different LUOW.

This information is useful for purposes such as these:

- If you are diagnosing a problem, this report helps you to identify other LUOWs that were active while a LUOW of interest was active. With this information, you can better understand how the LUOWs might have influenced each other.

- If you are diagnosing a problem, this report helps you to determine whether an unrelated event (that was not associated with a particular LUOW of interest) has occurred before, during, or after the LUOW was active or has not occurred at all.

Log Analyzer produces the Time Sequence report and writes it to the LUOWTSEQ data set when you specify the LUOWTSEQ keyword on the REPORTS control statement. For more information, see “LUOWTSEQ keyword” on page 125.

Information in the Time Sequence report

The information in the Time Sequence report is similar to the information in the LUOW Details report, except that the information is sorted by the time stamps of the log records instead of by LUOW number.

Each log record in the Time Sequence report is listed in sequence by its timestamp, without regard for the IMS system that created the record. Log records that were written by different IMS systems are intermingled in the report.

If input log sources contain log records from a single version of IMS, the heading of the Time Sequence report notes the IMS version as R=N (N is the version number). If input log sources contain log records from multiple versions of IMS, the Time Sequence report contains a column that indicates the IMS version which created each individual record.
The following table lists and describes the fields on the Time Sequence report.

### Table 22: Time Sequence report fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>The Code field contains the two-character log record type code and (if applicable) the two-character log subtype code that identify the type and subtype of the log record. For a list of log record type codes, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Type</td>
<td>The Type field describes the log record type.</td>
</tr>
<tr>
<td>R</td>
<td>If input log sources contained log records from multiple versions of IMS, the R field contains the IMS version that created the log record. To make room for this column, the three high-order (left-most) digits of the Log Sequence num field are omitted.</td>
</tr>
<tr>
<td>Log Seq Num</td>
<td>The Log Seq Num field contains the log sequence number (in hexadecimal format) that IMS assigned to the log record.</td>
</tr>
<tr>
<td>LUOW</td>
<td>The LUOW field contains the LUOW number that Log Analyzer assigned arbitrarily during the analysis job. For more information, see “Creating LUOWs” on page 281. LUOWs on the report are listed in sequence by the value in the LUOW field.</td>
</tr>
<tr>
<td>Variable</td>
<td>The Variable field contains relevant information that Log Analyzer snipped from within the log record. For more information about variable data, see “Variable data snipped from log records” on page 304. If this field contains the string (\text{no formatted data}), no variable data was considered to be important to snip for this type of log record.</td>
</tr>
</tbody>
</table>

## Using the Unresolved Orphans report

The Unresolved Orphans report provides information about records for which Log Analyzer was unable to resolve associations and relate the record to an LUOW.

### Figure 49: Unresolved Orphans report

2009-111  Log Analyzer for IMS V1.2.01.00  Page 1

Unresolved Orphans (ORPHAN)  R=8

Recovery Token=IMSBXRF 0000000D  # recs=0000006

<table>
<thead>
<tr>
<th>Code</th>
<th>Log Sequence Num</th>
<th>Variable</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>37</td>
<td>000000026605303D06</td>
<td>(no formatted data)</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>000000026605300D07</td>
<td>(no formatted data)</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>000000026605300D08</td>
<td>(no formatted data)</td>
<td></td>
</tr>
</tbody>
</table>

Recovery Token=IMSBXRF 0000000D  # recs=0000006

<table>
<thead>
<tr>
<th>Code</th>
<th>Log Sequence Num</th>
<th>Variable</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>37</td>
<td>000000001661C8C67</td>
<td>(no formatted data)</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>00000002661C8CB1</td>
<td>(no formatted data)</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>00000002661C8CB2</td>
<td>(no formatted data)</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>0000000266531258</td>
<td>(no formatted data)</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>000000026653125C</td>
<td>(no formatted data)</td>
<td></td>
</tr>
</tbody>
</table>

200  BMC Log Analyzer for IMS User Guide
Overview of the Unresolved Orphans report

You can use the Unresolved Orphans report to obtain information that can help you and BMC Customer Support to determine whether an unresolved association represents a problem and, if so, how the problem can be corrected.

Log Analyzer produces the Unresolved Orphans report and writes it to the ORPHAN data set when all of the following conditions exist:

■ You specify the ORPHANS keyword on the REPORTS control statement.
■ You request reports other than the Analysis Summary report.
■ One or more unresolved associations remain at the end of the analysis job.

For more information, see “ORPHANS keyword” on page 126.

Information in the Unresolved Orphans report

The Unresolved Orphans report contains one set of fields for each unresolved association, as described in the following table.

Table 23: Unresolved Orphans report fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recovery token</td>
<td>The Recovery token field contains the value that Log Analyzer tried to use as a mechanism for resolving associations, as explained in “Unresolved associations when creating LUOWs” on page 282.</td>
</tr>
<tr>
<td># recs</td>
<td>The # recs field provides a count of records that contain the indicated recovery token.</td>
</tr>
<tr>
<td>Code</td>
<td>The Code field indicates the log record type code and the subtype code of each record for which an association could not be resolved.</td>
</tr>
<tr>
<td>Log Sequence Num</td>
<td>The Log Sequence Num field contains the log sequence number (which is in hexadecimal format) that IMS assigned to the log record.</td>
</tr>
<tr>
<td>Variable</td>
<td>The Variable field contains relevant information that Log Analyzer snipped from within the log record. For more information about variable data, see “Variable data snipped from log records” on page 304. If this field contains the string (no formatted data), no variable data was considered to be important to snip for this type of log record.</td>
</tr>
</tbody>
</table>
Using the ABEND report

The ABEND report provides information about transactions or programs that have terminated abnormally.

Figure 50: ABEND report

<table>
<thead>
<tr>
<th>ABEND report</th>
<th>R=10</th>
</tr>
</thead>
<tbody>
<tr>
<td>--PSB--</td>
<td>--Tran--</td>
</tr>
<tr>
<td>AEP210</td>
<td>AET210</td>
</tr>
<tr>
<td>VIP085</td>
<td>VIT0851</td>
</tr>
<tr>
<td>VIP085</td>
<td>VIT0855</td>
</tr>
<tr>
<td>VIP226</td>
<td>VIT226</td>
</tr>
<tr>
<td>WCP323</td>
<td>WCT323</td>
</tr>
<tr>
<td>SPP202</td>
<td>SP202</td>
</tr>
</tbody>
</table>

Overview of the ABEND report

You can use the ABEND report to identify all transactions and programs that have terminated abnormally.

To produce this report, Log Analyzer focuses exclusively on type 07 records.

The ABEND report does not replace the report that Log Analyzer produces when you specify the SELECT=ABENDS=YES filter. With that filter, the LUOWDETAIL report displays all 07 records that are flagged as abending within the context of a logical unit of work (LUOW). The ABEND report displays the same 07 records without the LUOW context, producing a streamlined view of the information.

Log Analyzer produces the ABEND report and writes it to the ABEND data set when you specify the ABEND keyword on the REPORTS control statement. For more information, see “ABEND keyword” on page 110.

Information in the ABEND report

The ABEND report contains a single section that lists information from the type 07 records that Log Analyzer found in the input log record sources.

The following table identifies the fields in this report.
Table 24: ABEND report fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSB</td>
<td>The PSB field contains the program specification block (PSB) name of the program that was executing when the abend occurred.</td>
</tr>
<tr>
<td>Tran</td>
<td>The Tran field contains the transaction ID of the transaction that was executing when the abend occurred.</td>
</tr>
</tbody>
</table>
| Type        | The Type field identifies the type of entity that terminated abnormally, as indicated in the type 07 log record. The following entity types are possible:  
  ■ MPP  
  ■ BMP  
  ■ UOR  
  ■ JAVA  
  ■ CPI-C  
  ■ DBCTL  
  ■ QUICK  
  ■ RESOLVE |
| Userid      | The Userid field identifies the TSO user ID that was associated with the program or transaction that was executing when the abend occurred. |
| Jobname     | The Jobname field identifies the JES job name that was assigned to the program that was executing when the abend occurred. |
| Abend       | The Abend field identifies the abend code that was issued. |
| Abend date/time | The Abend date/time field indicates the Julian date (in yyyy.ddd format) and time (in hh:mm:ss.t format) when the abend occurred. |
| Execution time | The Execution time field indicates the length of time (in hh:mm:ss.t format) that the job had been executing when the abend occurred. |
| Total Msgs  | The Total Msgs field indicates the number of messages that the program had processed before the abend occurred. |
| False Schedule | The False Schedule field indicates whether the transaction was a false schedule. |

Using the Deadlocks report

The Deadlocks report provides information about record lockout situations, also known as a deadly embrace between two or more transactions.

In this situation, a transaction locks record A and attempts to lock record B. However, a second transaction has already locked record B and is attempting to lock record A. Because neither transaction can continue until the other transaction
releases its locks, IMS typically issues a user 777 or 123 pseudo abend for one of the transactions.

**Figure 51: Deadlocks report**

<table>
<thead>
<tr>
<th>2009-048</th>
<th>Log Analyzer for IMS V1.2.00.03</th>
<th>Page 1</th>
</tr>
</thead>
</table>

Deadlocks (DEADLOCK) R=A

**DMB name=DI21PART**  Lock name=814DF67DD800301D7

key is root key for lock

Key Dump:

00000000   00000000   00000000   00000000
Status IMS Tran-Job PSB name PCB-DBD PST# LUOW# Rgn Call Lock Func State
Waiter PRJ  IRLM001  IRLM001  DI21PART 00002  unknown   MPP GET GRIDX 30400358 06-P   Victim
Holder PRJ  IRLM002  IRLM002 00001 unknown   MPP   06-P

**DMB name=IVPDB1**  Lock name=00001D4A800401D7

key is root key for lock

Key Dump:

00000000   00000000   00000000   00000000
Status IMS Tran-Job PSB name PCB-DBD PST# LUOW# Rgn Call Lock Func State
Waiter PRAI IRLM2002 IRLM2002 IVPDB1 00002 0000112  MPP GET GRIDX 30400358 06-P   Victim

Overview of the Deadlocks report

You can use the Deadlocks report for purposes such as these:
- Identify LUOWs that were involved in a deadlock.
- Analyze specific deadlocks to determine causes and effects.
- Analyze the general frequency and severity of deadlocks in an environment.

Log Analyzer produces the Deadlocks report and writes it to the DEADLOCK data set when you specify the DEADLOCK keyword on the REPORTS control statement. For more information, see “DEADLOCK keyword” on page 118.

204  BMC Log Analyzer for IMS User Guide
Information in the Deadlocks report

The Deadlocks report is similar to the report that is produced by the IMS Record Format and Print Module (DFSERA30) exit routine of the File Select and Formatting Print Utility (DFSERA10).

The primary differences follow:

- Log Analyzer identifies the logical unit of work (LUOW) for each participant in the deadlock. You can perform further research on these LUOWs to obtain a complete understanding of the situation.

- Log Analyzer sorts the list of the participants in the deadlock to indicate the order in which they requested the locks. The abended transaction (the "victim") is first in the list, and the other participants are listed in cascade order. (The first transaction was waiting for the second transaction in the list; the second transaction is waiting for the third transaction, and so on.) If data for the lock holder is unavailable, Log Analyzer cannot sort the list; in this case, the report contains the message **entries not sorted** after the # participants field.

The Deadlocks report contains a single section that lists information about detected deadlocks. Each deadlock is listed with all transactions (LUOWs) that were involved in the deadlock.

The following table identifies the fields in this report; for details about the data that these fields contain, see the IMS documentation for DFSERA30.

Table 25: Deadlocks report fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deadlock #</td>
<td>The Deadlock # field contains a sequential number that Log Analyzer assigned during the analysis job to identify this deadlock in this report.</td>
</tr>
<tr>
<td># participants</td>
<td>The # participants field contains the number of participants that this deadlock involves.</td>
</tr>
<tr>
<td>DMB name</td>
<td>The DMB name field contains the name of the data management block. This block describes the database that the transaction was attempting to access during this deadlock.</td>
</tr>
<tr>
<td>Lock name</td>
<td>The Lock name field contains descriptive information that identifies the lock, including the relative byte address (RBA) of the lock and the type (full function or DEDB) of database.</td>
</tr>
<tr>
<td>key information</td>
<td>If the root key of the locked database record is known, the key information field contains descriptive information about the key.</td>
</tr>
<tr>
<td>Key Dump</td>
<td>If the root key of the locked database record is known, the Key Dump field contains the record key in hexadecimal format, followed by the record key in EBCDIC format.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Status</td>
<td>For each participant in the deadlock, the <strong>Status</strong> field indicates whether the participant was a <em>Waiter</em> (waiting to obtain a lock on the record) or the <em>Holder</em> (holding the lock on the record).</td>
</tr>
<tr>
<td>IMS</td>
<td>For each participant in the deadlock, the <strong>IMS</strong> field identifies the IMS system (or external subsystem) that was executing the transaction.</td>
</tr>
<tr>
<td>Tran-Job</td>
<td>For each participant in the deadlock, the <strong>Tran-Job</strong> field contains the name of the transaction or job.</td>
</tr>
<tr>
<td>PSB name</td>
<td>For each participant in the deadlock, the <strong>PSB name</strong> field contains the name of the program specification block that the transaction was using to access the database.</td>
</tr>
<tr>
<td>PCB-DBD</td>
<td>For each waiter in the deadlock, the <strong>PCB-DBD</strong> field contains the name of the program communication block that the transaction was using to access the database.</td>
</tr>
<tr>
<td>PST#</td>
<td>For each participant in the deadlock, the <strong>PST#</strong> field contains the number of the partition specification table that describes the dependent region.</td>
</tr>
<tr>
<td>LUOW#</td>
<td>For each participant in the deadlock, the <strong>LUOW#</strong> field contains the LUOW number to which Log Analyzer assigned this transaction during the analysis job. You can use this field to find the LUOW in other reports.</td>
</tr>
<tr>
<td>Rgn</td>
<td>For each participant in the deadlock, the <strong>Rgn</strong> field identifies the type of IMS control region that was executing the transaction.</td>
</tr>
<tr>
<td>Call</td>
<td>For each waiter in the deadlock, the <strong>Call</strong> field identifies the DL/I call that the application issued for the database record. The report cannot identify the specific function for a get-type or insert-type call because this information is not available in the log records.</td>
</tr>
<tr>
<td>Lock</td>
<td>For each waiter in the deadlock, the <strong>Lock</strong> field contains a translated representation of the lock request function.</td>
</tr>
<tr>
<td>Func</td>
<td>For each Waiter in the deadlock, the <strong>Func</strong> field contains the hexadecimal lock request function (as mapped by the LRHPARM DSECT).</td>
</tr>
<tr>
<td>State</td>
<td>For each participant in the deadlock, the <strong>State</strong> field identifies the type or level of the lock.</td>
</tr>
</tbody>
</table>

**Using the Application Checkpoint report**

The Application Checkpoint report provides information about basic and extended checkpoints that are issued by batch message processing (BMP) programs.

**Figure 52: Application Checkpoint report**
Application checkpoints indicate the completion of a unit of work. When a BMP issues a checkpoint, participating systems commit changes to the records that the BMP has updated. Committing the changes releases the locks on the updated records, making those records available for use by other programs. Tuning the frequency of checkpoints helps to ensure optimum data availability, application performance, and resource usage.

Overview of the Application Checkpoint report

You can use the Application Checkpoint report for purposes such as these:

- Identify BMP jobs that are issuing application checkpoints too often or not often enough.

- Identify candidate BMP jobs for tuning checkpoint frequencies and for implementing checkpoint-pacing services of the APPLICATION RESTART Control (AR/CTL) product from BMC.

- Evaluate efforts to tune the frequency of checkpoints or implement checkpoint pacing.

For more information, see “Obtaining information about application checkpoints” on page 323.

Log Analyzer produces the Application Checkpoint report and writes it to the APPCHECK data set when you specify the APPCHECK keyword on the REPORTS control statement. For more information, see “APPCHECK keyword” on page 111.

Information in the Application Checkpoint report

The Application Checkpoint report contains a single section that lists information about included BMP jobs.

The following table identifies the fields in this report. The report lists BMP jobs in order by the timestamp of the first record that Log Analyzer found in the input logs for each BMP job.
### Table 26: Application Checkpoint report fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>JOB</strong></td>
<td>The <strong>JOB</strong> field contains the name of the job that was submitted to run the BMP program. If a BMP job is included but does not issue any checkpoints and does not update any databases, it is not listed in the report.</td>
</tr>
<tr>
<td><strong>PSB</strong></td>
<td>The <strong>PSB</strong> field contains the name of the program specification block (PSB) through which the BMP program works with database records.</td>
</tr>
<tr>
<td><strong>LUOW#</strong></td>
<td>The <strong>LUOW#</strong> field contains the number that identifies the logical unit of work (LUOW) to which Log Analyzer has assigned this BMP job. This LUOW represents all of the work that the BMP job performed, as captured in the input logs.</td>
</tr>
<tr>
<td><strong>#CHKPTS/TYYPE</strong></td>
<td>The <strong>#CHKPTS/TYYPE</strong> field contains the number of checkpoint calls that this BMP job issued, followed by a string to identify the type of checkpoint call:</td>
</tr>
<tr>
<td></td>
<td>■ <strong>SIMPLE</strong>&lt;br&gt;The BMP job issued simple (basic) checkpoint calls, which are represented by type 41 log records.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>SYMBOLIC</strong>&lt;br&gt;The BMP job issued symbolic (extended) checkpoint calls, which are represented by type 41 and type 18 log records.</td>
</tr>
<tr>
<td><strong>JOB DURATION</strong></td>
<td>The <strong>JOB DURATION</strong> field indicates the amount of time (in hours, minutes, and seconds) that elapsed while this BMP job was executing. Log Analyzer calculates this time by subtracting the timestamp of the first record that was found in the logs for the BMP job from the timestamp of the last record for the BMP job. <strong>Note:</strong> If the logs do not contain all records from the execution, this value might not be the true duration of the execution.</td>
</tr>
</tbody>
</table>
### Field Description

| CHECKPOINT FREQUENCY/MIN /SEC | The CHECKPOINT FREQUENCY/MIN /SEC field indicates how often the BMP job issued checkpoints. Log Analyzer calculates this frequency by dividing the number of checkpoints issued by the duration of the BMP job. If you specified the APPCHECK=ALL keyword, Log Analyzer lists the frequency in the appropriate column as follows:  

- **/SEC**  
  The BMP job issued more than one checkpoint per second.  

- **/MIN**  
  The BMP job issued less than one checkpoint per second.  

If you specified the APPCHECK keyword with the THRESH subkeyword, Log Analyzer lists the frequency in the column that you specified with this subkeyword.  

If you specified the APPCHECK keyword with other subkeywords but you omitted the THRESH subkeyword, Log Analyzer lists the frequency in the /MIN column. |

| Exceptions | If you did not specify a checkpoint frequency filter (that is, you omitted the CHKFREQ subkeyword), the Exceptions field can flag the following conditions as applicable:  

- **more than 1 chkp / sec**  
  The BMP job issued checkpoints at an average frequency that was greater than one checkpoint every second.  

- **less than 4 chkps / min**  
  The BMP job issued checkpoints at an average frequency that was less than four checkpoints every minute.  

- **no checkpoints**  
  The BMP job updated database records but did not issue any checkpoints.  

If you specified the CHKFREQ subkeyword, these exception flags are suppressed. |

---

### Using the RBA Buffer Statistics report

The RBA Buffer Statistics report shows the distribution of updated relative byte addresses (RBAs) or block numbers in full-function IMS databases. (The report does not provide information about Fast Path databases.)
Overview of the RBA Buffer Statistics report

You can use the RBA Buffer Statistics report to tune buffers for the EXTENDED BUFFER MANAGER (XBM) Snapshot Copy function.

Log Analyzer produces the RBA Buffer Statistics report and writes it to the RBASTATS data set when you specify the RBASTATS keyword on the REPORTS control statement. For more information, see “RBASTATS keyword” on page 127.

Information in the RBA Buffer Statistics report

If you specified RBASTATS=ALL for the analysis, the RBA Buffer Statistics report contains three columns of information for each database data set group that was updated during the time span of the analysis.

Each column contains the fields described in the following table. If you specified RBASTATS=TOTALS, the report omits these fields.

Table 27: RBA Buffer Statistics report fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBD</td>
<td>Contains the DBD name of the database that was updated</td>
</tr>
<tr>
<td>DS</td>
<td>Contains the ID of the data set group that was updated</td>
</tr>
<tr>
<td>RBA/BLK#</td>
<td>Identifies the RBA or block that was updated</td>
</tr>
<tr>
<td>CNT</td>
<td>Indicates how many times the RBA or block was updated during the time span of the analysis</td>
</tr>
</tbody>
</table>

After each set of columns for a database data set group, the Total this DBD/DSG field indicates the total number of unique RBAs that were updated during the time span of the analysis.
You can view information about LUOWs and their associated log records through the Log Analyzer ISPF interface.

This information is stored in an index file that a Log Analyzer analysis job produces if you specify the INDEXFILE control statement in the JCL for executing the job.

Note

If a Log Analyzer analysis job did not produce an index file, reports for that job are not available through the ISPF interface.

The analysis job requires you to provide an EXTRACT control statement if you provide an INDEXFILE control statement, but the extract file is not required for use of the ISPF interface. If this file is not available, interface actions that require access to the extract file (such as displaying a dump of a log record) are not available.

The following table lists major types of report information that is available in the ISPF interface.

<table>
<thead>
<tr>
<th>Type of information</th>
<th>ISPF panel</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Facts about the overall results of the analysis job</td>
<td>Report Index File Summary</td>
<td>“Obtaining summary information” on page 229</td>
</tr>
<tr>
<td>List of LUOW that were created</td>
<td>LUOW List</td>
<td>“Listing LUOWs” on page 216</td>
</tr>
<tr>
<td>Type of information</td>
<td>ISPF panel</td>
<td>Topic</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Summary information about LUOWs</td>
<td>LUOW List</td>
<td>“Listing LUOWs” on page 216</td>
</tr>
<tr>
<td></td>
<td>Destination List</td>
<td>“Listing destinations” on page 220</td>
</tr>
<tr>
<td></td>
<td>Origin List</td>
<td>“Listing origins” on page 221</td>
</tr>
<tr>
<td></td>
<td>Userid List</td>
<td>“Listing user IDs” on page 222</td>
</tr>
<tr>
<td></td>
<td>PSB List</td>
<td>“Listing PSBs” on page 223</td>
</tr>
<tr>
<td></td>
<td>CICS Transaction List</td>
<td>“Listing CICS transactions” on page 224</td>
</tr>
<tr>
<td></td>
<td>Other LUOWs List</td>
<td>“Listing other LUOWs” on page 225</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“Listing LUOWs” on page 216</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“Listing destinations” on page 220</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“Listing origins” on page 221</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“Listing user IDs” on page 222</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“Listing PSBs” on page 223</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“Listing CICS transactions” on page 224</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“Listing other LUOWs” on page 225</td>
</tr>
<tr>
<td>Detailed information about an LUOW</td>
<td>LUOW luowid</td>
<td>“Obtaining LUOW details” on page 235</td>
</tr>
<tr>
<td>List of log records that are associated with an LUOW and summary information about those records</td>
<td>LUOW luowid Log Records</td>
<td>“Listing log records for an LUOW” on page 237</td>
</tr>
<tr>
<td>Detailed information about a log record</td>
<td>Log Record Detail</td>
<td>“Obtaining log record details” on page 240</td>
</tr>
<tr>
<td>Dump of a log record</td>
<td>Log Record logrecnum</td>
<td>“Dumping log records” on page 245</td>
</tr>
<tr>
<td>List of log records in context with other log records in the same time frame</td>
<td>Log Records - Time Sequence</td>
<td>“Listing adjacent log records” on page 242</td>
</tr>
<tr>
<td>List of deadlock victims and details about participants in a deadlock</td>
<td>Deadlock Victims List</td>
<td>“Analyzing deadlock information” on page 226</td>
</tr>
<tr>
<td></td>
<td>Deadlock Detail</td>
<td></td>
</tr>
</tbody>
</table>

**Tip**

Commands that enable easier navigation between records have been assigned to function keys for many of the panels in the Analyze portion of the ISPF interface. You might want to use the PFSHOW ON command to display these definitions until you become familiar with them.

---

**Accessing reports in the ISPF interface**

To access reports that were produced during an analysis job, use the Report Index File Analysis panel. On this panel, you select a report to view and provide the name of the index file that contains report information.
To access and work with the Report Index File Analysis panel

1. Access the Log Analyzer ISPF interface.
   
   For instructions, see “Accessing the ISPF interface” on page 46.

2. On the Log Analyzer Main Menu, enter 2 in the Log Analyzer option field.
   
   The Report Index File Analysis panel is displayed.

   **Figure 54: Report Index File Analysis panel**

   ![Report Index File Analysis panel](image)

3. In the Analysis action field, enter one of the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>View a list of LUOWs. If you select this option, you can use fields at the bottom of the panel to limit the list of LUOWs by origin, destination, user ID, or any combination of these elements.</td>
</tr>
<tr>
<td>2</td>
<td>View the names of all transactions and destinations that are associated with one or more LUOWs. The Destination List panel is displayed. For more information, see “Listing destinations” on page 220.</td>
</tr>
<tr>
<td>3</td>
<td>View the names of all origins that are associated with one or more LUOWs. The Origin List panel is displayed. For more information, see “Listing origins” on page 221.</td>
</tr>
<tr>
<td>4</td>
<td>View all user IDs that are associated with one or more LUOWs. The Userid List panel is displayed. For more information, see “Listing user IDs” on page 222.</td>
</tr>
<tr>
<td>Value</td>
<td>Result</td>
</tr>
<tr>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>5</td>
<td>View the names of all program specification blocks (PSBs) that are associated with one or more LUOWs. The PSB List panel is displayed. For more information, see “Listing PSBs” on page 223.</td>
</tr>
<tr>
<td>6</td>
<td>View the names of all CICS transactions that are associated with one or more LUOWs. The CICSTRAN List panel is displayed. For more information, see “Listing CICS transactions” on page 224.</td>
</tr>
<tr>
<td>7</td>
<td>View all types of nontransaction LUOWs and other LUOWs. The Other LUOWs panel is displayed. For more information, see “Listing other LUOWs” on page 225.</td>
</tr>
<tr>
<td>8</td>
<td>View summary information about the analysis job. The Report Index File Summary panel is displayed. For more information, see “Obtaining summary information” on page 229.</td>
</tr>
</tbody>
</table>
| 9     | View a list of deadlock victims. The Deadlock Victims List panel is displayed. For more information, see “Analyzing deadlock information” on page 226.  
**Note:** The ISPF interface can display deadlock information only if the REPORTS control statement specified the appropriate keywords when the index file was created. |

4. In the **Report index file** field, enter the data set name of the index file that you want to view.

The index file is created during the analysis job. You specified this data set name as the value of the DSN keyword on the INDEXFILE control statement.

**Note**
The version of Log Analyzer that created the specified index file must be compatible with the version of the Log Analyzer ISPF interface. Otherwise, the interface displays an error and gives you the option of viewing control statements that generated the specified index file; you can use those statements to run a new analysis job and re-create the index file.

The **Report index file** field on subsequent panels contains the data set name that you enter in this field.

5. If you enter 1 (LUOWs) in **Analysis action** field, you can enter values in the following fields to filter the list of LUOWs:
   - Origin(s)
   - Tran(s)/LTERM(s)
   - Userid(s)
   - PSB(s)
   - CICS trans(s)
   - Limit to LUOWs that have abended

For more information about these fields, see “Filtering LUOWs” on page 215.
Filtering LUOWs

You can filter a list of LUOWs by one or more specific origins, destinations, user IDs, or any combination of these elements.

Filtering helps you reduce the quantity of LUOWs that you must view and helps you narrow your focus as you analyze LUOWs interactively. To filter LUOWs, use the Filter LUOWs pop-up window.

**Figure 55: Filter LUOWs pop-up window**

<table>
<thead>
<tr>
<th>Command ===</th>
<th>Filter LUOWs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter information below to limit the list of logical units of work. Leave blank for all LUOWs. Then press Enter.</td>
<td></td>
</tr>
<tr>
<td>Origin(s)...</td>
<td>Tran(s)/LTERM(s)</td>
</tr>
<tr>
<td>UserId(s)...</td>
<td>PSB(s)</td>
</tr>
<tr>
<td>CICS trans(s)</td>
<td>Additional limitations. Select if desired.</td>
</tr>
<tr>
<td><em>Limit to LUOWs that have abended</em></td>
<td>F1=Help F3=Exit F12=Cancel</td>
</tr>
</tbody>
</table>

Filtering in the ISPF interface is temporary (filters are not saved). You can specify a new filter to view LUOWs that were hidden by a previous filter.

If you do not want to filter the list of LUOWs, leave all fields blank.

**To filter LUOWs**

1. To filter LUOWs, use either of the following methods:
   - On the Report Index File Analysis panel (see “Accessing reports in the ISPF interface” on page 212), enter 1 in the Analysis action field. Enter filter values in the fields under the LUOW filter heading, as described in the following steps.
   - On the LUOW List panel, enter the FILTER command. (Type FILTER in the command area and press Enter. Alternatively, place the cursor on the Filter action in the action bar at the top of the panel, and press Enter.) The Filter LUOWs pop-up window is displayed. Enter filter values in the fields in this window, as described in the following steps.

   **Note**

   To view all LUOWs (that is, to use no filter), enter blanks in all filter fields.

2. *(optional)* In the Origin(s) field, enter the name of an origin of interest.
You can enter multiple names; use a comma to separate each name from the next name. If you do not want to filter by origin, enter blanks.

3 (optional) In the Dest(s)/Trans(s) field, enter the name of a destination (such as an LTERM) of interest.

You can enter multiple names; use a comma to separate each name from the next name. If you do not want to filter by destination, enter blanks.

For a non-message-driven BMP, you can enter NMSGBMP in this field.

4 (optional) In the Userid(s) field, enter the name of a user ID of interest.

You can enter multiple user IDs; use a comma to separate each user ID from the next user ID. If you do not want to filter by user ID, enter blanks.

For a non-message-driven BMP, you can enter NMSGBMP in this field.

5 (optional) In the PSB(s) field, enter the name of PSB of interest.

You can enter multiple names; use a comma to separate each name from the next name. If you do not want to filter by PSB, enter blanks.

6 (optional) In the CICS tran(s) field, enter the name of a CICS transaction of interest.

You can enter multiple names; use a comma to separate each name from the next name. If you do not want to filter by CICS transaction, enter blanks.

7 (optional) In the Limit to LUOWs that have abended field, enter / or s to select only LUOWs that have terminated abnormally.

If you do not want to filter by abend status, enter blanks.

8 Press Enter.

The LUOW List panel is displayed. LUOWs with log records that match the specified filter are displayed. LUOWs with no matching records are hidden. For more information, see “Listing LUOWs” on page 216.

Listing LUOWs

To view a list of LUOWs that were created during an analysis job, use the LUOW List panel.
This panel lists LUOWs that match the specified report filter (or that are associated with a selected element), provides summary information about each LUOW, and provides access to detailed information about the LUOW and the log records that are associated with it.

**To access and work with the LUOW List panel**

1. Access the Report Index File Analysis panel.
   
   For instructions, see “Accessing reports in the ISPF interface” on page 212.

2. In the Analysis option field, enter 1.

   The LUOW List panel is displayed:

   **Figure 56: LUOW List panel**

   ```plaintext
   File Filter Sort Options Help
   Log Analyzer LUOW List Row 61 of 3041
   Command ===> Scroll ===> CSR
   Report index file: 'LXM.LUI.INDEX1'
   Type one or more action codes. Enter LEFT or RIGHT for additional columns.
   S=LUOW detail L=List log records Sorted by: NOTE
   Tran/ A LUOW Id   Origin   LTERM    Userid        Timestamp           Elapsed Time
   F1=Help    F3=Exit    F7=Up      F8=Down   F10=Left   F11=Right  F12=Cancel
   _ 00285221 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:04:18.592553          .261977
   _ 00286372 ACIP0D59 FSTB0IPU NBKGBB8  2008.177 12:04:23.078691          .044153
   _ 00290884 ACIP0D5A FSTF0DSU NBKGBB8  2008.177 12:04:40.605770          .067716
   _ 00292204 ACIP0D59 FSTB0IPU NBKGBB8  2008.177 12:04:45.529540          .061072
   _ 00293645 ACIP0D59 FSTB0ACW NBKGBB8  2008.177 12:04:51.223203          .055094
   _ 00299790 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:05:15.957114          .043460
   _ 00317664 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:06:24.889338          .060394
   _ 00318256 ACIP0D5A FSTB0ACW NBKGBB8  2008.177 12:06:27.106778          .344560
   _ 00318868 ACIP0D5A FSTB0APR NBKGBB8  2008.177 12:06:29.541109          .048625
   _ 00321164 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:06:38.766389          .023029
   _ 00325297 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:07:19.557342          .066726
   _ 00331713 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:08:37.683460          .075104
   _ 00340791 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:08:37.683460          .075104
   _ 00347224 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:08:37.683460          .075104
   _ 00352145 ACIP0D5A FSTB0IPU NBKGBB8  2008.177 12:08:37.683460          .075104
   F1=Help    F3=Exit    F7=Up      F8=Down   F10=Left   F11=Right  F12=Cancel
   ```

   3. View summary information about LUOWs in the following fields.

   To toggle between views that display fields that are not visible in the current view, use the LEFT command (F10) and the RIGHT command (F11).

   **Field** | **Description**
   --- | ---
   LUOW Id | The **LUOW Id** field contains the LUOW number that Log Analyzer has assigned arbitrarily during the analysis job. For more information, see “Creating LUOWs” on page 281.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin</td>
<td>For transaction-type LUOWs, the Origin field contains the origin (such as an LTERM) of the transaction. For other types of LUOWs, this field contains an eye-catcher ( &gt;&gt; ), followed by the type of the LUOW and relevant data (such as the database name for a database open or close event). For more information, see “Types of LUOWs” on page 302.</td>
</tr>
<tr>
<td>Tran/LTERM</td>
<td>For transaction-type LUOWs, the Tran/LTERM field contains the destination (such as an LTERM) of the transaction. For other types of LUOWs, this field is blank.</td>
</tr>
<tr>
<td>Userid</td>
<td>For transaction-type LUOWs, the Userid field contains the user ID that is associated with the transaction. For other types of LUOWs, this field is blank.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>The Timestamp field contains the timestamp of the first log record that was included in this LUOW. This field is displayed in the default view of the panel.</td>
</tr>
<tr>
<td>Elapsed Time</td>
<td>The Elapsed Time field contains the calculated elapsed time of the LUOW. Log Analyzer calculates this value by subtracting the timestamp of the first log record that was included in the LUOW from the timestamp of the last log record that was included in the LUOW. This field is displayed in the default view of the panel. If the ELAPSETIME=NO keyword is specified on the ANALYZE control statement, this field is not displayed.</td>
</tr>
<tr>
<td>Message Text (Offset n)</td>
<td>The Message Text field contains a portion of the data from the first type 01 or type 03 record in the LUOW. You can use the Offset field to control which part of the 01 or 03 record data is displayed. Specify the starting position of the data within the log record. The offset is relative to 0; if you specify 1, the starting position is the second byte of the segment. The field displays 40 bytes of data at a time. You can use this field to simulate a horizontal scrolling action.</td>
</tr>
<tr>
<td>Log Records</td>
<td>The Log Records field contains a count of log records that Log Analyzer associated with the LUOW.</td>
</tr>
</tbody>
</table>
Field | Description
---|---
Note | If applicable, the **Note** field contains more information about the LUOW:
- **ABEND=** `code` indicates that one or more abnormal terminations have occurred during processing of the LUOW (`code` is the first abend code that is encountered).
- **DB2** indicates that the LUOW contains DB2 log records.
- **EXTSUB** indicates that the LUOW was connected to an external subsystem. DB2 does not produce log records for read-only units of work (UOWs). Therefore, even if the LUOW was connected to a DB2 subsystem, the LUOW cannot report any DB2 log records for read-only work.
- **IMS CONNECT** indicates that the LUOW contains event trace records for Energizer for IMS Connect.
- **MULTI-IMS** indicates that two or more IMS systems have logged records that are included in the LUOW.
- **READ** indicates that one or more IMS databases were read but not updated in the LUOW.
- **UPDT** indicates that one or more IMS databases were updated in the LUOW.

This field is displayed in the alternate view of the panel.

4 *(optional)* In the **A** (action) field to the left of an LUOW, enter one of the following action codes:

<table>
<thead>
<tr>
<th>Code</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>S</strong></td>
<td>View detailed information about the LUOW. The LUOW <code>luowid</code> panel is displayed. For more information, see “Obtaining LUOW details” on page 235.</td>
</tr>
<tr>
<td><strong>L</strong></td>
<td>View a list of all log records that are associated with the LUOW. The LUOW <code>luowid</code> Log Records panel is displayed. For more information, see “Listing log records for an LUOW” on page 237.</td>
</tr>
</tbody>
</table>

5 *(optional)* To specify a report filter (or change a filter that you already specified), you can enter the **FILTER** command.

For more information, see “Filtering LUOWs” on page 215.

6 *(optional)* To sort the data on a different field, you can enter the **SORT** command.

For more information, see “Sorting listed items” on page 232.
Listing destinations

To view the names of all destinations (transactions and LTERMs) that are associated with one or more LUOWs, use the Destination List panel. This panel displays summary information about the destinations and provides access to LUOWs that are associated with a selected destination.

To access and work with the Destination List panel

1. Access the Report Index File Analysis panel.

   For instructions, see “Accessing reports in the ISPF interface” on page 212.

2. In the Analysis option field, enter 2.

   The Destination List panel is displayed:

   **Figure 57: Destination List panel**

<table>
<thead>
<tr>
<th>File</th>
<th>Sort</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>Destination List</td>
<td>Row 1 of 3</td>
<td>Scroll ===cker</td>
</tr>
<tr>
<td>Report index file: 'RIHDGW.LUI.JIMIPR.INDEX'</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type one or more action codes.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sorted by: LUOWS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S=List LUOWs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A Tran/LTERM LUOW count (732)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>_ DFSTCFI 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>_ SMASTER 99</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>_ <em>CICS</em> 632</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>********************************************** Bottom of Data **********************************************</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3. View summary information about destinations in the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tran/LTERM</td>
<td>The Tran/LTERM field contains the name of the destination (such as a transaction or logical terminal).</td>
</tr>
<tr>
<td>LUOW count (total)</td>
<td>The LUOW count field contains a count of LUOWs that are associated with the destination. The (total) field contains a count of all LUOWs that are associated with all destinations in the list.</td>
</tr>
</tbody>
</table>

4. *optional* In the A (action) field to the left of a destination, enter the S action code.

   The LUOW List panel is displayed to show all LUOWs that are associated with the selected destination. For more information, see “Listing LUOWs” on page 216.

5. *optional* To sort the data on a different field, enter the SORT command.
For more information, see “Sorting listed items” on page 232.

## Listing origins

To view the names of all origins that are associated with one or more LUOWs, use the Origin List panel. This panel displays summary information about the origins and provides access to LUOWs that are associated with a selected origin.

### To access and work with the Origin List panel

1. Access the Report Index File Analysis panel.
   
   For instructions, see “Accessing reports in the ISPF interface” on page 212.

2. In the Analysis option field, enter 3.
   
   The Origin List pane is displayed:

   ![Figure 58: Origin List panel](image)

   - **Origin** field contains the name of the origin (such as an LTERM) of the transaction.
   - **LUOW count (total)** field contains a count of LUOWs that are associated with the origin. The (total) field contains a count of all LUOWs that are associated with all origins in the list.

3. View summary information about origins in the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin</td>
<td>The <strong>Origin</strong> field contains the name of the origin (such as an LTERM) of the transaction. Lowercase characters denote an element with no specifically identifiable origin. For example, <strong>ims</strong> denotes that the origin was an IMS system (rather than a specific LTERM).</td>
</tr>
<tr>
<td>LUOW count (total)</td>
<td>The <strong>LUOW count</strong> field contains a count of LUOWs that are associated with the origin. The (total) field contains a count of all LUOWs that are associated with all origins in the list.</td>
</tr>
</tbody>
</table>

4. *(optional)* In the A (action) field to the left of an origin, enter S.
The LUOW List panel is displayed to show all LUOWs that are associated with the selected origin. For more information, see “Listing LUOWs” on page 216.

5 *(optional)* To sort the data on a different field, enter the SORT command.

For more information, see “Sorting listed items” on page 232.

**Listing user IDs**

To view a list of all user IDs that are associated with one or more LUOWs, use the Userid List panel. This panel displays summary information about the user IDs and provides access to LUOWs that are associated with a selected user ID.

**To access and work with the Userid List panel**

1 Access the Report Index File Analysis panel.

For instructions, see “Accessing reports in the ISPF interface” on page 212.

2 In the **Analysis option** field, enter **4**.

The Userid List panel is displayed:

**Figure 59: Userid List panel**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Userid</td>
<td>The <strong>Userid</strong> field contains the user ID that is associated with the transaction. Lowercase characters denote an element with no specifically identifiable user ID. For example, <strong>nonusid</strong> denotes a LUOW with a security segment (indicating that it is a transaction LUOW), but the security segment does not contain a user ID.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>LUOW count (total)</td>
<td>The LUOW count field contains a count of LUOWs that are associated with the user ID. The (total) field contains a count of all LUOWs that are associated with all user IDs in the list.</td>
</tr>
</tbody>
</table>

4. *(optional)* In the A (action) field to the left of a user ID, enter S.

The LUOW List panel is displayed to show all LUOWs that are associated with the selected user ID. For more information, see “Listing LUOWs” on page 216.

5. *(optional)* To sort the data on a different field, enter the SORT command.

For more information, see “Sorting listed items” on page 232.

**Listing PSBs**

To view a list of all PSBs that are associated with one or more LUOWs, use the PSB List panel. This panel displays summary information about the PSBs and provides access to LUOWs that are associated with a selected PSB.

**To access and work with the PSB List panel**

1. Access the Report Index File Analysis panel.

   For instructions, see “Accessing reports in the ISPF interface” on page 212.

2. In the **Analysis option** field, enter 5.

   The PSB List panel is displayed:

   ![Figure 60: PSB List panel](image)

3. View summary information about PSBs in the following fields:
**Field**     | **Description**  
---|---  
PSB | The **PSB** field contains the name of the PSB that is associated with the LUOW.  
LUOW count (**total**) | The **LUOW count** field contains a count of LUOWs that are associated with the PSB. The (**total**) field contains a count of all LUOWs that are associated with all PSBs in the list.

4 *(optional)* In the **A** (action) field to the left of a PSB, enter **S**.

The LUOW List panel is displayed to show all LUOWs that are associated with the selected PSB. For more information, see “Listing LUOWs” on page 216.

5 *(optional)* To sort the data on a different field, enter the SORT command.

For more information, see “Sorting listed items” on page 232.

### Listing CICS transactions

To view a list of all CICS transactions that are associated with one or more LUOWs, use the CICS Transaction List panel. This panel displays summary information about the CICS transactions and provides access to LUOWs that are associated with a selected CICS transaction.

**To access and work with the CICS Transaction List panel**

1 Access the Report Index File Analysis panel.

For instructions, see “Accessing reports in the ISPF interface” on page 212.

2 In the **Analysis option** field, enter **6**.

The CICS Transaction List panel is displayed:

**Figure 61: CICS Transaction List panel**

---

<table>
<thead>
<tr>
<th>Field</th>
<th>Sort</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>CICS Transaction List</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Command ===&gt;</td>
<td>Scroll ===&gt; CSR</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

Report index file: 'RIHDGW.LUI.JIMIPR.INDEX'

Type one or more action codes.  
Sorted by: CICSTRAN  
S=List LUOWs

A CICS Tran LUOW count (632)

---

- ASMC  630  
- ASMT  2  

**Bottom of Data**
3 View summary information about CICS transactions in the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS Tran</td>
<td>The CICS Tran field contains the name of a CICS transaction that is associated with the LUOW.</td>
</tr>
<tr>
<td>LUOW count (total)</td>
<td>The LUOW count field contains a count of LUOWs that are associated with the CICS transaction. The (total) field contains a count of all LUOWs that are associated with all CICS transactions in the list.</td>
</tr>
</tbody>
</table>

4 (optional) In the A (action) field to the left of a CICS transaction, enter S.

The LUOW List panel is displayed to show all LUOWs that are associated with the selected CICS transaction. For more information, see “Listing LUOWs” on page 216.

5 (optional) To sort the data on a different field, enter the SORT command.

For more information, see “Sorting listed items” on page 232.

**Listing other LUOWs**

To view a list of nontransaction types of LUOWs and miscellaneous LUOWs, use the Other LUOWs List panel. This panel displays summary information about "other" types of LUOWs and provides access to LUOWs of each type.

**To access and work with the Other LUOWs List panel**

1 Access the Report Index File Analysis panel.

   For instructions, see “Accessing reports in the ISPF interface” on page 212.

2 In the Analysis option field, enter 7.

The Other LUOWs List panel is displayed:

**Figure 62: Other LUOWs panel**
3 View summary information about other LUOWs in the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>The <strong>Type</strong> field contains the label that Log Analyzer assigned to the type of LUOW. For more information, see “Nontransaction and other LUOW types” on page 367.</td>
</tr>
<tr>
<td>LUOW count (total)</td>
<td>The <strong>LUOW count</strong> field contains a count of LUOWs that are associated with LUOWs of this type. The (total) field contains a count of all LUOWs that are associated with all types of LUOWs in the list.</td>
</tr>
</tbody>
</table>

4 *(optional)* In the **A** (action) field to the left of a type of LUOW, you can enter **S**.

The LUOW List panel is displayed to show all LUOWs of the selected type. For more information, see “Listing LUOWs” on page 216.

5 *(optional)* To sort the data on a different field, you can enter the **SORT** command.

For more information, see “Sorting listed items” on page 232.

### Analyzing deadlock information

To list deadlock victims and analyze deadlock information, use the Deadlock Victims List panel.

In a deadlock, a transaction locks record A and attempts to lock record B. However, a second transaction has already locked record B and is attempting to lock record A. Because neither transaction can continue until the other transaction releases its locks, IMS typically issues a user 777 or 123 pseudo abend for one of the transactions.

You can use the Deadlock Victims List panel for the following purposes:

- View a concise list of applications that were affected by deadlocks.
View essential information about deadlock victims, sort the list by information of interest, and locate specified data values.

Select deadlock victims for further research.

To access and use the Deadlock Victims List panel

1 Access the Report Index File Analysis panel.

For instructions, see “Accessing reports in the ISPF interface” on page 212.

2 In the Analysis option field, enter 8.

The Deadlock Victims List panel is displayed:

Figure 63: Deadlock Victims List panel

3 View summary information about the deadlock victims in the following fields.

For more information about the data in these fields, see the IMS documentation for the IMS Record Format and Print Module (DFSERA30) of the exit routine of the File Select and Formatting Print Utility (DFSERA10).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DL#</td>
<td>The DL# field contains a sequential number that Log Analyzer assigned during the analysis job to identify this deadlock.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DMB</td>
<td>The DMB field contains the name of the data management block. This block describes the database that the transaction was attempting to access during this deadlock.</td>
</tr>
<tr>
<td>Count</td>
<td>The Count field contains the number of participants that are involved in the deadlock.</td>
</tr>
<tr>
<td>IMS/ESS</td>
<td>For each participant in the deadlock, the IMS/ESS field identifies the IMS system (or external subsystem) that was executing the transaction.</td>
</tr>
<tr>
<td>Tran/Job</td>
<td>For each participant in the deadlock, the Tran/Job field contains the name of the transaction or job.</td>
</tr>
<tr>
<td>PSB</td>
<td>For each participant in the deadlock, the PSB field contains the name of the program specification block that the transaction was using to access the database.</td>
</tr>
<tr>
<td>PCB/DBD</td>
<td>For each waiter (participant that is waiting to obtain a lock on the record) in the deadlock, the PCB/DBD field contains the name of the program communication block that the transaction was using to access the database.</td>
</tr>
<tr>
<td>PST</td>
<td>For each participant in the deadlock, the PST field contains the number of the partition specification table that describes the dependent region.</td>
</tr>
<tr>
<td>LUOW Id</td>
<td>For each participant in the deadlock, the LUOW Id field contains the LUOW number to which Log Analyzer assigned this transaction during the analysis job. You can use this field to find the LUOW in other reports.</td>
</tr>
</tbody>
</table>

4 *(optional)* To sort the list by information of interest, enter the SORT columnName command or use the Sort menu (as described in “Sorting listed items” on page 232).

5 *(optional)* To locate a specified data value, enter the LOCATE data command.

6 *(optional)* To view detailed information about a deadlock victim, enter S in the A (Action) field.

The Deadlock Detail panel is displayed:

![Figure 64: Deadlock Detail panel](image-url)
This panel displays the same information that the batch Deadlocks report contains. For more information, see “Using the Deadlocks report” on page 203 or the online Help for the panel.

You can work with Deadlock Detail panel as you would with any ISPF browse panel. In addition, you can navigate from this panel to the LUOW List panel (see “Listing LUOWs” on page 216) for a specific LUOW by performing either of the following actions:

- Enter the LUOW *nnnnnnnn* command (where *nnnnnnnn* is a valid LUOW ID that Log Analyzer assigned during the analysis job).
- Place the cursor on a displayed LUOW ID, and press F4.

### Obtaining summary information

To view summary information about the analysis job, use the Report Index File Summary panel. This panel provides general information about log records and LUOWs that were processed during the job.

**To access and work with the Report Index File Summary panel**

1. Access the Report Index File Analysis panel.
   
   For instructions, see “Accessing reports in the ISPF interface” on page 212.

2. In the Analysis option field, enter 9.

The Report Index File Summary panel is displayed:

**Figure 65: Report Index File Summary panel**

<table>
<thead>
<tr>
<th>File</th>
<th>View</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Analyzer</td>
<td>Report Index File Summary</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Command ===> _________________________________________________________________

More: +

Input files. Select if desired.
- View SYSIN control statements

Output files.
3 View summary information about the analysis job in the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>View SYSIN control statements</strong></td>
<td>Enter s or/ in the <strong>View SYSIN control statements</strong> field to display the SYSIN Control Statements panel. This panel reconstructs the control statement commands and keywords that the analysis job used when creating the index file. You can use these statements as a reference to determine how the index file was created and to re-create the file by using the original control statements. You might need to re-create the index file if the version of Log Analyzer that created the file is incompatible with the version of Log Analyzer that you are using to view the file. You can work with this panel as you would work with any ISPF Browse panel.</td>
</tr>
<tr>
<td><strong>Report index file</strong></td>
<td>The <strong>Report index file</strong> field contains the data set name of the index file that is used to work with LUOWs through the ISPF interface. You specified this data set name as the value of the DSN keyword on the INDEXFILE control statement.</td>
</tr>
<tr>
<td><strong>Extract file</strong></td>
<td>The <strong>Extract file</strong> field contains the data set name of the extract file that is used to contain a copy of each log record which has been associated with an LUOW of interest. You specified this data set name as the value of the DSN keyword on the EXTRACT control statement.</td>
</tr>
<tr>
<td><strong>Version</strong></td>
<td>The <strong>Version</strong> field indicates the Log Analyzer version, release, maintenance, and (if applicable) level numbers that performed the analysis and created the index file.</td>
</tr>
<tr>
<td><strong>Created</strong></td>
<td>The <strong>Created</strong> field contains the Julian date and time when the analysis job was executed.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Time span</td>
<td>The <strong>Time span</strong> field contains the timestamp of the first log record and the timestamp of the last log record that Log Analyzer considered as a candidate for selection. For information about how Log Analyzer selects these records, see “Identifying candidate records and creating LUOWs” on page 269.</td>
</tr>
<tr>
<td>Total log records</td>
<td>The <strong>Total log records</strong> field contains a count of all log records (except header records) that are contained in all input log record sources. Log Analyzer read all of these records to determine whether they are candidates for selection. This field can help you to gain a general understanding of the quantity of IMS traffic that is represented by the records in the input log record sources.</td>
</tr>
<tr>
<td>Total LUOWs</td>
<td>The <strong>Total LUOWs</strong> field contains the count of LUOWs that Log Analyzer created during the analysis job. For information about how Log Analyzer creates LUOWs, see “Creating LUOWs” on page 281. The value in this field is the sum of the count of transaction-type LUOWs (LUOWs with an origin, a destination, and a user ID) and the count of “other” LUOWs. This field can help you to determine whether the volume of LUOWs is small enough for you to analyze effectively.</td>
</tr>
<tr>
<td>Pre-select</td>
<td>The <strong>Pre-select</strong> field contains a count of log records that Log Analyzer identified as candidates for selection. For information about how Log Analyzer identifies these candidates, see “Identifying candidate records and creating LUOWs” on page 269. This field can help you to evaluate the effectiveness of control statements that influenced candidate identification. If too many records have been passed to the selection process, you can rerun the analysis job with narrower specifications.</td>
</tr>
<tr>
<td>Post-select</td>
<td>The <strong>Post-select</strong> field contains a count of log records that Log Analyzer selected for inclusion in LUOWs. For information about how Log Analyzer selects these records, see “Selecting log records and LUOWs” on page 283. If you did not specify any filter criteria, the value in this field is identical to the value in the <strong>Pre-select</strong> field. This field can help you to evaluate the effectiveness of control statements that influenced the selection process. If too many records have been selected, you can rerun the analysis job with narrower specifications.</td>
</tr>
<tr>
<td>Records extracted</td>
<td>The <strong>Records extracted</strong> field contains a count of log records that Log Analyzer wrote to an extract data set. If you provide an EXTRACT control statement, this count reflects the count of selected log records minus the count of unresolved associations. If you do not provide an EXTRACT control statement, this count is zero. This field can help you to verify the volume of extracted log records.</td>
</tr>
<tr>
<td>Discarded orphans</td>
<td>The <strong>Discarded orphans</strong> field contains the count of records that Log Analyzer was unable to assign to an LUOW and chose to discard during storage compression. For information about the cause of this condition, see “Creating LUOWs” on page 281.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unresolved orphans</strong></td>
<td>The <strong>Unresolved orphans</strong> field contains the count of records that Log Analyzer was unable to assign to an LUOW. For information about the cause of this condition, see “Unresolved associations field” on page 305.</td>
</tr>
<tr>
<td><strong>Log records extracted</strong></td>
<td>The <strong>Log records extracted</strong> field contains a list of log record types (as identified by a two-character log record type code) that Log Analyzer handled during the analysis job. For information about these codes, see “IMS log record types” on page 345.</td>
</tr>
</tbody>
</table>

## Sorting listed items

For panels that list LUOWs, destinations, origins, user IDs, other LUOWs, and log records, you can change the column that is used to sort the items in the list and the order (ascending or descending) of the items.

To sort the list, use any of the following methods:

- Sort pull-down menu
- SORT command
- Sort type List pop-up window

### To use the Sort pull-down menu

1. Place the cursor on the Sort action in the action bar at the top of the panel and press **Enter**.

   The Sort pull-down menu is displayed:

   **Figure 66: Sort pull-down menu**

   - 1. LUOW ID
   - 2. Origin
   - 3. Tran/LTERM
   - 4. User Id
   - 5. Log records
   - 6. Time
   - 7. Elapsed time
   - 8. Note
   - 9. Extended...

2. Enter the choice that corresponds to the column that you want to use for the sort.

   If you enter a choice other than the last choice, the list is redisplayed and the items in the list are sorted by the selected column.
If you enter the last choice (Extended...), the Sort type List pop-up window is displayed. For more information, see “To use the Sort type List pop-up window” on page 233.

**To use the SORT command**

1. Place the cursor in command area, and enter one of the following strings:

   - **SORT**
     
     The Sort type List pop-up window is displayed. For more information, see “To use the Sort type List pop-up window” on page 233.

   - **SORT colName order**
     
     For `colName`, specify the label of the column that you want to use for sorting the items.
     
     For `order`, you can specify **A** (ascending A to Z) or **D** (descending Z to A) to indicate the order in which to sort the items. If you omit this parameter, the sort order is ascending.
     
     The list is redisplayed and the items in the list are sorted by the specified column in the specified order.

**To use the Sort type List pop-up window**

1. Access the Sort type List pop-up window as described in “To use the Sort pull-down menu” on page 232 or “To use the SORT command” on page 233.

   The following figure shows the Sort LUOW List pop-up window. Other types are similar.

   **Figure 67: Sort LUOW List pop-up window**

   ![Sort LUOW List pop-up window](image-url)
2 Enter a choice in the **Primary sort** field and *(optional)* in the **Secondary sort** field.

These fields control the columns that are used to sort the rows of the list. The choices that are displayed on this panel depend on the list that was being displayed when you entered the SORT command. Enter the number that corresponds to your choice. The following choices can be listed:

<table>
<thead>
<tr>
<th>Choice</th>
<th>Sort the rows by</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUOW ID</td>
<td>LUOW identifier</td>
</tr>
<tr>
<td>Origin</td>
<td>Origin (source) name</td>
</tr>
<tr>
<td>Tran/LTERM</td>
<td>Transaction or LTERM (destination) name</td>
</tr>
<tr>
<td>Userid</td>
<td>User ID</td>
</tr>
<tr>
<td>Log Records</td>
<td>Log records</td>
</tr>
<tr>
<td>Timestamp</td>
<td>Date and time</td>
</tr>
<tr>
<td>Elapsed time</td>
<td>Elapsed time</td>
</tr>
<tr>
<td>Note</td>
<td>Data in the Note field</td>
</tr>
<tr>
<td>Type</td>
<td>Type of LUOW</td>
</tr>
<tr>
<td>Log code</td>
<td>Log code</td>
</tr>
<tr>
<td>Description</td>
<td>Description</td>
</tr>
<tr>
<td>Sequence number</td>
<td>Log sequence number</td>
</tr>
<tr>
<td>Attributes</td>
<td>Attributes</td>
</tr>
<tr>
<td>Message text</td>
<td>Message text</td>
</tr>
</tbody>
</table>

*Sorting by this column requires significant I/O, and response might be delayed.*

3 Enter a choice in the **Primary sort sequence** field and *(optional)* in the **Secondary sort sequence** field.

These fields control the order of items in the list. Enter the number that corresponds to your choice:

<table>
<thead>
<tr>
<th>Choice</th>
<th>Sort the rows by</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (Asc)</td>
<td>Ascending order (a field that starts with A comes before a field that starts with B)</td>
</tr>
<tr>
<td>2 (Des)</td>
<td>Descending order (a field that starts with B comes before a field that starts with A)</td>
</tr>
</tbody>
</table>

4 Press **Enter**.
The panel from which you initiated the sort action is redisplayed. The list is sorted, first by the primary field and then by the secondary field, in the sequences that you specified.

Obtaining LUOW details

To view details about an LUOW, use the LUOW luowid panel.

In the panel title, luowid is the LUOW ID that is assigned by Log Analyzer. Most fields on this panel display the same information that is displayed in the columns on the LUOW List panel. Some fields provide more information that, because of space limitations, cannot be displayed on the LUOW List panel.

To access and work with the LUOW List panel

1. Access the LUOW List panel.
   
   For instructions, see “Listing LUOWs” on page 216.

2. In the A (Action) field to the left of an LUOW, enter S.

   The LUOW luowid panel is displayed:

   ![LUOW luowid panel](image)

   **Figure 68: LUOW luowid panel**

   File Actions Options Help
   1-------------------------- 1-------------------------- 1-------------------------- 1--------------------------
   Log Analyzer LUOW 00000004
   Command ===> _________________________________________________________________
   More: +
   Enter LISTRECS on the command line to display log records for this LUOW.

   LUOW ID . . . . : 00000004
   Origin . . . . : ims
   Transaction/LTERM: MASTER2
   Userid . . . . : nonusid
   First record time: 2009.022 18:20:10.866221
   Last record time : 2009.022 18:20:10.866432
   Elapsed time . . : 00:00:00.000211
   UOW-1 . . . . : RB1C C3A2BA2DC1A12E06

   Counts.
   Log records . . : 5
   Recovery tokens: 0
   Message text . . : DFS3257I ONLINE LOG NOW SWITCHED - FROM DFSOLP03 TO DFSOLP04

   Notes:

   F1=Help F3=Exit F7=Up F8=Down F10=Previous F11=Next

3. View detailed information about the LUOW in the following fields:
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUOW ID</td>
<td>The LUOW ID field contains the LUOW number that Log Analyzer assigned arbitrarily during the analysis job. For more information, see “Creating LUOWs” on page 281.</td>
</tr>
<tr>
<td>Origin</td>
<td>For transaction-type LUOWs, the Origin field contains the origin (such as an LTERM) of the transaction. For other types of LUOWs, this field is blank.</td>
</tr>
<tr>
<td>Transaction/LTERM</td>
<td>For transaction-type LUOWs, the Transaction/LTERM field contains the destination (such as an LTERM) of the transaction. For other types of LUOWs, this field is blank.</td>
</tr>
<tr>
<td>Userid</td>
<td>For transaction-type LUOWs, the Userid field contains the user ID that is associated with the transaction. For other types of LUOWs, this field is blank.</td>
</tr>
<tr>
<td>TPIPE</td>
<td>For transaction-type LUOWs, the TPIPE field indicates the name of an OTMA transaction pipe that is associated with the transaction. For other types of LUOWs, or if no TPIPE is associated, this field is omitted.</td>
</tr>
<tr>
<td>TMEMBER</td>
<td>For transaction-type LUOWs, the TMEMBER field indicates the name of an OTMA transaction member that is associated with the transaction. For other types of LUOWs, or if no TMEMBER is associated, this field is omitted.</td>
</tr>
<tr>
<td>Client ID</td>
<td>For transaction-type LUOWs, the Client ID field identifies the OTMA client that is associated with the transaction. For other types of LUOWs, or if no client is associated, this field is omitted.</td>
</tr>
<tr>
<td>LU name</td>
<td>For transaction-type LUOWs, the LU name field indicates the name of an OTMA logical unit that is associated with the transaction. For other types of LUOWs, or if no LU name is associated, this field is omitted.</td>
</tr>
<tr>
<td>TPNAME</td>
<td>For transaction-type LUOWs, the TPNAME field indicates the name of an OTMA transaction program that is associated with the transaction. For other types of LUOWs, or if no TPNAME is associated, this field is omitted.</td>
</tr>
<tr>
<td>First record time</td>
<td>The First record time field contains the timestamp of the first log record that Log Analyzer included in the LUOW.</td>
</tr>
<tr>
<td>Last record time</td>
<td>The Last record time field contains the timestamp of the last log record that Log Analyzer included in the LUOW.</td>
</tr>
<tr>
<td>Elapsed time</td>
<td>The Elapsed time field contains the calculated elapsed time of the LUOW. Log Analyzer calculates the elapsed time by subtracting the timestamp of the first log record from the timestamp of the last log record.</td>
</tr>
<tr>
<td>UOW-1</td>
<td>The UOW-1 field contains the UOW-1 number that Log Analyzer has obtained (or created) and has assigned to this LUOW. For more information, see “LUOW numbers and UOW-1 numbers” on page 304.</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
Log Records | The Log Records field contains a count of log records that Log Analyzer associated with the LUOW.
Recovery Tokens | The Recovery Tokens field contains a count of recovery tokens that Log Analyzer found in log records which are associated with the LUOW.
Message Text | The Message Text field contains a portion of the data from the first type 01 or type 03 record in the LUOW. The Message Text (Offset n) field on the LUOW List panel determines the offset of the displayed data in the record.
Notes | If applicable, the Notes field contains more information about the LUOW. The following values are possible:
- **Abend=code**
  This value indicates that one or more abnormal terminations have occurred during processing of the LUOW; *code* is the first abend code that is encountered.
- **This LUOW contains DB2 log records.**
- **This LUOW contains Energizer for IMS Connect event trace records.**
- **This LUOW read databases.**
- **This LUOW spans more than one IMS.**
- **This LUOW updated databases.**
- **This LUOW was connected to an external subsystem.**

To display a list of log records that are associated with this LUOW, enter the LISTRECS command.

The LUOW `luowid` Log Records panel is displayed. For more information, see “Listing log records for an LUOW” on page 237.

## Listing log records for an LUOW

To list the log records for an LUOW, use the LUOW `luowid` Log Records panel (where `luowid` is the LUOW ID that is assigned by Log Analyzer). This panel displays information about every log record that Log Analyzer has associated with an LUOW and provides access to other panels for working with these records.
To access and work with the LUOW luowid Log Records panel

1 Access one of the following panels:
   - LUOW List panel
     For instructions, see “Listing LUOWs” on page 216.
   - LUOW luowid panel
     For instructions, see “Obtaining LUOW details” on page 235.

2 Perform one of the following actions:
   - On the LUOW List panel, enter L in the A (Action) field to the left of an LUOW.
   - On the LUOW luowid panel, enter the LISTRECS command.

For instructions, see “Obtaining LUOW details” on page 235.

The LUOW luowid Log Records panel is displayed:

**Figure 69: LUOW luowid Log Records panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>Scroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command</td>
<td>CSR</td>
</tr>
</tbody>
</table>

Report index file: 'RIHDGW.LUI.JIMIPR.INDEX'

Type one or more action codes. Enter LEFT or RIGHT for additional columns.

S=Log record detail  T=List adjacent records
D=Dump record       B=Browse record  ?=Log code help

Sorted by: TIMESTAMP

<table>
<thead>
<tr>
<th>A Code</th>
<th>Type</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ 01</td>
<td>input msg</td>
<td>drrn=0800004F (additional data segments)</td>
</tr>
<tr>
<td>_ 30</td>
<td>msg prefix</td>
<td>origims=RB1C destims=RB1C drrn=0800004E</td>
</tr>
<tr>
<td>_ 30</td>
<td>msg prefix</td>
<td>origims=RB1C destims=RB1C drrn=0800004E</td>
</tr>
<tr>
<td>_ 01</td>
<td>input msg</td>
<td>drrn=08000050 (additional data segments)</td>
</tr>
<tr>
<td>_ 30</td>
<td>msg prefix</td>
<td>origims=RB1C destims=RB1C drrn=0800004E</td>
</tr>
<tr>
<td>_ 35</td>
<td>msg enqueue</td>
<td>dest=SYMQEC08 tpipe=11255 drrn=0800004E</td>
</tr>
<tr>
<td>_ 08</td>
<td>pgm start</td>
<td>recovery token=RB1C 00003f7a00000000 tran=SYMQEC08</td>
</tr>
<tr>
<td>_ 5067</td>
<td>ext subsys</td>
<td>(start unit-of-recovery) psb=SYMQEC0H</td>
</tr>
<tr>
<td>_ 31</td>
<td>msg get</td>
<td>recovery token=RB1C 00003f7a00000000 dest=SYMQEC08</td>
</tr>
<tr>
<td>_ 03</td>
<td>output msg</td>
<td>origims=RB1C destims=RB1C origin=appcotma dest=appcotma</td>
</tr>
<tr>
<td>_ 03</td>
<td>output msg</td>
<td>drrn=08000052 (additional data segments)</td>
</tr>
<tr>
<td>_ 03</td>
<td>output msg</td>
<td>drrn=08000053 (additional data segments)</td>
</tr>
</tbody>
</table>

F1=Help    F3=Exit   F7=Up    F8=Down   F10=Left   F11=Right   F12=Cancel

3 View summary information about the log records in the following fields.

To toggle between views that display fields that are not visible in the current view, use the LEFT command (F10) and the RIGHT command (F11). To toggle between views that display more data in the Attributes field, enter the ALTVIEW command.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Code</td>
<td>The Log Code field contains the two-character log record type code and (if applicable) the two-character log record subtype code that identify the type of the log record. For a list of log record type codes, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Type</td>
<td>The Type field contains the short name of the type of the log record. For a list of log record short names, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Attributes</td>
<td>The Attributes field contains relevant information that Log Analyzer snipped from within the log record. The information that is displayed in this field depends on the type of log record. For more information about variable data, see “Variable data snipped from log records” on page 304. If this field contains the following string, no variable data was considered to be important to snip for this type of log record: (no formatted data)</td>
</tr>
<tr>
<td>Sequence No.</td>
<td>The Sequence No. field contains the log sequence number (which is in hexadecimal format) that IMS assigned to the log record.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>The Timestamp field contains the Julian date and the time when the log record was created.</td>
</tr>
<tr>
<td>Time difference between records</td>
<td>The Time difference between records field contains the calculated time difference between the record’s timestamp and the previous record’s timestamp. To highlight time differences that exceed a specified value, enter the value in the Highlight time field.</td>
</tr>
<tr>
<td>Time difference from benchmark</td>
<td>The Time difference from benchmark field contains the calculated time difference between the record’s timestamp and a selected benchmark record’s timestamp. You can set or change the benchmark by entering K in the A (action) field of the record that you want to use as the benchmark. The new benchmark record is denoted by (benchmark) in this field. You can clear the contents of this field by entering the RESET command.</td>
</tr>
</tbody>
</table>

4 To access other panels or work with log records, enter one of the following codes in the A (Action) field to the left of a log record:

<table>
<thead>
<tr>
<th>Code</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>View detailed information about the log record. The Log Record Detail panel is displayed. For more information, see “Obtaining log record details” on page 240.</td>
</tr>
<tr>
<td>T</td>
<td>View a list of log records that came before and after the selected record within the sequence of log records that were selected. The Log Records - Time Sequence panel is displayed. For more information, see “Listing adjacent log records” on page 242.</td>
</tr>
</tbody>
</table>
**Obtaining log record details**

To view detailed information about a log record, use the Log Record Detail panel.

Some fields on this panel display the same information that is displayed in the columns on the LUOW luowid Log Records panel. Some fields provide more information that, because of space limitations, cannot be displayed on the LUOW luowid Log Records panel or are displayed in abbreviated format.

<table>
<thead>
<tr>
<th>Code</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>Mark the selected record as the benchmark time for delta time calculations. Log Analyzer calculates the delta time difference for a log record by subtracting the timestamp of the benchmark record from the timestamp of the log record. The difference is reported in the <strong>Time difference from benchmark</strong> field.</td>
</tr>
<tr>
<td>D</td>
<td>View a dump of the log record. The Log Record luowid panel is displayed. For more information, see “Dumping log records” on page 245.</td>
</tr>
<tr>
<td>B</td>
<td>View the contents of the log record in an ISPF Browse session. For more information, see “Browsing log records” on page 246.</td>
</tr>
<tr>
<td>?</td>
<td>View general information about the log record type. This information is similar to the information in “IMS log record types” on page 345.</td>
</tr>
</tbody>
</table>

**5 (optional)** If the **Time difference between records** field is displayed, use the **Highlight time** field if you want to specify a threshold time value.

Enter a numeric value from .0001 through 3599. Times that equal or exceed this specified value are displayed in a contrasting color for greater visibility. To remove the highlighting, clear the value in this field.

**6 (optional)** If the **Attributes** field is displayed, use the FIND command on the **Command** line if you want to search for a specified string.

The string must contain one or more contiguous (nonblank) characters. The interface repositions the panel at the first record that contains the string. The interface displays other records that also contain the string in a contrasting color for greater visibility. To remove the highlighting, enter the RESET command.

To reposition the panel to the next record that contains the string, use the RFND command (typically assigned to the **F5** key).

**7 (optional)** You can change the column on which the records are sorted and the order in which they are listed.

For more information, see “Sorting listed items” on page 232.
To access and work with the Log Record Detail panel

1. Access the LUOW luowid Log Records panel.
   For instructions, see “Listing log records for an LUOW” on page 237.

2. In the A (Action) field to the left of a log record, enter S.
   The Log Record Detail panel is displayed:

   **Figure 70: Log Record Detail panel**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Code</td>
<td>The Log Code field contains the two-character log record type code and (if applicable) the two-character log record subtype code that identify the type of the log record. For a list of log record type codes, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Description</td>
<td>The Description field contains the long name of the type of the log record. For a list of log record short names, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Sequence Number</td>
<td>The Sequence Number field contains the log sequence number (which is in hexadecimal format) that IMS assigned to the log record.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>The Timestamp field contains the Julian date and the time when the log record was created.</td>
</tr>
</tbody>
</table>
## Listing adjacent log records

To view information about the log records that come before and after a selected record, use the Log Records - Time Sequence panel.

You can use this panel for the following purposes:

- View a log record in context with other log records, which might be associated with the same or a different LUOW.

- Identify other LUOWs that were active while a LUOW of interest was active so that you can better understand how the LUOWs might have influenced each other.

- Determine whether an unrelated event (that was not associated with a particular LUOW of interest) has occurred before, during, or after the LUOW was active or has not occurred at all.

### To access and work with the Log Records - Time Sequence panel

1. Access the LUOW luowid Log Records panel.

   For instructions, see “Listing log records for an LUOW” on page 237.

2. In the A (Action) field to the left of a log record, enter T.

The Log Records - Time Sequence panel is displayed:

**Figure 71: Log Records - Time Sequence panel**

```
File  View  Options  Help
Log Analyzer  Log Records - Time Sequence  Row 1 of 101
Command ===> ________________________________________________ Scroll ===> CSR

Report index file: 'RIHDGW.LUI.JIMIPR.INDEX'
Type one or more action codes. Enter LEFT or RIGHT for additional columns.
S=Log record detail  L=Display LUOW
D=Dump log record  B=Browse log record  Records before 100  After 100
```
The selected record is at the center of the panel, with the adjacent records before and after it. You can scroll up and down to see more adjacent records.

3 View summary information about the log records in the following fields.

To toggle between views that display fields that are not visible in the current view, use the LEFT command (F10) and the RIGHT command (F11). To toggle between views that display more data in the Attributes field, enter the ALTVIEW command.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Code</td>
<td>The Log Code field contains the two-character log record code and (if applicable) the two-character log record subtype code that identify the type of the log record. For a list of log record type codes, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Type</td>
<td>The Type field contains the short name of the type of the log record. For a list of log record short names, see “IMS log record types” on page 345.</td>
</tr>
<tr>
<td>Sequence No.</td>
<td>The Sequence No. field contains the log sequence number (which is in hexadecimal format) that IMS assigned to the log record.</td>
</tr>
<tr>
<td>Attributes</td>
<td>The Attributes field contains relevant information that Log Analyzer snipped from within the log record. The information that is displayed in this field depends on the type of log record. For more information about variable data, see “Variable data snipped from log records” on page 304. If this field contains the following string, no variable data was considered to be important to snip for this type of log record: (no formatted data)</td>
</tr>
<tr>
<td>LUOW ID</td>
<td>The LUOW ID field contains the LUOW number that Log Analyzer assigned arbitrarily during the analysis job. For more information, see “Creating LUOWs” on page 281.</td>
</tr>
<tr>
<td>&lt;</td>
<td>The &lt; field contains a symbol (&lt; ) to indicate the record that you selected to display the Log Record - Time Sequence panel. This record is also highlighted.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>The Timestamp field contains the Julian date and the time when the log record was created.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Time difference between records</td>
<td>The Time difference between records field contains the calculated time difference between the record’s timestamp and the previous record’s timestamp. To highlight differences that exceed a specified value, enter the value in the Highlight time field.</td>
</tr>
<tr>
<td>Time difference from benchmark</td>
<td>The Time difference from benchmark field contains the calculated time difference between the record’s timestamp and a selected benchmark record’s timestamp. The default benchmark is the record that you selected with the T action code on the LUOW luowid Log Records panel. You can change the benchmark by entering K in the A (action) field of the record that you want to use as the benchmark. The new benchmark record is denoted by (benchmark) in this field. You can clear the contents of this field by entering the RESET command.</td>
</tr>
</tbody>
</table>

4 To access other panels to work with log records, enter one of the following codes in the A (Action) field to the left of a log record:

<table>
<thead>
<tr>
<th>Code</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>View detailed information about the log record. The Log Record Detail panel is displayed. For more information, see “Obtaining log record details” on page 240.</td>
</tr>
<tr>
<td>L</td>
<td>View the LUOW that is associated with this log record. The LUOW List panel is displayed. For more information, see “Listing LUOWs” on page 216.</td>
</tr>
<tr>
<td>K</td>
<td>Mark the selected record as the benchmark time for delta time calculations. Log Analyzer calculates the delta time difference for a log record by subtracting the timestamp of the benchmark record from the timestamp of the log record. The difference is reported in the Time difference from benchmark field.</td>
</tr>
<tr>
<td>D</td>
<td>View a dump of the log record. The Log Record logrecnum panel is displayed. For more information, see “Dumping log records” on page 245.</td>
</tr>
<tr>
<td>B</td>
<td>View the contents of the log record in an ISPF Browse session. For more information, see “Browsing log records” on page 246.</td>
</tr>
<tr>
<td>?</td>
<td>View general information about the log record type. This information is similar to the information in “IMS log record types” on page 345.</td>
</tr>
</tbody>
</table>

5 *(optional)* If the Time difference between records field is displayed, use the Highlight time field if you want to specify a threshold time value.

Enter a numeric value from .0001 through 3599. Times that equal or exceed this specified value are displayed in a contrasting color for greater visibility. To remove the highlighting, clear the value in this field.

6 *(optional)* If the Attributes field is displayed, use the FIND command on the Command line if you want to search for a specified string.
The string must contain one or more contiguous (nonblank) characters. The interface repositions the panel at the first record that contains the string. The interface displays other records that also contain the string in a contrasting color for greater visibility. To remove the highlighting, enter the RESET command.

To reposition the panel to the next record that contains the string, use the RFND command (typically assigned to the F5 key).

## Dumping log records

To view the entire contents of log records that Log Analyzer has associated with an LUOW, use the Log Record logrecnum panel in standard dump format or vertical dump format.

You can use this panel for purposes such as these:

- If you need to conduct in-depth research on log records, this panel shows the data in a format that supports problem diagnosis.
- If Log Analyzer has not yet provided a map for a type of log record, this panel provides a method for obtaining relevant information from this type of record.

### To access and work with the Log Record logrecnum panel in standard or vertical dump format

1. Access one of the following panels:
   - LUOW luowid Log Records panel
     For instructions, see “Listing log records for an LUOW” on page 237.
   - Log Records - Time Sequence
     For instructions, see “Listing adjacent log records” on page 242.

2. In the A (Action) field to the left of a log record, enter D.

The Log Record logrecnum panel is displayed. The format of data on the panel depends on whether Standard format or Over/under (vertical) format is selected as the display preference. See the following figures for examples.

### Figure 72: Log Record logrecnum (dump in standard format) panel

```
Log Analyzer                                     Log Record 0000000000031508                      Line 00000000 Col 001 132
Command ===> _________________________________________________________________________________________Scroll ===> CSR
********************************************************** Top of Data ****************************************************
Record Description=IMS input message
Record code=X'01'   Length=00523  X'020B'
UOW-1=IR8V    B73812463498305
Log sequence number=X'0000000000031508'
Log STCK time=20020511347212
```
Browsing log records

To view the entire contents of log records that Log Analyzer has associated with an LUOW, use the Log Record *logrecnum* panel in ISPF browse format. You can use this panel to conduct in-depth research on log records and to view details for record types that Log Analyzer has not mapped.

To access and work with the Log Record *logrecnum* panel in browse format

1. Access one of the following panels:
- **LUOW luowid Log Records panel**
  For instructions, see “Listing log records for an LUOW” on page 237.

- **Log Records - Time Sequence**
  For instructions, see “Listing adjacent log records” on page 242.

2 In the **A** (Action) field to the left of a log record, enter **B**.

The Log Record **logrecnum** panel is displayed in browse format:

**Figure 74: Log Record logrecnum panel (in browse format)**

```plaintext
File  Help
-------------------------------------------------------------------------------
Log Analyzer Log Record 0000000000000007 Line 00000000 Col 001 080
Command ===> ________________________________________________ Scroll ===> CSR
********************************* Top of Data **********************************
-...á.BG{øIív-¿.....ê..ë...........PR9FSDBA z/OS 010700BA....mØ.
******************************** Bottom of Data ********************************
```

3 View data on the panel.

The data is presented in an ISPF Browse session. Standard ISPF Browse functions are available.
Allocating log record sources

During an analysis job, Log Analyzer must allocate one or more sources that contain the log records to be analyzed.

Log Analyzer can use the following types of log record sources:

- IMS system log data set (SLDS) or online log data set (OLDS) that is allocated through explicit specification
  
  To allocate IMS log data sets through explicit specification, provide the data set names (and other required information as applicable) by using the SLDS keyword on the ANALYZE control statement. For more information, see “Explicitly specifying IMS log data sets to allocate” on page 250.

- IMS SLDS or OLDS that is allocated through implicit specification
  
  To allocate SLDSs and OLDSs through implicit specification, omit the SLDS keyword from the ANALYZE control statement. Log Analyzer searches IMS RECON data sets for SLDS entries and obtains the necessary information for allocating the SLDS from those entries. For more information, see “Implicitly specifying IMS log data sets to allocate” on page 252.

- Log Analyzer extract file that was created during a previously executed analysis job and that is allocated through explicit specification
  
  To allocate an input extract file, provide the data set name by using the INPEXT keyword on the ANALYZE control statement. For more information, see “Using extract files as input” on page 257.

- Log extract file that was not created by Log Analyzer and that is allocated through explicit specification
  
  A log extract file can be created by the IMS File Select and Formatting Print utility (program DFSERA10), the IEBGENER utility, or a similar utility which creates a file that contains nothing except IMS log records in their original format.
  
  To allocate an input log extract file that was not created by Log Analyzer, provide the data set name by using the SLDS keyword on the ANALYZE control statement. For more information, see “Gathering data from remote systems” on page 259.
- **Energizer for IMS Connect** journal that contains IMS Connect event records and is allocated through explicit or implicit specification

  To allocate an input Energizer journal, provide the data set name by using the IPRJRNL keyword on the ANALYZE control statement, or provide the prefix that Energizer used when it created the journal by using the IPRPREFIX keyword. For more information, see “Using Energizer journals as input” on page 260.

- **DB2 log** that contains DB2 unit of recovery (UR) records and is allocated through explicit or implicit specification

  To allocate an input DB2 log, provide the data set name by using the DB2LOG keyword on the ANALYZE control statement, or provide the name of the DB2 bootstrap data set by using the DB2BSDS keyword. For more information, see “Using DB2 logs as input” on page 262.

- **WebSphere for MQ extract file** that contains extracted WebSphere for MQ bridge records, adapter records, or both types of record and that is allocated through explicit specification

  To allocate an WebSphere for MQ extract file, provide the data set name by using the MQEXT keyword on the ANALYZE control statement. For more information, see “Using WebSphere for MQ extract data sets” on page 265.

---

**Explicitly specifying IMS log data sets to allocate**

The following topics provide conceptual information and examples that are related to explicit specification of IMS log data sets.

**Overview of explicit specification**

You might want to use explicit allocation in the following circumstances:

- You know the data set names of the logs that you want to analyze.

- You want to process an active OLDS. (However, you are not required to use explicit specification. Log Analyzer can allocate an active OLDS through implicit allocation if you specify the ACTIVEOLDS=YES keyword.)

- You do not want to access the RECON data sets. (Log Analyzer must access the RECON data sets if you use implicit specification.)

To allocate a log data set through an explicit specification, use the SLDS keyword on the ANALYZE control statement to specify the data set name, the version of IMS that
created the data set, and (if the data set is not cataloged) the name of the volume that contains the data set. This method is valid for SLDSs and OLDSs.

When you use explicit specification, you must be careful to specify the correct IMS version. Although Log Analyzer attempts to detect a mismatch between the specified version and the version that created the records, only a few log record types contain the IMS version. Those records might not be part of the analysis (depending on your criteria for record selection), and a mismatch might be undetected.

**Specifying SLDS and OLDS data set names**

The following example shows the explicit allocation of a single log data set named IMSA.SLDS.G0059152.

This data set was created by an IMS version 9.1 system and is cataloged.

```
ANALYZE
SLDS=(IMSA.SLDS.G0059152,R91)
END
```

For more information, see “SLDS keyword” on page 77.

**Explicitly specifying log data sets that are not cataloged**

The following example shows the explicit allocation of several log data sets.

These data sets were created by various versions of IMS and are not cataloged. The third and fourth parameter values specify the unit and the volume name (VOLSER) so that Log Analyzer can find the data set.

```
ANALYZE
SLDS=(IMSA.SLDS,R91,TAPE,TEMPO1)
SLDS=(IMSB.SLDS,R81,TAPE,TEMPO2)
SLDS=(IMSC.SLDS,R91,TAPE,TEMPO3)
END
```

**Specifying the data set name of an active OLDS**

The following example shows the inclusion of an active OLDS among a set of other explicitly specified log data sets.

All data sets were created by IMS version 9.1 systems and are cataloged. Log Analyzer automatically detects end of file on an active OLDS and does not read past
this point. Message BMCLUI00643I indicates the data set name and the sequence
number of the last record that Log Analyzer read from the file.

```
ANALYZE SLDS=(IMSA.OLDS1,R91)
SLDS=(IMSA.OLDS2,R91)
SLDS=(IMSA.SLDS1,R91)
SLDS=(IMSA.SLDS2,R91)
END
```

**Implicitly specifying IMS log data sets to allocate**

The following topics provide conceptual information and examples that are related
to implicit specification of IMS log record sources.

**Overview of implicit specification**

To allocate SLDSs and OLDSs through implicit specification, omit the data set names
from the ANALYZE control statement.

Log Analyzer searches RECON data sets for SLDS entries and obtains the necessary
information for allocating the SLDS from those entries.

---

**Note**

BMC recommends that you allow Log Analyzer to search for log data sets in the
RECONs whenever possible. Implicit specification is the easiest and most reliable
method to use unless the log data sets that you want to use are not registered in the
RECONs.

Even with a large number of log data sets to process, Log Analyzer allocates the
RECONs for a maximum of a few seconds. This short duration of the allocation
prevents analysis jobs from interfering with other processes that access the RECONs.

---

When you use implicit specification to select a log data set, you do not need to know
the data set name, the version of IMS that created the data set, or the volume on
which the data set resides. Log Analyzer also ensures that all relevant logs
(including the OLDS, if applicable) are selected, so you do not have to be concerned
that a log might be omitted from the analysis accidentally.

To identify the RECON data sets to use, you can use either of the following methods:

- Specify RECON data set names explicitly by using the RECON keyword on the
  ANALYZE control statement.
Allocate RECON data sets through an IMS dynamic allocation (MDA) member by omitting the RECON keyword and the SLDS keyword from the ANALYZE control statement.

When you use implicit specification, you can use other keywords for customized results:

- To identify IMS systems that produced the SLDSs that you want to analyze, you can use the IMSID keyword on the ANALYZE control statement. If you omit this keyword, Log Analyzer selects SLDSs for all IMS systems that are defined in the RECON.

- To define the search for SLDS entries, you must specify START and STOP keywords on the INTERVAL control statement. These keywords define the range of time or log sequence numbers to be searched.

- To prevent an excessive number of SLDSs from being allocated through implicit specification, you can use the MAXLOGS keyword on the ANALYZE control statement.

- To allow Log Analyzer to select an active OLDS, you can use the ACTIVEOLDS keyword on the ANALYZE control statement.

Log Analyzer issues message BMCLUI000594I to list all system log data sets that were selected automatically from information in the RECONs.

### Specifying RECON data set names

To allocate log data sets through implicit specification, you can specify the data set names of the set of RECON data sets to use.

You must use this method if either or both of the following conditions are true:

- You want Log Analyzer to search more than one set of RECON data sets.

- You want to allocate one or more log data sets or extract data sets through explicit specification and also allocate SLDSs through implicit specification.

Specify the RECON data set names explicitly by using the RECON keyword on the ANALYZE control statement.

One of the specified data sets must be designated as COPY1 by IMS. Log Analyzer reads COPY1 to identify the log data sets for analysis.
The following example shows the specification of multiple sets of RECON data set names for processing.

```
ANALYZE
  RECON=(RCNDC.PR9I.RECON1,
       RCNDC.PR9I.RECON2,
       RCNDC.PR9I.RECON3)
  RECON=(RCNDC.PR9F.RECON1,
       RCNDC.PR9F.RECON2,
       RCNDC.PR9F.RECON3)
INTERVAL START=2007115/07152317 STOP=2007115/07451129
END
```

The following example shows the specification of one set of RECON data set names for processing. A SLDS is also explicitly specified.

```
ANALYZE
  RECON=(RCNDC.PR9I.RECON1,
       RCNDC.PR9I.RECON2,
       RCNDC.PR9I.RECON3)
  SLDS=(IMSA.SLDS,R91)
INTERVAL START=2007115/07152317 STOP=2007115/07451129
END
```

For more information, see “RECON keyword” on page 76.

### Allocating RECON data sets automatically

Log Analyzer can use IMS dynamic allocation (MDA) members to allocate a set of RECON data sets to use for implicit specification of log data sets.

This approach offers advantages such as these:

- You do not need to know the RECON data set names.
- You do not need to determine which of the RECON data sets is designated as the COPY1 data set.

This approach has limitations such as these:

- Only one set of RECON data sets can be used in the analysis job.
- You must include the name of the library that contains the RECON MDA members in the STEPLIB concatenation of the analysis job.

You can allocate the RECON data sets automatically by omitting the RECON keyword and the SLDS keyword from the ANALYZE control statement. Log Analyzer searches the STEPLIB concatenation and obtains the data set names from the first RECON1, RECON2, and RECON3 MDA modules that are found.
The following example shows the use of RECON MDA members to allocate the set of RECON data sets:

```plaintext
//LUIMAIN EXEC PGM=LUIMAIN,REGION=OM
//STEPLIB DD DISP=SHR,DSN=bmclui.loadlib
//         DD DISP=SHR,DSN=bmclui.BBLINK
//         DD DISP=SHR,DSN=bmclui.ICOLIB
//         DD DSN=SYS1.PR8F.LOADMDA,DISP=SHR MDA macros
//SYSIN DD *
ANALYZE
INTERVAL START=2007115/07152317 STOP=2007115/07451129 END
```

### Specifying IMSIDs

If you use implicit specification to obtain the list of log data sets to analyze, you can also use the IMSID keyword to identify one or more IMS systems that have created log data sets of interest.

Omit this keyword if you want Log Analyzer to select logs for all IMS systems.

You cannot specify the IMSID keyword if you specify the SLDS keyword.

The following example shows the selection of logs from several of IMS systems in multiple sets of RECON data sets:

```plaintext
ANALYZE IMSID=(PR9I,PR9M,PR9F,PR9J)
RECON=(RCNDC.PR9I.RECON1,
       RCNDC.PR9I.RECON2,
       RCNDC.PR9I.RECON3)
       (RCNDC.PR9F.RECON1,
       RCNDC.PR9F.RECON2,
       RCNDC.PR9F.RECON3)
INTERVAL START=2007115/07152317 STOP=2007115/07451129 END
```

For more information, see “IMSID keyword” on page 70.

### Specifying a maximum number of logs to read

When you are using implicit allocation, you usually do not know exactly how many logs are eligible for processing in any specified interval.

If the specified interval is too large, a large number of logs could be processed and the results of the analysis are likely to be unusable. You can use the MAXLOGS keyword prevent Log Analyzer from processing an excessive number of logs. If the number of logs to be processed is greater than the specified MAXLOGS number, Log
Analyzer issues warning message BMCLUI000601W and stops processing logs at the specified number.

Use caution when you specify the MAXLOGS keyword. Log data sets that are necessary for the analysis could be left out of the execution if the MAXLOGS value is exceeded.

The following example shows the use of the MAXLOGS keyword:

```
ANALYZE
  MAXLOGS=5
  RECON=(RCNDC.PR8F.RECON1,
          RCNDC.PR8F.RECON2,
          RCNDC.PR8F.RECON3)
  INTERVAL START=2007199/07121312 STOP=2007199/09231211
END
```

If 36 logs would be selected during this job, the following warning message would be issued:

```
BMCLUI000601W Selected logs 36 exceeds maxlog specification 5
```

For more information, see “MAXLOGS keyword” on page 74.

**Using an active OLDS through implicit specification**

To prevent IMS performance problems, by default Log Analyzer does not try to select an OLDS that is in active use by IMS.

If you want to allow Log Analyzer to select an active OLDS through implicit specification, you can specify the ACTIVEOLDS=YES keyword. (The ACTIVEOLDS keyword has no effect on the use of an active OLDS through explicit specification with the SLDS keyword.)

Log Analyzer automatically detects end of file on an active OLDS and does not read past this point. Message BMCLUI00643I indicates the data set name and the sequence number of the last record that Log Analyzer read from the file.

The following example shows the use of the ACTIVEOLDS keyword. If Log Analyzer finds an active OLDS within the specified interval, the active OLDS is selected for processing.

```
ANALYZE
  ACTIVEOLDS=YES
  RECON=(RCNDC.PR8F.RECON1,
          RCNDC.PR8F.RECON2,
          RCNDC.PR8F.RECON3)
  INTERVAL START=2007199/07121312 STOP=2007199/09231211
END
```
Implicitly specifying log data sets that are not cataloged

When you use implicit specification for noncataloged log data sets, Log Analyzer obtains the name (VOLSER) of the volume from the RECON data sets. You do not need to provide volume information.

Using implicit and explicit specifications in the same job

You can use implicit and explicit specifications in the same analysis job.

This approach is useful if you need to analyze logs that are not registered in the RECON data sets along with logs that are registered.

The following example shows an analysis job that uses implicit and explicit specifications of log data sets.

```
ANALYZE
RECON=(RCND.C.PR9F.RECON1,
       RCND.C.PR9F.RECON2,
       RCND.C.PR9F.RECON3)
RECON=(RCND.C.PR9I.RECON1,
       RCND.C.PR9I.RECON2,
       RCND.C.PR9I.RECON3)
SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS01,R91)
SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS02,R91)
SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS03,R91)
SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS04,R91)
SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS05,R91)
SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS06,R91)
SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS07,R91)
TIMEZONE=LOCAL
INTERVAL START=2007086/14581223 STOP=2007086/15383411
INTERVAL START=2007087/14370000 STOP=2007087/15010000
INTERVAL START=2007087/16100000 STOP=2007087/16250000
END
```

Using extract files as input

The following topics provide conceptual information and examples that are related to using a previously created extract file as a source of log records to read.
Overview of input extract files

An analysis job creates an extract file if you provide the EXTRACT control statement. This file contains copies of log records that are associated with LUOWs of interest. You can use an extract data set that was created by a previously executed analysis job as input to a new analysis job.

You might want to use an input extract file in situations such as these:

- You want to reduce the number of log records to process.
- You are gathering data from remote systems and processing them on a local system.

Allocating input extract files

To allocate a previously created Log Analyzer extract file, use the INPEXT keyword on the ANALYZE control statement to specify the data set name.

You can use any number of input extract files as input. You can combine the use of input extract files with other SLDSs and OLDSs that are specified explicitly or that are specified implicitly (if they are allocated from a RECON that is specified explicitly).

**Note**

If you allocate an input extract file, Log Analyzer requires a work file for sorting the extracted log records. You use the WORKFILE control statement to allocate this temporary file.

The following example shows the use of an input extract file along with other log data sets that are specified implicitly through RECON data sets.

```
ANALYZE
  RECON=(RCND,PR8F.RECON1,
        RCND,PR8F.RECON2,
        RCND,PR8F.RECON3)
  INPEXT=TME.RIP.LUI.EXAMPL.EXTR01
  INTERVAL START=2007199/084035/66 STOP=2007199/084214/15
WORKFILE
  STORCLASS=DEVSMS
  UNIT=DISK
  PRISPACE=50
  SECSPACE=50
  SPACEUNITS=CYL
END
```

For more information, see “INPEXT keyword” on page 70.
Reducing the number of selected log records

An analysis job can select a number of records that is too large for you to work with effectively.

If that job created an extract file, you can use the extract file as input to a new analysis job that has narrower specifications for selecting records. By reading log records from the extract file instead of from the original SLDSs and OLDSs, the new job consumes less time and fewer resources than the original job (assuming that the extract file contains a subset of the records from original log record sources).

The following example shows a job that creates an extract file for a specified time interval. (The log data sets are specified implicitly through RECON data sets that are allocated through MDA members.) The extract file is used by a subsequent job with a narrower time interval.

```
//JOB1 ...
...
ANALYZE
INTERVAL START=2007199/08403566 STOP=2007199/08421415
EXTRACT
   DSN=TME.RIP.LUI.EXAMPL.EXTR01
   UNIT=SYSDA
   VOLS=T0000A
   PRISP=300
   SECSP=120
   SPU=CYL
END

//JOB2 ...
ANALYZE
INTERVAL START=2007199/08410000 STOP=2007199/08415000
WORKFILE
   STORCLASS=DEVSMS
   UNIT=DISK
   PRISPACE=50
   SECSPACE=50
   SPACEUNITS=CYL
END
```

Gathering data from remote systems

Log Analyzer can create logical units of work that involve IMS systems that are related through Multiple Systems Coupling (MSC) links.

In some cases, systems logs are not physically in the same data center. You can use any combination of the following types of data sets as input to a Log Analyzer analysis job:

- OLDS
- SLDS
- Log Analyzer extract file that was created by a previously executed Log Analyzer analysis job
- Log extract file that was created by the IMS File Select and Formatting Print utility (program DFSERA10), the IEBGENER utility, or a similar utility which creates a file that contains nothing except IMS log records in their original format

For example, you can run a Log Analyzer analysis job on a remote MSC system to produce an extract file, transport the extract file to the main system, and use this extract file (with OLDSs, SLDSs, and extract files from the main system and other remote systems) as input to an analysis job that is running on the main system.

The following example shows an extract file that was created on a local IMS system (PR9F) and is being processed as input on a remote IMS system (PR9M):

```
ANALYZE
IMSID=(PR9M)
RECON=(RCNDC.PR9M.RECON1,
       RCNDC.PR9M.RECON2,
       RCNDC.PR9M.RECON3)
INPEXT=TME.RIP.LUI.SAMPMSC1.EXTRAC1
TIMEZONE=LOCAL
FILTER
   LOGRECORDCODES=(01,03,07,08,31,35,36,50,48,FA,F9)
   SELECT=CONTENT=(C’OVER’,01,19,4)
   CONTENT0103=DATA
INTERVAL
   START=2007086/14581234
   STOP=2007086/15381211
WORKFILE
   STORCLASS=DEVSMS
   UNIT=DISK
   PRISPACE=50
   SECSPACE=50
   SPACEUNITS=CYL
END
```

Using Energizer journals as input

The following topics provide conceptual information and examples that relate to analyzing journals that the BMC Energizer for IMS Connect product creates.

Overview of Log Analyzer processing of Energizer journals

If Energizer journal records are available, Log Analyzer incorporates selected records in LUOWs to provide information about IMS Connect data and timings.
Log Analyzer includes the following transaction-related Energizer events (listed in the order in which they usually appear in the LUOW):

1. Energizer event 73 indicates an IMS Connect read-socket event (meaning data came from a client in the TCP/IP network). This event is usually first in an LUOW that includes IMS Connect records.

2. Energizer event 65 indicates that IMS Connect sent a message to IMS.

3. IMS log records indicate work that was performed in IMS.

4. Energizer event 66 indicates that IMS sent a message to IMS Connect.

5. Energizer event 74 indicates that IMS Connect sent a message to the client in the TCP/IP network.

Log Analyzer reports pertinent data from Energizer journal records, such as the TCP/IP address and the client port number, in the LUOW Detail and Record Dumps reports. The LUOW Summary report inserts IMSconnect as a note to flag LUOWs that include Energizer data.

### Allocating Energizer journals explicitly

To allocate an Energizer journal explicitly, use the IPRJRNRL keyword on the ANALYZE control statement.

This keyword specifies the data set name of the Energizer journal to allocate.

In the following example, Log Analyzer allocates a single Energizer journal named BMC.IPRJRNRL.D2009113.

```
ANALYZE ...
IPRJRNRL=(BMC.IPRJRNRL.D2009113)
END
```

For more information, see “IPRJRNRL keyword” on page 71.

### Allocating Energizer journals implicitly

To allocate an Energizer journal implicitly, use the IPRPREFIX keyword on the ANALYZE control statement.

This keyword specifies the data set name prefix Energizer used when it created journal data sets. This prefix was specified in the Energizer options.
Note
Log Analyzer chooses Energizer journals to allocate by determining whether the journal matches the time range for the analysis. This time range could include an Energizer journal that is still being used by IMS Connect. In this case, the Log Analyzer allocation of the journal data set fails and the analysis terminates. You can issue an Energizer journal SWITCH command to cause IMS Connect to close and free the journal data set. The Log Analyzer analysis job might be reprocessed successfully after IMS Connect completes the switch. To avoid this issue, ensure that all Energizer journal data sets are freed before running the analysis job.

In the following example, Log Analyzer allocates the appropriate journals from two IMS Connect systems. The prefix that is defined in the Energizer options for this system is IMS1.IPRJRNLI.

```
ANALYZE ...
IPRPREFIX=(IMS1.IPRJRNLI)
IPRPREFIX=(IMS1.IPRJRNLI)
END
```

For more information, see “IPRPREFIX keyword” on page 72.

Using DB2 logs as input

The following topics provide conceptual information and examples that relate to analyzing DB2 logs with Log Analyzer.

Overview of DB2 log processing

If DB2 log records are available, Log Analyzer incorporates selected records in LUOWs to provide information about DB2 data and timings within IMS transactions. Log Analyzer includes the following transaction-related DB2 log records that comprise a unit of recovery (UR) in DB2:

- Type 0020—unit-of-recovery control
  Type 0020 records have the following subtypes:
  - 0001—begin the unit of recovery
  - 0002—begin Commit Phase 1
  - 0004—end Commit Phase 1
  - 0008—begin Commit Phase 2
  - 000C—transition from Commit Phase 1 to Commit Phase 2
  - 0010—end Commit Phase 2
  - 0020—begin abort
Allocating DB2 logs explicitly

To allocate a DB2 log explicitly, specify the data set name of the log by using the DB2LOG keyword on the ANALYZE control statement.

You can specify multiple logs; use a separate DB2LOG keyword to specify each log data set to allocate. You cannot specify the DB2LOG keyword if you specify the DB2BSDS keyword.

The following example allocates two DB2 logs named BMC.DB2LOG.D2009237 and BMC.DB2LOG.D2009238.

```
ANALYZE
... 
  DB2LOG=BMC.DB2LOG.D2009237
  DB2LOG=BMC.DB2LOG.D2009238
END
```

For more information, see “DB2LOG keyword” on page 68.

Allocating DB2 logs implicitly

Log Analyzer can allocate DB2 log data sets implicitly by obtaining log information from one or more DB2 bootstrap data sets (BSDSs).

A BSDS lists the names and statuses of all active and archived DB2 log data sets. Log Analyzer reads a BSDS to find and allocate appropriate archived DB2 logs. The allocation is based on the selected logical units of work (LUOWs) that Log Analyzer has created from IMS records. This process of implicitly allocating logs saves you time and effort when you create JCL to run an analysis job.

Considerations

Consider the following information when using BSDSs:
- Log Analyzer does not select active DB2 logs from a BSDS.

- Log Analyzer uses BSDSs that are VSAM key-sequenced data sets (KSDSs). Log Analyzer cannot use archived copies of BSDSs.

- A BSDS might contain information about other DB2 subsystems that match subsystems in Log Analyzer LUOWs. In that case, Log Analyzer attempts to use the BSDSs for those subsystems, too.

### Allocating DB2 logs from a DB2BSDS

To allocate a DB2 log implicitly, specify the data set name of the BSDS by using the DB2BSDS keyword on the ANALYZE control statement.

You can specify multiple BSDSs; use a separate DB2BSDS keyword to specify each BSDS to allocate. You cannot specify the DB2BSDS keyword if you specify the DB2LOG keyword.

You can allocate DB2 logs from a BSDS by specifying the DB2BSDS keyword on the ANALYZE control statement. You can build Log Analyzer control statements manually or by using the Log Analyzer ISPF interface.

#### To allocate DB2 logs from a BSDS (manual method)

1. Specify the DB2BSDS keyword on an ANALYZE control statement, as follows:

   ```
   ANALYZE
   DB2BSDS=
   bsdsName ...
   ```

   Replace `bsdsName` with the name of the BSDS that you want to use. To specify multiple BSDSs, use a separate DB2BSDS keyword for each one.

   **Note**
   
   You cannot specify the DB2BSDS keyword if you specify the DB2LOG keyword.

#### To allocate a DB2 log from a BSDS (ISPF interface method)

1. In the Log Analyzer ISPF interface, select the **DB2 BSDS** field on the ANALYZE Control Statement panel.

   **Note**
   
   You cannot select the **DB2 log** field if you select the **DB2 BSDS** field.

2. On the DB2 BSDS Input panel, specify one or more BSDS names.
Log Analyzer builds JCL for the analysis job that uses the BSDS names that you specified.

**Examples for using BSDSs**

The following example explicitly allocates a single IMS system log data set (SLDS) and implicitly allocates DB2 logs through a BSDS.

Log Analyzer creates LUOWs from the IMS log records in the specified SLDS. Then, Log Analyzer reads the specified BSDS to allocate DB2 logs that have time stamps within the ranges of those LUOWs.

```plaintext
ANALYZE
  SLDS=(IMSA.SLDS.D2009288.GEN010, R101)
  DB2BSDS=BMC.DB2.BSDS
...
INTERVAL
  START=FIRST
  STOP=LAST
END
```

The next example implicitly allocates IMS logs through a RECON and implicitly allocates DB2 logs through a BSDS. The INTERVAL keyword specifies the time range of the log records of interest. Log Analyzer reads the specified RECON to allocate IMS logs that contain records in the specified range, and creates LUOWs from those log records. Then, Log Analyzer reads the specified BSDS to allocate DB2 logs that have time stamps within the ranges of those LUOWs.

```plaintext
ANALYZE
  RECON=(IMSA.RECON1, IMSA.RECON2)
  DB2BSDS=BMC.DB2.BSDS
...
INTERVAL
  START=2010060/1830
  STOP=2010060/1845
END
```

For more information, see “DB2BSDS keyword” on page 68.

**Using WebSphere for MQ extract data sets**

Log Analyzer supports input log data from an IBM WebSphere for MQ (formerly known as IBM MQSeries) environment.

WebSphere for MQ writes log records when you use the WebSphere for MQ bridge between IMS and clients on other platforms and when you use the WebSphere for MQ adapter to process application calls. Log Analyzer can incorporate relevant WebSphere for MQ bridge records, adapter records, or both types of records into logical units of work (LUOWs). The resulting LUOWs provide a comprehensive
view of all transaction activity, from the point of origin through IMS through the final destination in WebSphere for MQ.

Creating WebSphere for MQ extract files

Log Analyzer cannot read WebSphere for MQ logs directly.

In a separate job step that runs before the analysis job step, you must run the IBM CSQ1LOGP utility to create extract files from the logs. For sample JCL to run the utility, see member LUI#MQEX in the CNTL library.

Specifying WebSphere for MQ extract files

To identify the WebSphere for MQ extract files to Log Analyzer, you must specify the MQEXT keyword of the ANALYZE control statement.

The value of MQEXT is the data set name of the extract file. If you want to include multiple extract files, specify one MQEXT keyword for each extract file, and specify the extract files in chronological order.

Example

You have created two extract files to include in the analysis.
MQTEST.EXTRACT.T0000 contains records with the earliest time stamps.
MQTEST.EXTRACT.T0800 contains records with the latest time stamps. In this example, you have provided the required IMS log input by specifying the SLDS keyword.

```
ANALYZE ...
SLDS=IMSTEST.SLDS.T0000
MQEXT=MQTEST.EXTRACT.T0000
MQEXT=MQTEST.EXTRACT.T0800
...
```

If you are using the ISPF interface to create control statements for Log Analyzer, you can select the MQ Extract field on the ANALYZE Control Statement panel. The MQ Extract Input panel is displayed for you to specify the extract files in chronological order.

WebSphere for MQ log records in LUOWs

Log Analyzer includes WebSphere for MQ log records in appropriate locations in LUOWs:
- Bridge records represent transactions that WebSphere for MQ sends to IMS and receives from IMS. These records appear at the beginning and end of LUOWs, and Log Analyzer considers these records in elapsed-time calculations and filters.

- Adapter records represent calls that the application has issued to write data to a WebSphere for MQ queue.

WebSphere for MQ does not write log records for application read calls because these calls are not recoverable. Therefore, Log Analyzer cannot include WebSphere for MQ read calls in LUOWs.

**WebSphere for MQ log records in reports and panels**

Log Analyzer reports and panels identify WebSphere for MQ log records by including `MQ` in **Note** and **Notes** fields. Reports and panels also include a code to indicate the type of log record as follows, and the **Variable** field contains the corresponding text:

<table>
<thead>
<tr>
<th>Log record code</th>
<th>Variable field text</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>alter</td>
</tr>
<tr>
<td>0002</td>
<td>define</td>
</tr>
<tr>
<td>0003</td>
<td>MQ get</td>
</tr>
<tr>
<td>0004</td>
<td>MQ put</td>
</tr>
<tr>
<td>0005</td>
<td>abort</td>
</tr>
<tr>
<td>0006</td>
<td>phase 1</td>
</tr>
<tr>
<td>0007</td>
<td>phase 2</td>
</tr>
<tr>
<td>0008</td>
<td>expire</td>
</tr>
<tr>
<td>0000</td>
<td>expire</td>
</tr>
</tbody>
</table>
Identifying candidate records and creating LUOWs

Log Analyzer can filter log records, filtering reduces the volume of log records that you must handle interactively or manually. As more data is included, interactive or manual analysis becomes slower and more difficult. If you have a busy IMS environment, for best results you should identify ranges and types of records that are related to the purpose of the analysis.

To identify a candidate log record and associate it with an LUOW, Log Analyzer performs the following tasks:

1. Log Analyzer reads a record from a specified log record source. For more information, see “Allocating log record sources” on page 249.

2. Log Analyzer applies the following tests to identify candidates as applicable:
   - Skip and limit tests specify a range of records, based on the total number of records that have been read, to consider as candidates. For more information, see “Skipping and limiting log records” on page 270.
   - The log record code test identifies types of log records to consider as candidates. For more information, see “Identifying log record codes” on page 271.
   - The interval test specifies a range of records, based on time or log sequence number, to consider as candidates. For more information, see “Specifying intervals” on page 277.

3. Log Analyzer retains or discards the record as follows:
   - If the record passes all applicable tests, it is retained as a candidate.
   - If the record does not pass all applicable tests, it is discarded. A discarded record is unavailable for further analysis.

4. Log Analyzer associates the candidate record with an LUOW. For more information, see “Creating LUOWs” on page 281.
Log Analyzer applies any specified SELECT filters and identifies LUOWs of interest.

For more information, see “Selecting log records and LUOWs” on page 283.

Skipping and limiting log records

The following topics provide conceptual information and examples that are related to skipping and limiting log records which are read from log data sets.

Note: Skipping and limiting log records is one way to specify a range of records to consider. Another way is to specify an interval of time or log sequence numbers. For more information, see “Specifying intervals” on page 277.

Overview of skip and limit tests

While reading records from log record sources, Log Analyzer keeps a running total number of records that it has read so far.

Skip and limit tests are based on this total number of records. You can use keywords to specify a number of records to be skipped before starting to consider records as candidates and to prevent more than a specified number of records from being considered.

You might want to use the SKIP keyword or the LIMIT keyword under the following conditions:

- You know that the beginning of a log data set contains records that are not needed in the analysis.

- You want to limit the number of log records that Log Analyzer tries to process so that fewer computing resources are consumed.

If you specify these keywords, you should be careful because they can cause log records that could be vital to your analysis to be omitted from the process.

If you specified the SKIP keyword on the ANALYZE control statement, Log Analyzer compares the SKIP value to the total. If the SKIP value is greater than or equal to the total, Log Analyzer discards the record. If the SKIP value is less than the total, Log Analyzer keeps the record and applies the next applicable test to that record. Log Analyzer stops applying the skip test to the next and subsequent records that are read.
If you specified the LIMIT keyword on the ANALYZE control statement, Log Analyzer compares the LIMIT value to the total. If the LIMIT value is less than or equal to the total, Log Analyzer keeps the record and applies the next applicable test to that record. If the LIMIT value is greater than the total, Log Analyzer stops reading records from the log record source as though end of file has been reached.

If you specified the SKIP keyword and the LIMIT keyword on the ANALYZE control statement, Log Analyzer waits to apply the limit test until the skip test no longer applies.

If you omit the SKIP keyword and the LIMIT keyword, Log Analyzer does not apply skip and limit tests.

If you specify the SKIP keyword or the LIMIT keyword, you must also specify the SLDS keyword.

**Specifying SKIP and LIMIT keywords**

The following example shows the use of the SKIP keyword and the LIMIT keyword.

Log Analyzer ignores the first 100 records of the input log data set, processes the next 2000 records, and then stops processing records.

```
ANALYZE
  SLDS=(TME.RIP.LUI.IMSV8.SLDSPL15,R81)
  SKIP=100
  LIMIT=2000
  TIMEZONE=LOCAL
END
```

For more information, see “SKIP keyword” on page 76 and “LIMIT keyword” on page 73.

**Identifying log record codes**

The following topics provide conceptual information and examples that are related to identification of types (and subtypes) of records to be considered as candidates for the record selection process.

**Overview of the log record code test**

The log record code test employs a filter that is based on the log record type and, if applicable, the log record subtype.
The type and subtype of a log record are identified by a log record code. If the subtype is omitted from the log record code, the test applies to all subtypes of the record type. Codes are assigned by the creator of the log record. For more information, see “IMS log record types” on page 345.

To prepare for the log record code test, Log Analyzer builds a list of included and excluded log record codes as follows:

- If you have omitted the LOGRECORDCODES keyword from the FILTER control statement, and if you have specified search criteria by using the SELECT keyword, Log Analyzer uses a default list of codes. For more information, see “Omitting the LOGRECORDCODES keyword” on page 272.

- If you are have coded the LOGRECORDCODES keyword on the FILTER control statement, the list contains codes that you specified explicitly for inclusion or exclusion. For more information, see “Specifying the LOGRECORDCODES keyword” on page 273.

The list also can contain codes that Log Analyzer has included automatically. For more information, see “Automatically included log record types” on page 274.

For a log record to pass the log record codes test, the type (and subtype, if applicable) of the log record must match an included record type (and subtype) and must not match an excluded type (and subtype). If a log record passes the log record codes test, Log Analyzer applies the next applicable test. If a log record does not pass the test, the record is discarded.

**Omitting the LOGRECORDCODES keyword**

If you do not have expert knowledge of log records, or if you want to avoid spending the time and effort that are required when you specify log record codes explicitly, you can omit the LOGRECORDCODES keyword from the FILTER control statement.

If you omit this keyword and you specify the SELECT keyword, Log Analyzer builds a default log record code list that corresponds to values in your SELECT statement. For more information, see “Automatically included log record types” on page 274.

If you omit the LOGRECORDCODES keyword and you do not specify the SELECT keyword, Log Analyzer uses the default LOGRECORDCODES=ALL.
Specifying the LOGRECORDCODES keyword

If you have expert knowledge of log records, or if you want a greater level of control over included and excluded log records than you can get when you omit the LOGRECORDCODES keyword, you can specify the LOGRECORDCODES keyword on the FILTER control statement.

Any combination of the following specifications are valid for the keyword value:

- You can include log record types and subtypes by specifying two-character log record type codes or four-character log record type and subtype codes.

- You can exclude log record types and subtypes by specifying a logical-not symbol (¬) preceding the log record codes.

- You can specify the name of a member in the log record code PDS. This member contains a list of included and excluded log record codes.

- You can code the value ALL to include all log record codes.

Note

Log Analyzer overrides the LOGRECORDCODES=ALL specification if all of the following conditions exist:

- You specify the LOGRECORDCODES=ALL keyword with no included or excluded log record codes.

- You specify the SELECT keyword on the FILTER control statement.

- You request one or more reports other than the Analysis Summary report (by specifying a keyword other than the SUMMARY keyword on the REPORTS control statement).

In this case, Log Analyzer uses a default list of log record codes to match the SELECT keyword values. A warning message and return code 4 are issued.

The following example shows the specification of a set of log record types that are related to transactions. (This set does not include every log record type that Log Analyzer associates with a transaction-type LUOW.)

```
ANALYZE
  SLDS=(TME.RIP.LUI.IMSV8.SLDSPL15,R81)
FILTER
  LOGRECORDCODES=(01,03,07,08,16,31,33,35,36,50,56)
END
```

For more information about syntax rules, see “LOGRECORDCODES keyword on the FILTER control statement” on page 85.
Automatically included log record types

In the following situations, Log Analyzer can include one or more log record types automatically that you have omitted:

- You specified a SELECT subkeyword that operates on the omitted type of record. For example, if you specify FILTER SELECT=CONTENT('string',01), and if you specify LOGRECORDCODES=(...) but omit the 01 code from the list, Log Analyzer automatically includes 01 records in the analysis. In this case, you can override automatic inclusion of a record type by specifying the LOGRECORDCODES keyword on the FILTER control statement with a logical-not symbol (¬) to exclude the record type.

- Log Analyzer requires the omitted type of record to determine associations between one or more explicitly specified types of records. In these cases, you cannot override automatic inclusion of a record type by specifying the LOGRECORDCODES keyword on the FILTER control statement with a logical-not symbol (¬) to exclude the record type. However, you can prevent the automatically included record type from being displayed in reports by specifying the LOGRECORDCODES keyword on the REPORTS control statement with a logical-not symbol (¬) to exclude the record type.

Log Analyzer issues message BMCLUI000607I to inform you when a record type was included automatically.

For a list of the log record codes that can be included automatically and the conditions under which they are included, see “Log record types that are included automatically” on page 369.

Summary of keywords that affect the log record code test

The following table summarizes how various combinations of keywords affect the log record codes that Log Analyzer uses to perform a log record code test.

Table 29: Summary of keywords that affect the log record code test

<table>
<thead>
<tr>
<th>Combination of SELECT keyword and REPORTS control statement keywords</th>
<th>Result when the LOGRECORDCODES keyword is omitted</th>
<th>Result when LOGRECORDCODES=ALL is specified with no included or excluded codes</th>
<th>Result when the LOGRECORDCODES keyword is specified with included or excluded codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>The SELECT keyword is omitted. (Keywords on the REPORTS control statement are irrelevant in this case.)</td>
<td>All log record codes are included in the list.</td>
<td>All log record codes are included in the list.</td>
<td>Specified log record codes are included in the list.</td>
</tr>
</tbody>
</table>
Using a log record code PDS

When you use Log Analyzer in expert mode, you can use several methods to specify the types of log records that are candidates for the log record selection process.

One method is to predefine a list of log record codes in a member of the log record code partitioned data set (PDS). Use of the log record code PDS is optional.

**Specifying keywords to use a member of the PDS**

To identify the name of the log record code PDS to use during an analysis job, use the LOGRECORDCODEPDS keyword on the ANALYZE control statement.

To identify a PDS member to use during an analysis job, specify the member name as the value of the LOGRECORDCODES keyword on the FILTER control statement.

The following example shows the use of a log record code PDS in an analysis job. The LOGRECORDCODEPDS keyword on the ANALYZE control statement specifies the name of a preallocated PDS. The LOGRECORDCODES keyword on the FILTER
control statement specifies the name of a member of this data set. This member contains the list of log record codes to be included in or excluded from the analysis.

```
ANALYZE ...
  LOGRECORDCODEPDS=BMC.LUI.LOGRECD
...
FILTER ...
  LOGRECORDCODES=LCUSRETO
...
END
```

For more information, see “LOGRECORDCODEPDS keyword” on page 73 and “LOGRECORDCODES keyword on the FILTER control statement” on page 85.

Creating the log record code PDS

The log record code PDS uses the following standard allocation parameters that are typically used for a JCL data set:
- Fixed-blocked (RECFM=FB)
- Partitioned organization (DSORG=PO)
- 80-byte records (LRECL=80)
- Block size that is a multiple of 80 bytes (BLKSIZE=n * 80)

Creating members in the log record code PDS

Each member of the log record code PDS contains a list of two-character or four-character log record codes.

**Note**

The following names are reserved and cannot be used as the name for a member of the log record code PDS:
- ALL
- Any two-character log record code (00 through FF)

Although some log record codes would otherwise be valid member names (A0 through FF), Log Analyzer always interprets a two-character value within the range of 00 through FF as a log record code.

Specify the contents of the member according to the following rules:

- Specify data in columns 1 through 72.
- Use one or more spaces (or a new line) as a separator between entries. You can specify leading blanks before the first entry on a line.
- You can use the logical-not symbol (¬) to specify log record codes to be excluded.
You can include masked values by specifying the first character of the log record type, followed by an asterisk (*), or by specifying the first three characters of the log record type and subtype, followed by an asterisk (nnn*).

Within a member, you cannot refer to another member of the log record code PDS.

You can specify an asterisk (*) in column 1 to indicate a comment.

---

**Example**
The following example shows the contents of a member of a log record codes PDS. This member lists the types of log records that typically are created during the execution of an IMS application program.

```
01 03 07 08 09 11 12 13 18 31 35 41 5*
```

---

**Example**
The following example shows the contents of a member of a log record codes PDS. This member lists the types of log records that typically are created during access to an IMS database.

```
20 21 24 25 26 27 28 4C 5*
```

---

**Sample members**
The product sample library contains samples that you can copy to create the members of your log record code PDS. These members are named LUIZxxxx.

For an annotated list of samples, see “Sample and syntax reference” on page 337.

### Specifying intervals

This section provides conceptual information and examples that are related to the specification of intervals of log records to consider as candidates for the record selection process.

### Overview of the interval test

The interval test determines whether a log record falls within a specified range of records.

Log Analyzer applies this test if you specified a range of time stamps (or log sequence numbers) by using START and STOP keywords on the INTERVAL control statement. If the time stamp (or the log sequence number) of the record is later than
If the START keyword value and is earlier than the STOP keyword value, the record is a candidate for the record selection process. If the record is outside of this range, it is discarded.

If you omit the INTERVAL control statement, Log Analyzer does not apply the interval test.

**Note**

If the interval is not long enough, records that are important to a correct analysis might be omitted. If the interval is too long, unnecessary processing of records and waste of computing resources can result.

### Specifying start and stop times

If you use implicit specification to obtain a list of log data sets to analyze from **RECON** listings, you must provide date and time stamps for the START keyword and the STOP keyword.

**RECON** data sets typically contain numerous log entries, and limiting the scope of the time interval reduces the potential number of logs that would be selected for analysis. For prudent use of computing resources, specify the most narrow time frame that you can.

The following example shows a typical interval specification.

**Example**

```
ANALYZE ...
INTERVAL START=2007086/14581223 STOP=2007086/15383411
END
```

You can specify generic time stamps with specific dates. The following example shows the use of generic time stamps.

**Example**

```
ANALYZE
  SLDS=(TME.RIP.PR9M.D07086.T1459553,R91)
  TIMEZONE=LOCAL
INTERVAL
  START=2007086/15060000
  STOP=2007086/15090000
END
```

For more information, see “START keyword” on page 80 and “STOP keyword” on page 82.
Specifying first and last log records

If you are using explicit specification for a log data set (with the SLDS keyword), you can use a specific start time in combination with a generic stop time of LAST, or you can use a generic start time of FIRST with a specific stop time.

Log data sets may contain millions of log records, and starting with a specific time and stopping at a specific time can reduce the overall number of log records to process. In the same manner, starting at the beginning of a log and stopping at a specific time can cut down on the number of log records to be analyzed.

The following example shows the use of a specific start time combined with a generic stop time of LAST:

```
Example
ANALYZE
  SLDS=(TME.RIP.PR9M.D07086.T1459553,R91)
  TIMEZONE=LOCAL
  INTERVAL
    START=2007086/15081215
    STOP=LAST
END
```

The following example shows the use of a generic start time of FIRST combined with a specific stop time:

```
Example
ANALYZE
  SLDS=(TME.RIP.PR9M.D07086.T1459553,R91)
  TIMEZONE=LOCAL
  INTERVAL
    START=FIRST
    STOP=2007086/15023423
END
```

For more information, see “START keyword” on page 80 and “STOP keyword” on page 82.

Specifying log sequence numbers

If you are using one log data set or one extract file as input, you can specify intervals by using the log sequence numbers that IMS assigns to log records.

You might want to use this approach if you know the log sequence numbers of a range of records of interest or if you want to analyze a few known records as part of a large unit of work.
The following example shows the selection of the first 4095 log records from an input log data set:

```
Example
ANALYZE
   SLDS=(TME.RIP.PR9M.D07086.T1459553,R91)
   TIMEZONE=LOCAL
INTERVAL
   START=00001
   STOP=00FFF
FILTER LOGRECORDCODES=ALL
END
```

The following example shows the use of an extract file as input and the specification of an interval in terms of log sequence numbers:

```
Example
ANALYZE
   INPEXT=TME.RIP.LUI.SAMPMSC1.EXTRAC1
   TIMEZONE=LOCAL
INTERVAL
   START=16D01
   STOP=1FFFF
FILTER LOGRECORDCODES=ALL
END
```

For more information, see “START keyword” on page 80 and “STOP keyword” on page 82.

**Specifying multiple intervals in an analysis job**

You can split time intervals for processing by specifying multiple INTERVAL control statements.

Multiple intervals offer advantages such as these:

- More ranges allow flexibility in capturing information of interest if records are spread across a relatively wide time range.

- Log Analyzer might be able to process fewer input logs to satisfy the analysis request.

The following example shows the specification of multiple intervals in a single analysis job:
Creating LUOWs

The following topics provide conceptual information and examples that are related to creation of LUOWs.

Overview of LUOWs and LUOW creation

An LUOW is a set of related log records that represent all actions that were performed to accomplish a particular activity in the IMS environment.

To create an LUOW, Log Analyzer associates log records with each other.

For each log record that is identified as a candidate for the record selection process, Log Analyzer determines whether the record belongs to an LUOW that has already been created and, if so, associates the record with that LUOW. If not, Log Analyzer creates a new LUOW and associates the record with the new LUOW.

The process of creating LUOWs and associating records with them is performed concurrently with the process of identifying candidates and (if you specified SELECT filters) comparing records to the filters. For more information, see “Selecting log records and LUOWs” on page 283.
Unresolved associations when creating LUOWs

In some cases, Log Analyzer never finds a record that is required for resolution of an LUOW.

This condition can occur if the required record to make the connection is not included in the list of candidates for selection. In this case, an orphan record results. This condition is known as an *unresolved association*.

Details about unresolved associations are reported on the Unresolved Orphans report. For more information, see “Using the Unresolved Orphans report” on page 200.
Selecting log records and LUOWs

Log Analyzer can filter log records by applying search criteria that you have specified by using the SELECT keyword on the FILTER control statement.

SELECT filters reduce the volume of log records that you must handle interactively or manually. As more data is included, interactive or manual analysis becomes slower and more difficult. If you have a busy IMS environment, for best results you might have to identify a characteristic (such as a transaction origin, destination, or user ID) that is associated with the records of interest to you.

To apply SELECT filters and evaluate LUOWs, Log Analyzer performs the following steps:

1. If you specified one or more SELECT filters, Log Analyzer processes each candidate log record as follows:
   a. Log Analyzer compares values of fields from within the record to the SELECT filters that you specified. For more information about SELECT filters, see “Specifying SELECT filters” on page 284.
   b. Log Analyzer records the result of the comparison. For more information, see “Selecting log records and LUOWs Log Analyzer can filter log records by applying search criteria that you have specified by using the SELECT keyword on the FILTER control statement”.

2. Log Analyzer evaluates the LUOW to determine whether it is an LUOW of interest.
   For more information, see “Selecting log records and LUOWs Log Analyzer can filter log records by applying search criteria that you have specified by using the SELECT keyword on the FILTER control statement”.

3. Log Analyzer handles the evaluated LUOWs as follows:
   ■ If an LUOW is of interest, Log Analyzer retains it.
   ■ If an LUOW is not of interest, Log Analyzer discards it.

4. Log Analyzer finalizes data and produces output data sets.
   For more information, see “Working with analysis data” on page 291.
Specifying SELECT filters

This section provides conceptual information and examples that are related to specifying SELECT filters.

Overview of SELECT filters

You can use SELECT filters to narrow the focus of an analysis job to log records and LUOWs that meet a specific set of conditions.

You specify filters by using keywords and values, and you can combine filters with Boolean logic to narrow the scope of the search for LUOWs. A narrower scope of records and LUOWs results in a smaller amount of output data from the analysis. Fewer elements can be managed more efficiently and effectively during problem determination.

Knowledge of the applications that are running in the IMS systems is valuable when you are determining the values to use for filter keywords.

You might want to start by making a broad sweep of the logs (by running the analysis with interval and log record code limitations but without filters) and producing only the Analysis Summary report. This action provides a general picture of the data within the specified interval and helps you to determine whether the time frame is of interest.

You specify SELECT filters by using the SELECT keyword on the FILTER control statement. Each filter consists of the following elements that form an *expression*:

- A subkeyword indicates the type of filter, which usually corresponds to a field (such as a transaction name, a logical terminal, or a user ID) to search in the record.
- A filter value indicates the value that is the target for comparison.
- A logical operator indicates how the value of the record field must be compared to the filter value for the comparison to result in a true condition.

Specifying a filter type

You can specify the following types of filters:

- Abended units of work
- APPC logical terminal name
- CICS transaction name
- Client ID
- Content (a specified string, at an optional offset within the record)
- Database name (with read-only access, update access, or both)
- Destination of a transaction
- Elapsed time of an LUOW
- Logical terminal name
- APPC logical unit name (LUNAME)
- Message processing region (MPR) name
- Node name
- Originating IMS
- Origin of a transaction
- OTMA port number
- PSB name
- Partition specification table (PST) identifier
- OTMA transaction member (TMEMBER) name
- OTMA transaction pipe (TPIPE) name
- APPC transaction program name (TPNAME)
- Transaction name
- Unit of work (UOW) identifier (labeled as UOW-1 in reports)
- User ID that is associated with the record

You can search for databases that have been read, updated, or both read and updated. For full-function databases that have been updated, you can specify an optional data set ID. For Fast Path database that have been updated, you can specify an optional area name and an optional RBA.

**Example**

The following example shows the use of a SELECT DBD filter to search for LUOWs that involve the DI21PART database:

```
ANALYZE
  SLDS=(TME.RIP.LUI.IMSV9.SLDSP115.P1,R91)
  INTERVAL START=FIRST STOP=LAST
  FILTER
    SELECT=DBD=DI21PART
END
```

For more information, see “SELECT keyword” on page 89.
**Specifying values in SELECT filters**

Choosing the right SELECT filter for selection depends on the problem that you are researching.

The value that you specify for the filter is also important. Any known specific information can be used in your search criteria. Precise information helps to produce precise results, but partial values can be helpful as well. As the number of data values that you provide for the search increases, the length of time needed to identify and diagnose problems decreases.

The following example shows the use of SELECT filters to search for a specific user ID:

```
ANALYZE SLDS=(TME.RIP.LUI.IMSV8.SLDSPL15,R81)
INTERVAL START=FIRST STOP=LAST
FILTER SELECT=USERID=RIP0005
END
```

The following example shows the use of partial (masked) values:

```
ANALYZE SLDS=(TME.RIP.LUI.IMSV9.SLDSPL15.P1,R91)
INTERVAL START=FIRST STOP=LAST
FILTER SELECT=(USERID=LO* OR USERID=RIP*)
END
```

**Using logical operators in SELECT filters**

A logical operator indicates how the value of the record field must be compared to the filter value for the comparison to result in a true condition.

For SELECT expressions other than the SELECT=ETIME expression, to select the record if the value in the record matches the specified value, use the equal sign (=) logical operator. To select the record if the value in the record does not match the specified value, use the NOT= (a literal NOT, followed by an equal sign) logical operator.

For the SELECT=ETIME expression, use the greater-than sign (>) or less than (<) sign as the logical operator.

A logical operator indicates whether the value of the record field must be equal to or not equal to the filter value for the comparison to result in a true condition.
The following example shows the use of complex SELECT filters. They search for LUOWs in which the transaction name LOCLTRAN was input from OTMA except those from TPIPEs that begin with the value FRED.

```
ANALYZE
  SLDS=(TME.RIP.LUI.IMSV9.SLDspl15.P1,R91)
  SLDS=(TME.RIP.LUI.IMSV9.SLDspl15.P2,R91)
INTERVAL START=FIRST STOP=LAST
FILTER
  CONT=DATA
  SELECT=(01DEST=LOCLTRAN AND TPIPE NOT=FRED*)
END
```

### Searching for content in 01 and 03 records

Sometimes problem determination can be based on data that was passed into or out of applications when other pertinent information is not known.

Log Analyzer can search for content in the data portion or all portions of type 01 (IMS input message) log records and type 03 (IMS output message) log records. For other log record types, Log Analyzer searches the entire log record. You can use the CONTENT0103 keyword on the FILTER control statement to control how content searches are performed.

To ensure that the search text is found in the target area of the data component of a type 01 or type 03 record, you must specify offset, length, and value parameters. Some data portions of these log records are large, and using offset and length parameters can help limit the amount of storage that is required to handle an LUOW of interest.

Scratch-pad area (SPA) segments are always excluded from the area that is searched for the CONTENT value. SPA segments are not transmitted to or from a terminal and are not relevant to a content search.

The offset that you can specify with the CONTENT keyword is relative to zero. For example, if you specify 1 as the value, the search starts at the second byte of the record.

### Example

The following example shows a content search for the character string `OVER` starting in position 20 of the user data portion and stopping after 4 bytes:

```
ANALYZE
  SLDS=(TME.RIP.LUI.IMSV9.SLDspl15.P1,R91)
  SLDS=(TME.RIP.LUI.IMSV9.SLDspl15.P2,R91)
INTERVAL START=FIRST STOP=LAST
FILTER
  CONT=DATA
  SELECT=CONTENT = (C'OVER',01,19,4)
END
```
The following example shows a content search for the character string **OTMACLB** in type 35 records:

```
ANALYZE
  SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS01,R91)
  SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS02,R91)
  SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS03,R91)
  SLDS=(TME.RIP.LUI.IMSV9.SQ.SLDS04,R91)
FILTER
  LOGRECORDCODES=(01,03,07,08,31,35,36,50,48,FA,F9)
  SELECT=CONTENT=(C'OTMACLB',35
CONTENT0103=ALL
END
```

### Specifying multiple SELECT filters with Boolean operators

Use of Boolean operators further qualifies a search to focus on particular LUOWs.

You can combine different known data values with related keywords to create better search queries when the know data values have dependencies on or relationships to one another.

To specify multiple filters, use Boolean operators (AND and OR) between each filter and the next one. You can group filters by enclosing them within parentheses.

The following example shows the use of a Boolean search for LUOWs that have a TMEMBER value of OMT20018 and a PSB value of APPL0001:

```
ANALYZE
  SLDS=(TME.RIP.LUI.IMSV8.SLDSPL15,R81)
  INTERVAL START=FIRST STOP=LAST
FILTER
  SELECT= (TMEMBER=OMT20018 AND PSB=APPL0001)
END
```

You can build complex queries to search on various pieces of data. The following example shows the use of content, user ID, PSB, and TPIPE filters to find LUOWs that match all of specified criteria:
Selection process for log records and LUOWs

This section provides conceptual information and examples that are related to selecting log records and LUOWs to find LUOWs of interest.

Record evaluation

If you specified SELECT filters, Log Analyzer processes a candidate log record by comparing each individual filter against the corresponding value in the record.

If you did not specify SELECT filters, Log Analyzer bypasses the evaluation of records, and all records are selected.

LUOW evaluation

Log Analyzer evaluates each LUOW to determine whether it is an LUOW of interest as follows:
If you have specified no SELECT filters, all LUOWs are designated as LUOWs of interest.

If you have specified a single SELECT filter, and if the evaluation of at least one record that is associated with the LUOW results in a true condition, the LUOW is designated as an LUOW of interest.

If you have specified multiple SELECT filters (expressions), Log Analyzer evaluates the LUOW by working with pairs of expressions. The evaluation proceeds from left to right, starting within the innermost parentheses and working toward the outermost parentheses. If you specified the AND Boolean operator between expressions, both expressions must be true for the evaluation to result in a true condition. If you specified the OR Boolean operator between expressions, either expression can be true for evaluation to result in a true condition. If the overall evaluation results in a true condition, the LUOW is designated as an LUOW of interest.

An LUOW of interest is retained. An LUOW that is not designated as an LUOW of interest is discarded.

For example, a series of log records in an LUOW contain the following data values:

- USERID=RIHLXM
- DEST=LT74X
- DBD=PART001

Log Analyzer evaluates SELECT expressions for this LUOW as follows:

<table>
<thead>
<tr>
<th>SELECT expressions</th>
<th>Results of LUOW evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT = USERID=RIHL*</td>
<td>True</td>
</tr>
<tr>
<td>SELECT = USERID=RIHL* AND DEST=ABC</td>
<td>False</td>
</tr>
<tr>
<td>SELECT = (USERID=RIHL* OR DEST=ABC) AND DBD=PART*</td>
<td>True</td>
</tr>
<tr>
<td>SELECT = (USERID=RIHL* AND DEST=ABC) OR DBD=PART*</td>
<td>True</td>
</tr>
</tbody>
</table>
Working with analysis data

This chapter provides detailed information about working with data that is produced by the BMC Log Analyzer component for IMS.

Overview of working with analysis data

After evaluating logical units of work (LUOWs), Log Analyzer sorts the retained LUOWs of interest and finalizes statistics to be reported.

Log Analyzer can report the results of the analysis job in the following ways:

- If you specified keywords on the REPORTS control statement, Log Analyzer can produce a series of batch reports.
  For more information, see “Producing batch reports” on page 291.

- If you specified the EXTRACT control statement, Log Analyzer writes log records of interest to an extract file. You can use this file as input to another analysis job. If you also specified the INDEXFILE control statement, you can view the extracted records through the ISPF interface.
  For more information, see “Working with extract files” on page 296.

- If you specified the INDEXFILE control statement (and the EXTRACT control statement), Log Analyzer prepares the results of the analysis job for viewing through the ISPF interface.
  For more information, see “Working with index files” on page 300.

Producing batch reports

This section provides detailed information and examples that are related to producing batch reports.
Overview of batch reports

Log Analyzer produces batch reports that you can use to verify the success of the analysis job and to examine LUOWs manually.

The following reports are available:

- The Analysis Summary report summarizes information about the log records that Log Analyzer has handled during the analysis job and about how Log Analyzer has organized these records into LUOWs.

- The Audit report provides information that is relevant to an audit of events which have occurred in the IMS environment.

- The LUOW Summary report summarizes information about the LUOWs that Log Analyzer has created during the analysis job.

- The LUOW Detail report lists key information about the individual log records that Log Analyzer has associated with an LUOW.

- The Record Dumps report displays the contents of log records in a standard format or a vertical format.

- The Time Sequence report lists every log record that was selected for analysis in the order in which it was written to an IMS log.

- The Unresolved Orphans report provides information about records for which Log Analyzer was unable to resolve LUOW associations.

Batch reports can be helpful for several purposes:

- During initial analysis of a set of logs for problem determination, batch reports can help you assess the overall contents of the logs and whether the logs contain the necessary records for a specific origin, destination, user ID, or other element of interest.

- Batch reports are a convenient method for providing information to be analyzed by other personnel (such as an auditor, an application developer, a DBA).

- A single analysis job might not produce all of the information that is required for problem determination or to meet the needs of other personnel. You can run multiple analysis jobs with different criteria and find the answer in the combination of batch reports from all of the jobs.
Choosing batch reports to produce

To control whether and how Log Analyzer produces batch reports, use keywords on the REPORTS control statement.

Log Analyzer writes reports to SYSOUT-type data sets. For more information, see “Specifying REPORTS control statements” on page 107.

Production of batch reports is optional. The Analysis Summary report is produced if you do not specify a REPORTS control statement.

Remember the following facts when you are requesting batch reports other than the Analysis Summary report:

- Some IMS log records can be large.
- Many production IMS systems can produce thousands of log records in seconds.
- If you provide specific filtering criteria, report results are easier to use and manage.

Example

The following example shows a simple way to perform an initial analysis on a log data set:

```
ANALYZE
  SLDS=(ims.log.data.set.name,Rvr)
END
```

In the preceding example, the only specification is the name of the log data set to be analyzed. Because the REPORTS control statement and all other output data set statements are omitted, Log Analyzer generates the Analysis Summary report by default. This report contains general information about the log contents, including these fields:

- Time span of the log
- Counts of records and logical units of work (LUOW)
- Lists of LUOWs by specific origin name, destination name, LUOW type (such as checkpoint), and user ID
Example
The following two-part example shows a more complex request for batch reports. The first analysis job shows a search for records that reflect the work which two users performed on the IMSA system on April 24, 2007:

```
//JOB1 ...
ANALYZE
  RECON=(RCNDPK.IMSA.RECON1,RCNDPK.IMSA.RECON2,
          RCNDPK.IMSA.RECON3)
  INTERVAL START=2007114/ STOP=2007115/
  FILTER
    SELECT=((USERID=PINK) OR (USERID=BROWN))
  REPORTS
    SUMM=ALL
    LSUMM=ALL
    LDET=ALL
END
```

An auditor examines the output of the first analysis job and requests another execution with greater detail about a specific time range and types of activities. The second analysis job includes a more restrictive INTERVAL control statement and the request for a detailed report:

```
//JOB2 ...
ANALYZE
  RECON=(RCNDPK.IMSA.RECON1,RCNDPK.IMSA.RECON2,
          RCNDPK.IMSA.RECON3)
  INTERVAL START=2007114/13450000 STOP=2007114/16520000
  FILTER
    SELECT=((USERID=PINK) OR (USERID=BROWN))
  REPORTS
    SUMM=ALL
    LSUMM=ALL
    LDET=ALL
END
```

Specifying log record codes for batch reports

You can specify the types of log records to include in or exclude from the overall analysis process by specifying the LOGRECORDCODES keyword on the FILTER control statement.

By default, all log record types that are included in an analysis job are included in the batch reports for that job.

You can specify a set of log record types to include in or exclude from batch reports by specifying the LOGRECORDCODES keyword on the REPORTS control statement. However, a specified log record type must be included on the FILTER control statement to be included in batch reports.
Example

The following example shows the use of different sets of log record codes for filtering and for the production of batch reports:

```
ANALYZE
   SLDS=(IMS.SLDS.DATASET,R101)
FILTER
   LRC=(ALL, ¬48, ¬43)
REPORTS
   SUMM=ALL
   LSUMM=ALL
   LDET=ALL
   LRC=(01,03,50)
END
```

For more information, see “Specifying FILTER control statements” on page 83.

Including data from type 01 and type 03 records

Thousands of occurrences of a single transaction can be logged to a specific log data set from a production IMS system. Finding a problematic occurrence within the many that completed correctly can be difficult. If you can identify specific data that was submitted for the problematic occurrence, you can use this data in a content search with Log Analyzer.

Example

The following example shows a search for the character string James Carson at specified offsets in type 01 and type 03 records:

```
ANALYZE
   RECON=(RCNDPK.IMSD.RECON1,RCNDPK.IMSD.RECON2)
      RCNDPK.IMSD.RECON3)
   INTERVAL START=2007190/08000000 STOP=2007190/08182356
FILTER
   LRC=(01, 03)
   SELECT = (CONTENT = (C'James Carson',01,120,12)
      or CONTENT = (C'James Carson',03,88,12))
REPORTS
   SUMM=ALL
   LSUMM=1200
END
```

Specifying the format of dumped log records

Log Analyzer provides two formats for dumped log records: standard format (requested with the DUMPREC keyword) and vertical “over-and-under” format (requested with the DUMPVERT keyword).

If you request a dump in vertical format, you can use the DUMPFORM keyword to control whether the dumped information contains character representations, hexadecimal representations, or both representations of the data.
You might want to try the different formats (in separate jobs because the keywords are mutually exclusive) and data representations to determine which format you prefer.

**Example**

The following example shows a request to dump the first 100 lines of the report in vertical format.

```
ANALYZE
  SLDS=(LUI.CUSTOMER.SLDS1,R91)
  LRCPDS=R1HJER.LUI.CNTL
FILTER
  SEL = (USERID = F177862)
INTERVAL STOP=2007222/00000100
REPORTS
  SUMM=ALL
  LSUMM=6000
  LDET=6000
  DUMPF=BOTH
  DUMPV=0000000100
END
```

---

**Working with extract files**

This section provides detailed conceptual information, advice, scenarios, and examples that are related to working with extract files.

**Overview of extract files**

During an analysis job, Log Analyzer can create an optional extract file that contains copies of all log records that have been associated with LUOWs of interest.

You can use the extract file for the following purposes:

- To obtain a subset of log records for further analysis, create an extract file and use it as input to a subsequent execution of the analysis job. Typically, less time and fewer resources are used in processing an extract file than are required for processing the original log record data sets. Sending a single extract file offsite is easier than sending multiple log record data sets. The analysis job can handle multiple input extract files.

- You can analyze IMS log data that was created at multiple sites by creating an extract file at each required local site and sending this file to the central site. When Log Analyzer processes all of these extract files in a single analysis job, all related message-switch data is combined into LUOWs that provide a single view of multiple IMS log records across multiple systems. This view is especially useful in MSC connected environments in which multiple message switches occur.
Certain functions of the ISPF interface, such as viewing dumps of log records during interactive analysis of LUOWs, require access to the original log records. Log Analyzer finds these log records in the extract file. You must create an extract file if you plan to use the ISPF interface for these functions.

### Allocating an extract file

If you specify the EXTRACT control statement in the analysis job, Log Analyzer allocates an extract file as a standard physical sequential data set.

Log Analyzer uses NOTE and POINT logic to access data in the extract file, so the extract file cannot be allocated on multiple volumes.

If the specified extract file does not exist in the system catalog, Log Analyzer allocates a new file (with the DISP=NEW parameter). For the new file, Log Analyzer uses the parameters that you specified with keywords on the EXTRACT control statement. See “Specifying EXTRACT, INDEXFILE, and WORKFILE control statements” on page 128.

If the specified extract file already exists, Log Analyzer reuses the existing file and overwrites the data in the file. The file is not reallocated; if an out-of-space condition occurs for the file, you must delete the file before rerunning the analysis job.

Creation of the extract file is optional. If the extract file is not available, certain functions of the ISPF interface are not available.

If an analysis job creates an extract file without creating an index file, the job must create the LUOW Detail report. (You must specify the LUOWDETAIL keyword on the REPORTS control statement.) Otherwise, Log Analyzer cannot populate the extract file.
Example

The following example shows the creation of an extract file that contains records which document the actions of user BMCABC on Julian date 07200, between 8:00 and 10:00 in the morning:

```
ANALYZE
  SLDS=(BMC.IMS8.SLDS,R81)
FILTER
  SEL=USERID = BMCABC
INTERVAL
  START=2007200/08000000
  STOP=2007200/10000000
EXTRACT
  DSN=BMC.LUI.EXTRACT.BMCABC.J07200
  UNIT=SYSDA
  VOLS=T0000A
  PRISP=300
  SECSP=120
  SPU=CYL
END
```

Example

If you always want Log Analyzer to allocate a new extract file, your analysis job should include a step that deletes an existing extract file. Insert this step before the analysis step, as shown in the following example:

```
//LUI#JCL3 JOB
  ...
//DELETE   EXEC PGM=IEFBR14
//EXTRACT  DD DSN=USER.LUI.EXTRACT,DISP=(MOD,DELETE,DELETE),
//         SPACE=(TRK,(1,1))
//*
//ANALYZE   EXEC PGM=LUIMAIN,REGION=0M
  ...
//SYSIN     DD *
  ...
EXTRACT DSN=USER.LUI.EXTRACT ...
```

Example

The extract file can be created as a generation data group (GDG) data set, as shown in the following example:

```
EXTRACT DSN=USER.LUI.EXTRACT(+01) ...
```

Using an extract file through the ISPF interface

When you are working with the results of an analysis job through the Log Analyzer ISPF interface, the interface obtains access to the log records in the extract file through entries in the index file.

For more information about the index file, see “Working with index files” on page 300.
Using an extract file as input to another analysis job

You can use an extract file that was created by one analysis job as input to another analysis job.

**Example**

In the following example, the extract file that was created in “Allocating an extract file” on page 297 becomes the input to an analysis job. (The work file is required for sorting the records in the input extract file.)

```
ANALYZE
  INPEXT=BMC.LUI.EXTRACT.BMCABC.J07200
REPORTS
  SUMM=ALL
  LSUMM=ALL
  LDET=000006000
  LRC=(03,3*,4*,D*)
WORKFILE
  PRISPACE=20
  SECSPACE=20
  SPACEUNITS=CYL
  VOLSER=SYSDA
END
```

Using the Reformat Extract File utility

You can use the Reformat Extract File utility (program LUIRFEXT) to prepare an extract file for use by processes other than Log Analyzer processes. This utility strips out Log Analyzer headers and data.

The following figure shows JCL for executing the Reformat Extract File utility. Sample JCL is located in member LUI#FEXT of the Log Analyzer control library.

**Figure 75: Reformat Extract File utility (program LUIRFEXT) JCL**

```
//LUIRFEXT JOB ...
//RFEXT EXEC PGM=LUIRFEXT,REGION=0M,TIME=04
//STEPLIB DD DISP=SHR,DSN=bmclui.loadlib
//          DD DISP=SHR,DSN=bmclui.BBLINK
//          DD DISP=SHR,DSN=bmclui.ICOLIB
//LOGIN DD DISP=SHR,DSN= BMC.LUI.EXTRACT.BMCABC.J07200
//LOGOUT DD DISP=(,CATLG),DSN=BMC.LUI.REFORMAT.J07200,
//          UNIT=ALLSYSDA, VOL=SER=DEVXXX,
//          DCB=RECFM=VB,LRECL=16376,BLKSIZE=16380,
//          SPACE=(TRK,(150,100))
//SYSOUT DD SYSOUT=* /*
```

To reformat an extract file

1. Specify the JCL to execute the Reformat Extract File utility:
   a. Specify the EXEC statement to execute program LUIRFEXT.
b Specify the STEPLIB statement to allocate the library that contains the Log Analyzer load modules.

c Specify the LOGIN statement to allocate the input existing Log Analyzer extract data set to be prepared for use by other processes.

d Specify the LOGOUT statement to allocate an output data set to contain the reformatted data from the extract data set. Use parameter values that are typical for log data sets.

e Specify the SYSOUT statement to allocate a SYSOUT-type data set to contain information (such as utility messages) that is generated during execution of the utility.

2 Submit the JCL for execution.

3 Check the results of the job to ensure that it completed successfully.

Working with index files

This section provides detailed conceptual information, advice, scenarios, and examples that are related to working with index files.

Overview of index files

During an analysis job, Log Analyzer can create an optional index file to contain information about the LUOWs that Log Analyzer has created to organize log records and about the log records that have been selected for inclusion in LUOWs.

The index file has a single purpose: to allow access to analysis data when you are interactively analyzing LUOWs through the interface.

Allocating an index file

If you specify the INDEXFILE control statement in the analysis job, Log Analyzer allocates an index file as a VSAM key-sequenced data set (KSDS).

If the specified index file does not exist in the system catalog, Log Analyzer allocates a new file (with the DISP=NEW parameter). For the new file, Log Analyzer uses the parameters that you have specified on the INDEXFILE control statement.
“Specifying EXTRACT, INDEXFILE, and WORKFILE control statements” on page 128.

If the specified index file already exists, Log Analyzer reuses the existing file and overwrites the data in the file. The file is not reallocated; if an out-of-space condition occurs for the file, you must delete the file before rerunning the analysis job.

Creation of the index file is optional. If the index file is not available, interactive analysis of LUOWs through the ISPF interface is not available. If you request an index file, you must also request an extract file.

If you want to use the ISPF interface to analyze the contents of an extract file, you must request the creation of an index file during the analysis job that created the extract file. If you have created an extract file and did not create a matching index file, or if the matching index file is lost or damaged, you can create a new index file by using the existing extract file as input to an analysis job and requesting the creation of an output extract file and an index file.

Example

The following example shows the creation of an extract file and an index file:

```
ANALYZE
   SLDS=(BMC.IMS8.SLDS.R81)
   LRCPDS=BMC.LUI.CNTL
FILTER
   SEL=USERID = BMCABC
INTERVAL
   START=2007200/08000000
   STOP=2007200/10000000
EXTRACT
   DSN=BMC.LUI.EXTRACT.BMCABC.J07200
   UNIT=SYSDA
   VOLS=T0000A
   PRISP=300
   SECSP=120
   SPU=CYL
INDEXF
   DSN=BMC.LUI.INDEX.BMCABC.J07200
   UNIT=SYSDA
   VOLS=T0000B
   PRISP=300
   SECSP=120
   SPU=CYL
END
```
Example

If you always want Log Analyzer to allocate a new index file, your analysis job should include a step that deletes an existing index file. Insert this step before the analysis step, as shown in the following example:

```
//LUI#JCL3 JOB
.../
//DELETES  EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=* */
//SYSIN DD * */
DELETE USER.LUI.INDEX /*
//ANALYZE EXEC PGM=LUIMAIN,REGION=OM /*
//SYSIN DD */
... INDEXFILE DSN=USER.LUI.INDEX ...
```

Using an index file

To use the index file that has been created by an analysis job, specify the data set name of the file in the Report index file field on the Report Index File Analysis panel.

Multiple users can use the same index file at the same time. For more information, see “Working with reports in the ISPF interface” on page 211.

Interpreting information in reports

This section provides detailed information and examples that are related to interpreting LUOW information which Log Analyzer provides in batch reports and in the ISPF interface.

Types of LUOWs

Log Analyzer reports the following types of LUOWs:

- Transaction LUOWs
- Miscellaneous LUOWs
- Nontransaction LUOWs
Transaction LUOWs

A transaction typically has an origin (a source, such as an LTERM), a destination (such as another LTERM), and an originating user.

A special type of transaction LUOW represents activity from a non-message-driven BMP. CICS transactions that are processed in an IMS DB/CTL system appear as non-message-driven BMPs. This type of LUOW does not include type 01 records or type 03 records and has no identifiable destination or user ID. In reports for this type of LUOW, Log Analyzer uses the label NMSGBMP as the destination and the user ID and uses the PSB name as the origin.

For a list of log record types that Log Analyzer associates with transaction LUOWs, see “Log records types in transaction-type LUOWs” on page 366.

Miscellaneous LUOWs

A miscellaneous LUOW has no identifiable origin, destination, or user ID.

This type of LUOW is often the result of work that IMS systems must perform to move data between systems in a shared-queue environment.

A miscellaneous LUOW might also result from a traditional transaction that did not contain a type 01 record or a type 03 record, possibly because the record was written to a log that was not included in the input to the analysis job or because the record had a timestamp that was not within the specified range for analysis.

A miscellaneous LUOW is categorized as an "other" LUOW on Log Analyzer reports.

Nontransaction LUOWs

Nontransaction LUOWs represent events that are not directly related to transactions.

These events include users signing on to the system and signing off from the system, databases being opened and closed, databases taking extents, system checkpoints being taken, and so on. IMS writes a single log record to track most of these events (IMS can write many log records for some events). Log Analyzer assigns each event to a separate LUOW.

A nontransaction LUOW is categorized as an "other" LUOW on Log Analyzer reports.

For a glossary of nontransaction LUOWs types that Log Analyzer can create, see “Nontransaction and other LUOW types” on page 367.
**LUOW numbers and UOW-1 numbers**

During an analysis job, Log Analyzer assigns an arbitrary number to each LUOW that is created.

The numbers are assigned in sequence (the first LUOW that Log Analyzer creates is LUOW 1, the second is LUOW 2, and so on). The LUOW number is used to identify a specific LUOW in all reports from a single run of the analysis job. However, LUOW numbers are not persistent; if you rerun the analysis job with a different set of data or filter criteria, a different LUOW is likely to be assigned to a particular LUOW number.

In contrast, the UOW-1 number remains static between runs of the analysis job. Log Analyzer obtains or assigns a UOW-1 number from the first record that was associated with the LUOW as follows:

- For transaction-type LUOWs, IMS assigns the UOW-1 number. The UOW-1 number consists of the 8-byte IMSID of the originating IMS system, followed by a 16-byte hexadecimal representation of the timestamp of the record.

- For other types of LUOWs, Log Analyzer assigns a UOW-1 number. This number consists of the type of LUOW, followed by a 16-byte hexadecimal representation of the timestamp of the record.

You can use the UOW-1 number to identify a specific LUOW in reports from different runs of the analysis job.

**Variable data snipped from log records**

When Log Analyzer provides details about log records (in batch reports and the ISPF interface), whenever possible it reports relevant data from within the log record.

Log Analyzer "snips" relevant data by mapping relevant fields in the log record and saving the data from the fields as the record is read. Relevant data depends on the type of the log record.

For example, relevant data for type 03 records includes the IMSID of the originating IMS system, the IMSID of the destination IMS system, and so on. Because relevant data is not constant across all types of log records, it is displayed in a Variable data field on reports and ISPF panels. Depending on the available space in the report or on the panel, the data has a short or long label to identify the field.

For some types of log records, no data was considered to be important to snip. In this case, reports and ISPF panels display the following string: *(no formatted data)*

For a glossary of variable data types, see “Variable data types” on page 374.
Unresolved associations field

Unresolved associations can occur during the process of creating LUOWs.

Log Analyzer reports the count of in the **Unresolved associations** field on the Analysis Summary report.

In most cases you can ignore this condition, or you can try to resolve it by rerunning the analysis job with specifications that cause a wider range of records to be considered for selection.

For more information, see “Unresolved associations when creating LUOWs” on page 282.

Refining the results of automated analysis

If the automated analysis job did not produce the results that you need, you can change your control statement specifications and resubmit the job.

The following table shows some common results and the changes that you can make to refine them.

**Table 30: Refining the results of automated analysis**

<table>
<thead>
<tr>
<th>Result</th>
<th>Suggested control statement changes</th>
</tr>
</thead>
</table>
| Too many log records were selected. | ■ Specify a more narrow range of time (or log sequence numbers) by using the START keyword and the STOP keyword on the INTERVAL control statement.  
■ Specify fewer types of log records to analyze by using the LOGRECORDCODES keyword on the FILTER control statement.  
■ Specify more specific search criteria by using the SELECT keyword on the FILTER control statement. |
| A specific LUOW or log record that you need to see is not included in the analysis. | ■ Specify a more broad range of time (or log sequence numbers) by using the START keyword and the STOP keyword on the INTERVAL control statement.  
■ Specify more generic search criteria by using the SELECT keyword on the FILTER control statement. |
### Result

| A type of log record that you want to see is not included in the analysis or in the batch reports. |  
|:---|:---|
| ![Check the LOGRECORDCODES keyword on the FILTER control statement. Ensure that the log record code is included in the specified value (or in the specified member of the log record code PDS) and not excluded in the specified value (or in the specified member of the log record code PDS).](#) |  
| ![Check the LOGRECORDCODES keyword on the REPORTS control statement. Ensure that the log record code is included in the specified value (or in the specified member of the log record code PDS) and not excluded in the specified value (or in the specified member of the log record code PDS).](#) |  
| Information from the analysis cannot be displayed in the ISPF interface. |  
| ![Include the EXTRACT control statement and the INDEX control statement to create the files that are required for viewing information through the ISPF interface.](#) |  
| An incorrect range of records was selected. |  
| ![Check the values of the START keyword and the STOP keyword on the INTERVAL control statement.](#) |  
| ![Check the value of the TIMEZONE keyword.](#) |  

Sometimes the problem that you are trying to solve is not clearly defined. In this case, you might need to run multiple analysis jobs, refining the results in successive jobs, to identify the problem exactly.
Example

The following example shows the use of this technique. An event occurred in the IMS system during the weekend of July 14, 2007, and caused a problem on a printer device (P112RT). The first job explores the broader time range and possible problem transactions:

```//JOB1 ...
ANALYZE
  RECON=(RCNPK.IMSD.RECON1,RCNPK.IMSD.RECON2)
INTERVAL
  START=2007195/
  STOP=2007197/
FILTER
  SELECT = (01DEST = P112RT or 03DEST = P112RT)
REPORTS
  SUMM=ALL
  LSUMM=6000
END
```

The second job narrows the time frame and produces dumps of problem records:

```//JOB2 ...
ANALYZE
  RECON=(RCNPK.IMSD.RECON1,RCNPK.IMSD.RECON2)
INTERVAL
  START=2007196/03300000
  STOP=2007196/04045959
FILTER
  SELECT = 01DEST = SLAM22
  SELECT = 03DEST = P112RT
REPORTS
  SUMM=ALL
  LDET=ALL
  DUMPR=6000
END```
Refining the results of automated analysis
Using Log Analyzer to diagnose problems and answer questions

You can use Log Analyzer to diagnose a variety of typical and atypical problems and answer a wide range of questions that can arise in IMS environments.

The examples and scenarios that are described in this chapter represent only a small fraction of the situations that you can handle efficiently and effectively with Log Analyzer:

- “Auditing user actions” on page 310 explains scenarios in which Log Analyzer provides information for internal and external audits.

- “Viewing database update activities” on page 312 explains scenarios in which Log Analyzer provides information for analyzing actions that have updated IMS databases.

- “Analyzing message-switching transactions and MSC traffic” on page 313 explains scenarios in which Log Analyzer provides information for analysis of message-switching transactions and traffic in systems that are participating in a multiple system coupling (MSC) network.

- “Filtering LUOWs by elapsed time” on page 319 explains scenarios in which Log Analyzer provides information for analysis of LUOWs, based on the overall amount of time that has elapsed for the LUOW.

- “Identifying abnormal terminations (abends)” on page 319 explains scenarios in which Log Analyzer provides information about LUOWs that have experienced abends.

- “Working with DL/I call trace information” on page 321 explains scenarios in which Log Analyzer can help you analyze the information that is returned from a trace of DL/I calls.

- “Obtaining information about application checkpoints” on page 323 explains how Log Analyzer can help you determine whether applications are issuing too many checkpoints.
If you are confronted with a situation that you are not sure about how to approach, you can contact BMC Customer Support for advice and guidance.

---

**Note**

The example reports in this chapter might differ slightly (for example, in layout) from the reports that you receive.

---

## Auditing user actions

Internal and external auditors might need to examine the actions that specific users (who are identified by their user IDs) or transactions have performed within an IMS system during a particular time period.

Log Analyzer can generate batch reports that auditors can analyze manually, and it can generate extract and index files that auditors can analyze interactively through the Log Analyzer ISPF interface.

The following example shows control statements for the collection of information for an audit. The auditor wants to see the actions that user ID F014389 performed during the afternoon and evening of January 9, 2007. The auditor prefers to use batch reports (instead of the ISPF interface) to analyze the collected information.

```plaintext
ANALYZE
  SLDS=(IMS.SYSTEM.SLDS,R81)
FILTER
  SEL = USERID = F014389
INTERVAL
  START=2007009/1200000
  STOP=2007010/0000010
REPORTS
  AUDIT=ALL
  SUMM=ALL
  LDET=ALL
END
```

The following example shows the Auditing report that was generated by the control statements from the previous example. During the target time range, the user signed off of two terminals.

<table>
<thead>
<tr>
<th>Log Analyzer for IMS V1.1.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOG time span: FROM 2006-257 16:45:00.6 TO 2006-257 17:14:44.3</td>
</tr>
<tr>
<td>TYPE</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>SIGNOFF</td>
</tr>
<tr>
<td>SIGNOFF</td>
</tr>
</tbody>
</table>

The following example shows control statements for the collection of information for an audit. The auditor wants to see the actions that user ID F014389 performed during the afternoon and evening of January 9, 2007. Because the auditor prefers to perform
detailed research on the collected information through the ISPF interface, output extract and index files will be created.

```
ANALYZE
  SLDS=(IMS.SYSTEM.SLDS,R81)
FILTER
  SEL = USERID = F014389
INTERVAL
  START=2007009/1200000
  STOP=2007010/0000010
REPORTS
  AUDIT=ALL
EXTRACT
  DSN=BMC.LUI.EXTRACT.JOB0028C
  UNIT=SYSDA
  STORC=DEVSMS
  PRISP=120
  SECSP=70
  SPU=CYL
INDEXFILE
  DSN=BMC.LUI.INDEX.JOB0028C
  UNIT=SYSDA
  STORC=DEVSMS
  PRISP=120
  SECSP=70
  SPU=CYL
END
```

The following example shows an audit request that searches for a transaction (TRAN02) instead of a user ID:

```
ANALYZE
  SLDS=(IMS.SYSTEM.SLDS,R81)
FILTER
  SEL = DEST = TRAN02
INTERVAL
  START=2007009/1200000
  STOP=2007010/0000010
REPORTS
  AUDIT=ALL
  SUMM=ALL
  LDET=ALL
END
```

The following example shows a portion of the Auditing report that is generated by the control statements from the previous example:

```
2007-267                       Log Analyzer for IMS V1.1.00
Auditing report (AUDIT   )
LOG time span: FROM 2006-257 16:45:00.6 TO 2006-257 17:14:44.3
TYPE     TIME             TARGET    LUOW#    Log seq #         Additional data
SIGNON   20062571646394   F474634   0000513  0000000001851EA5  term=GHA33320
COMMAND  20062571648424             0001115  0000000001854F09   STA  TRA  TRAN02 .
COMMAND  20062571648424             0001119  0000000001854F33   STA  TRA  TRAN02 .
COMMAND  20062571665167   HQ71C1E4  0001505  0000000001856E6B   DIS  NOD  HQGWYL* .
COMMAND  200625716650364  HQ71C1E4  0001573  000000000185733D   DIS  NOD  HQ84C1*. 
SIGNON   20062571650396   F474634  00001593  000000001857469  term=GHA33672
COMMAND  200625716650426  HQ71C1EE  0001610  000000001857577   DIS  NOD  HQ71C1*. 
COMMAND  200625716650459  HQ71C1E4  0001631  0000000018576C4   DIS  NOD  HQ84C1*. 
SIGNON   20062571652315   F474634  00002156  000000001859D88  term=GHB14869
SIGNOFF  20062571652412   F474634  00002204  00000000185A107  term=GHB14869
SIGNOFF  200625716653536   F474634  00002405  00000000185B2F2  term=GHA33672
```

Chapter 11 Using Log Analyzer to diagnose problems and answer questions 311
Viewing database update activities

Sometimes you need to identify the update activity that has occurred against an IMS database.

For example, you might need to diagnose the cause of database lock situations or find the reason why an excessive amount of processing has been performed against the database. Log Analyzer can provide this information.

**Example**

The following example shows control statements for the collection of information about the activities that occurred for a transaction (HA90810) on January 30, 2007, from 8:10 to 8:15 in the morning:

```
ANALYZE
  SLDS=(BMC.TEST019.SLDS,R91)
  LRCPDS=RIHJER.LUI.CNTL
FILTER
  SEL = DEST = HA90810
INTERVAL
  START=2007030/08100000
  STOP=2007030/08150100
REPORTS
  SUMM=ALL
  LSUMM=000001010
  LDET=0000003010
END
```

The following example shows a portion of an Analysis Summary report that was generated by the control statements from the previous example. The report shows that all of the activity against transaction HA90810 originated from the APPC/OTMA environment.
Analyzing message-switching transactions and MSC traffic

Log Analyzer can help to diagnose problems with transactions in message-switching environments and with MSC traffic.

This type of analysis is best performed by people who are familiar with the applications and the systems that are associated with the transaction.

A message-switching and MSC-traffic scenario

The examples in this section are based on the scenario that is shown in the following figure.

Figure 76: A message-switching and MSC-traffic scenario

Application program APPL0001 runs in a local IMS system that is identified by IMSID PR9F. This program creates four remote SMBs, which are transmitted over
MSC links to the remote IMS system that is identified by IMSID PR9M. The SMBs execute in remote system PR9M, and the output is routed back across MSC links to local system PR9F.

The following figure shows a selection of messages from the PR9F system. These messages show some of the MSNAMEs that can be seen in the reports.

**Figure 77: Messages from the local system in the scenario**

```
R 39;/DIS ASMT MSNAME ALL
IEE600I REPLY TO 39 IS;/DIS ASMT MSNAME ALL
DFS000I 5 LINK983 983 984 MSC983 PR9F
DFS000I 2 LINK681 782 781 MSC682 PR9F
DFS000I 7 LINK983 987 988 MSC987 PR9F
DFS000I 6 LINK983 985 986 MSC985 PR9F
DFS000I 4 LINK681 982 981 MSC684 PR9F
DFS000I 3 LINK681 882 881 MSC683 PR9F
DFS000I 1 LINK681 682 681 MSC681 PR9F
DFS000I 8 LINK983 989 990 MSC989 PR9F
DFS000I *2007262/124600* PR9F
A2 DFS996I *IMS READY* PR9F
```

---

**Analyzing transactions if data is available from a remote site**

The ideal way to trace message-switching transactions is to perform the following steps:

1. Collect data from each remote site by executing a log analysis job that creates an extract file, and forward the extract file to the local site.

2. At the local site, execute a log analysis job that uses the remote extract file as input and processes local log data sets.

This procedure generates comprehensive reports showing all records that were produced for the transactions.

If you cannot collect data from the remote site, Log Analyzer still produces useful information. See “Analyzing transactions if data is not available from a remote site” on page 317.

---

**Creating an extract file at the remote site**

The following figure shows an analysis job that is executed at the remote site.
The job processes the SLDS from the remote PR9M system and creates an extract file. This extract file is then sent to the local site for processing. Reports in this job are optional but can help to verify that the correct data has been collected.

Figure 78: Analysis job to create an extract file at the remote site

```
//REMOTE JOB ...
ANALYZE
  SLDS=(TME.RIP.PR9M.D07117.T0919538.P01,R91)
  FILTER
    LOGRECORDCODES=(01,03,07,08,16,31,33,35,36,50,56,63)
  INTERVAL
    START=2007117/09122222 STOP=2007117/10363535
EXTRACT
  DSN=TME.RIP.LUI.TP014EV9.EXTR01
  STORCLASS=DEVSMS
  UNIT=DISK
  PRISPACE=5
  SECSPACE=5
  SPACEUNITS=CYL
REPORTS
  SUMMARY=200
  LUOWSUMM=200
  LUOWDETAIL=1000
  LUOWDETAILDATA=(0,40)
END
```

Processing the remote extract file with local logs

The following figure shows an analysis job that is executed at the local site.

The job processes the SLDS from the local PR9F system and the extract file that was created from the remote PR9M system. Log records from both IMS systems are included in the reports.

Figure 79: Analysis job to process the remote extract file with local logs

```
//LOCAL JOB ...
ANALYZE
  SLDS=(TME.RIP.PR9F.D07117.T0918548.P01,R91)
  INPEXT=TME.RIP.LUI.TP014EV9.EXTR01
FILTER
  LOGRECORDCODES=(01,03,07,08,16,31,33,35,36,50,56,63)
INTERVAL
  START=2007117/09122222 STOP=2007117/10363535
REPORTS
  SUMMARY=200
  LUOWSUMM=200
  LUOWDETAIL=1000
  LUOWDETAILDATA=(0,40)
END
```

Analyzing the resulting comprehensive data

The report provides a comprehensive look at the transaction as it was running in the local system and the remote system. Comments that were added to explain the report (that were not part of the original report) are indicated as follows:
The following figure shows a portion of the LUOW Detail report that was produced by the job which is shown in “Processing the remote extract file with local logs” on page 315.
Analyzing transactions if data is not available from a remote site

You can analyze MSC traffic even if the data from one or more of the participating remote IMS systems is not available.

If you are analyzing data from a single IMS that is in a MSC network, you see log records for the transactions from the local system:

■ Log records show the messages or transactions being enqueued to the MSlink names.

■ Log records show the replies or messages being taken off of the queue from the MSlink name origin as the transaction came back to the originating IMS system.

Log Analyzer creates logical units of work (LUOWs) to associate the available records, but the LUOWs cannot show records from a remote IMS system if the records are not available. Log Analyzer does not highlight the fact that records are missing and or try to explain why they are missing, but people who are familiar with the application can recognize points at which the transaction switched to the other system and returned.
The following figure shows a portion of the LUOW Detail report that was generated only for the local PR9F system; data from the remote PR9M system is not available. (You can compare this report to the report in “Analyzing the resulting comprehensive data” on page 315 to see the differences when the remote data is available.) Comments that were added to explain the report (that were not part of the original report) are indicated as follows:

>>> comment <<<

Figure 81: Report data from the local analysis job
Filtering LUOWs by elapsed time

Log Analyzer is not a monitoring tool, but it can help answer basic questions about the elapsed time that was consumed during execution of LUOWs.

Elapsed time information can be helpful for identifying intermittent performance problems, long-running (and short-running) LUOWs, effects of application and system changes on response time, and similar issues.

Log Analyzer calculates the elapsed time for an LUOW by subtracting the timestamp of the first record that is included in the LUOW from the timestamp of the last record that is included in the LUOW. In contrast, a monitoring tool reports elapsed time for individual units of work and provides more granular values, such as times for application processing, queue waits, and so on.

For example, users of the IVTNX transaction have reported that sometimes this transaction seems to respond more slowly than usual. From previous experience, you know that this transaction typically is completed in less than a second. You submit an analysis job to find instances of IVTNX transactions that took longer than 1 second:

```
FILTER SELECT=(TRAN=IVTNX AND ETIME>1)
```

The LUOW Summary report indicates that the analysis job found two LUOWs for the IVTNX transaction with elapsed times greater than 1 second:

<table>
<thead>
<tr>
<th>LUOW</th>
<th>Origin</th>
<th>Dest</th>
<th>PSB</th>
<th>User</th>
<th>Recs</th>
<th>1st rec date/time</th>
<th>Elapsed</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000013 T0033 IVTNX DFSIVPX IRMT0033 0000029 2002.051-14:47:21.230684 00:00:03.209985 UPDT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0000193 T0033 IVTNX DFSIVPX IRMT0033 0000030 2002.051-14:47:32.372788 00:00:10.314566 UPDT ABEND=U0777</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From this report, you determine that one of the LUOWs experienced a user 777 abend, which indicates a deadlock situation and might explain a slower response time. If you need to do more research, you can examine the LUOW Detail report and the Log Record Detail report for these LUOWs to identify other possible issues.

Identifying abnormal terminations (abends)

Log Analyzer can identify abnormal terminations (abend) as follows:

- In the context of logical units of work (LUOWs)
- As individual incidents that have been recorded in input log records
Abends in LUOWs

To indicate that one or more abends have been detected in LUOWs for a specific element (origin name, destination name, user ID, or PSB), Log Analyzer places an asterisk following the count of LUOWs for the element on the Analysis Summary report.

With a manual scan of this report, you easily can see that a problem has occurred.

For an automated approach, you can run the analysis process with a filter that detects abended LUOWs. Specify the SELECT=ABENDS=YES keyword on the FILTER control statement, and request the Analysis Summary report and the LUOW Summary report:

```plaintext
FILTER
SELECT=ABENDS=YES
REPORTS
SUMMARY=(ALL)
LUOWSUMM=(ALL)
```

The LUOW summary report identifies each specific LUOW for which an abend has been detected. The Notes field contains the abend code that was detected.
Abends as individual incidents

Log Analyzer can produce the ABEND report that focuses exclusively on type 07 records in the input log sources.

To produce this report, Log Analyzer processes records individually and does not correlate them with LUOWs. You can specify a large volume of input records, and obtain fast results to identify transactions or programs that have terminated abnormally.

To request the ABEND report, specify the ABEND keyword on the REPORTS control statement. When you specify this keyword, you cannot request other reports, or specify filters with the SELECT keyword. For more information, see “ABEND keyword” on page 110.

Working with DL/I call trace information

Log Analyzer can help you analyze the information that is returned from a PSB trace of DL/I calls.

Because a large volume of trace information can be generated, you typically do not want to request tracing of DL/I calls from all PSBs. However, if you can identify a specific PSB that is involved in problem transactions, you can request a call trace for the PSB, run the transactions, turn off the trace, and then analyze the logs with Log Analyzer. Log Analyzer associates the DL/I call trace (type 5F) records with LUOWs to provide a comprehensive view of the DL/I calls and transaction activities.

To work with DL/I call trace information in Log Analyzer

1. Start tracing the calls that are issued by a PSB, and allow transactions to run until a sufficient number of calls have been issued by the PSB.

Type 5F records are written to the IMS log during this time range. You can make a note of the time range so that you can specify it in the Log Analyzer analysis job.
For more information about trace commands, see the documentation for your version of IMS.

2 Stop tracing the calls for the PSB.

3 Specify the JCL for the analysis job as follows:

- You can specify the START and STOP keywords on the INTERVAL control statement and specify the time range of the traced calls.

- Specify the SELECT keyword on the FILTER control statement to specify the PSB name.

- You can specify the LOGRECORDCODES keyword on the FILTER control statement to specify the record types to include and exclude in LUOWs. Log Analyzer automatically includes type 5F records in transaction LUOWs.

- Specify the LUOWDETAIL keyword on the REPORTS control statement to request the LUOW Detail report. You can specify other REPORTS keywords if you want to view other reports, but the LUOW Detail report contains the information from the type 5F records.

The following example shows analysis job control statements:

```
Example

ANALYZE ...
INTERVAL START=yyyyddd/hhmmssth
        STOP=yyyyddd/hhmmssth
FILTER
    LOGRECORDCODES=(1,3,7,8,18,3*,5*,¬59)
    SELECT=PSB=DFSSAM04
REPORTS ...
    LUOWDETAIL=(ALL)
```

4 Execute the analysis job.

5 View the reports.

The LUOW Detail report displays the type 5F records in context with other records that provide information about the activities that occurred during the LUOW:
Obtaining information about application checkpoints

Log Analyzer can report information about basic and extended application checkpoints that are issued by batch message processing (BMP) jobs. You can use this information as follows:

- To identify BMP jobs that are issuing application checkpoints too often or not often enough
- To identify candidate BMP jobs for tuning checkpoint frequencies and for implementing checkpoint-pacing services of the APPLICATION RESTART Control (AR/CTL) product from BMC
- To evaluate efforts to tune the frequency of checkpoints or implement checkpoint pacing

Overview of application checkpoints

Application checkpoints indicate the completion of a unit of work.

When a BMP job issues a checkpoint, participating systems commit changes to the records that the BMP job has updated. Committing the changes releases the locks on the updated records, making those records available for use by other programs. Tuning the frequency of checkpoints helps to ensure optimum data availability, application performance, and resource usage.

For many organizations, BMP jobs that issue checkpoints at inappropriate frequencies are a growing problem. These organizations are still running BMP jobs that were developed and tuned for mainframes that were manufactured years ago, not for today’s faster mainframes. For example, a BMP job that runs on a five-year-old mainframe...
old mainframe can take an hour to run and issue one checkpoint every five seconds. In contrast, the same program that runs on a newer mainframe can run in 15 minutes and issue 10 checkpoints per second, most of which are unnecessary.

Unnecessary checkpoints waste CPU cycles at a significant cost. Reducing the number of unnecessary checkpoints has saved hundreds of thousands of dollars for many organizations.

**Overview of the Application Checkpoint report**

Log Analyzer can produce the Application Checkpoint report to identify BMP jobs that fail to comply with optimum checkpoint-frequency rules.

The report provides the following information:

- Elapsed times of BMP jobs
- Number of checkpoints issued during those jobs
- Calculated frequency of checkpoints issued by the jobs

With this information, you can focus checkpoint tuning efforts where they will yield the greatest benefit.

Log Analyzer supports internally defined default filters and user-specified filters for the Application Checkpoint report.

**Default filters**

Default filters include all BMP jobs that are found in the input logs and include BMP jobs that have any checkpoint frequency.

The report sorts the frequencies into a minutes column and a seconds column, and it contains notes to flag certain exceptions. Using default filters offers the following advantages:

- Gives you an overall impression of checkpoint processing in your environment
- Helps to pinpoint BMP jobs that might be candidates for checkpoint tuning

The following example uses the default filters, as specified by APPCHECK=ALL in the REPORTS control statement. The example reports the following information:

- Numerous BMP jobs issued more than one checkpoint per second.
- Job CGPBG21J issued more than 100 checkpoints per second.
- Two BMP jobs updated database records but did not issue any checkpoints.
Log time span: FROM 2008-217 07:18:38.50 TO 2008-217 08:05:17.29 DURATION 00:46:38

<table>
<thead>
<tr>
<th>JOB</th>
<th>PSB</th>
<th>LUOW#</th>
<th>#CHKPTS/TYPE</th>
<th>DURATION</th>
<th>/MIN</th>
<th>/SEC</th>
<th>--------Exceptions--------</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGPBG00J</td>
<td>CGPGU00</td>
<td>0000268</td>
<td>SYMBOLIC</td>
<td>00:46:37</td>
<td>49.07</td>
<td>0.74</td>
<td>*** Less than 4 chkp / min</td>
</tr>
<tr>
<td>CGPBG00J</td>
<td>CGPGU00</td>
<td>0000612</td>
<td>SYMBOLIC</td>
<td>00:45:08</td>
<td>8.24</td>
<td></td>
<td>*** More than 1 chkp / sec</td>
</tr>
<tr>
<td>JMSCPUG</td>
<td>JCC810L</td>
<td>0002541</td>
<td>SIMPLE</td>
<td>00:03:59</td>
<td>50.00</td>
<td>0.83</td>
<td>*** No Checkpoints</td>
</tr>
<tr>
<td>NSRB0810</td>
<td>NSRG1150</td>
<td>0063034</td>
<td>SYMBOLIC</td>
<td>00:00:04</td>
<td>7.50</td>
<td></td>
<td>*** No Checkpoints</td>
</tr>
<tr>
<td>NSRB0812</td>
<td>NSRG1155</td>
<td>007948</td>
<td>SYMBOLIC</td>
<td>00:00:09</td>
<td>14.00</td>
<td></td>
<td>*** More than 1 chkp / sec</td>
</tr>
<tr>
<td>IMSCPUBS</td>
<td>ZCC810L</td>
<td>002541</td>
<td>SIMPLE</td>
<td>00:00:10</td>
<td>10.40</td>
<td></td>
<td>*** More than 1 chkp / sec</td>
</tr>
<tr>
<td>NSRB0811</td>
<td>NSRG1150</td>
<td>0000612</td>
<td>SYMBOLIC</td>
<td>00:00:04</td>
<td>7.50</td>
<td></td>
<td>*** More than 1 chkp / sec</td>
</tr>
<tr>
<td>NSRB0812</td>
<td>NSRG1155</td>
<td>007948</td>
<td>SYMBOLIC</td>
<td>00:00:14</td>
<td>23.00</td>
<td></td>
<td>*** More than 1 chkp / sec</td>
</tr>
</tbody>
</table>

User-specified filters

Log Analyzer supports the following user-specified filters for the Application Checkpoint report:

- **BMP filters** report information about selected BMP jobs, as identified by the job name or program specification block (PSB) name.
- **Checkpoint frequency filters** report information about included BMP jobs, based on how often the BMP job issues checkpoints.

**Example**

For the next example, assume that Log Analyzer filtered the report by using the following user-specified filters:

```
REPORTS
APPCHECK=(JOB = NSRB* CHKFREQ > 2 THRESH = SEC) ...
```

Based on the filtering criteria, the resulting report lists all BMP jobs that have names that start with NSRB and that issue more than two checkpoints per second:
Fields in the Application Checkpoint report

The Application Checkpoint report contains a single section that lists information about included BMP jobs.

The report lists BMP jobs in order by the timestamp of the first record that Log Analyzer found in the input logs for each BMP job. For more information, see “Information in the Application Checkpoint report” on page 207.

Requesting the Application Checkpoint report

You can request the Application Checkpoint report by specifying the APPCHECK keyword on the REPORTS control statement.

You can build Log Analyzer control statements manually or by using the Log Analyzer ISPF interface.

To request the Application Checkpoint report (manual method)

1 On the ANALYZE control statement, identify the source of the log data to be processed:

   - To specify the input log data explicitly, use the SLDS keyword:

     ```plaintext
     ANALYZE
     SLDS=(sldsName) ...
     ```

   - To specify the input log data implicitly, use the RECON keyword:

     ```plaintext
     ANALYZE
     RECON=(reconName) ...
     ```

2 If you specified the RECON keyword, add the INTERVAL keyword to specify the range of input log data to process. (If you used the SLDS keyword, including the INTERVAL keyword is optional.)

3 Enter a REPORTS control statement, and include the APPCHECK keyword to indicate the type of filters to use:

   - To create the report with default (internal) filters, use APPCHECK=ALL.
   - To use your own filters, use APPCHECK=yourCriteria.

For more information about the subkeywords that you can use for yourCriteria, see “APPCHECK keyword” on page 111.
**Note**
The filtering techniques that Log Analyzer uses to produce the Application Checkpoint differ from the techniques for producing most other reports. The FILTER control statement is invalid if you specify the APPCHECK keyword. The filtering techniques for the Application Checkpoint report accommodate a larger volume of input log records than the techniques for most other reports. The analysis job should take only a few minutes, even if the job is processing several day’s worth of data.

4 *(optional)* If you want to request the Analysis Summary report, include the SUMMARY keyword.

**Note**
The Analysis Summary report and the LUOW Summary report can provide supporting information to help with checkpoint-tuning efforts. Do not request any other reports. Other reports are likely to be too large to be useful.

5 *(optional)* If you want to request the LUOW Summary report, include the LUOWSUMM keyword.

**To request the Application Checkpoint report (ISPF interface method)**

1 On the ANALYZE Control Statement panel (and supporting panels), identify the source of the log data to be processed.

2 On the INTERVAL Control Statement panel, specify the range of input log data to process.

   If you specified the name of a RECON, the range is required. If you specified the name of an SLDS, the range is optional.

3 On the REPORTS Control Statement panel, select the **Application Checkpoint** field.

4 Use the **Application Checkpoint report options** fields to specify report options.

**Controlling checkpoint frequency with AR/CTL**

You can use the APPLICATION RESTART CONTROL for IMS product (AR/CTL) from BMC to control the frequency of application checkpoints automatically.

With the AR/CTL Checkpoint Pacing feature, AR/CTL can control the pacing (minimum) interval between checkpoints and bypass checkpoints that do not meet the defined pacing requirements. Checkpoint pacing allows the program to issue
requests for checkpoints at logical points during processing, without regard for performance issues. The program can also issue an explicit call to force a checkpoint, without regard for the pacing requirements.

You can define multiple checkpoint pacing classes. In each class, you can set up separate pacing options that depend on the current shift (such as prime, second, and third), on whether the program is executing with batch or online characteristics, or on other criteria that you define. You can use a pacing user exit routine to override or accept the pacing recommendation.

No application program changes are required to implement checkpoint pacing. For more information, see the AR/CTL documentation set.

You can use the Log Analyzer Application Checkpoint report to identify candidate BMPs for implementing checkpoint pacing. The report can also help you to determine initial values to set for checkpoint-pacing options and to evaluate whether those values are appropriate.
Troubleshooting and FAQs

This appendix provides information about handling problems and questions that might arise while you are using the BMC Log Analyzer component for IMS.

General troubleshooting procedures and tools

This section provides information about general tools that can help you to diagnose and solve Log Analyzer problems.

Contacting BMC Customer Support

Information about how to contact BMC Customer Support is located immediately after the title page of this document.

That section also describes general diagnostic information that you should gather before you contact Customer Support. In addition to this general information, you should have access to the IMS logs and the RECON data sets (if applicable) that Log Analyzer was analyzing when the problem occurred.

Using the Trace facility

If a problem occurs, BMC Customer Support might request that you enable the Log Analyzer Trace facility during execution of the analysis job.

To invoke the Trace facility, you provide an LUITRACE DD statement. This statement is usually specified as a SYSOUT=* data set.

Note

If the LUITRACE DD statement is present, the analysis job can generate a potentially large amount of output. You should use this statement only at the request of BMC Customer Support.
Obtaining information about Log Analyzer messages

Information about the causes of error and informational messages that Log Analyzer can issue and the actions that you can take in response is available in the messages manual and in the Log Analyzer ISPF interface.

You can access message information in the ISPF interface by using various methods. You can choose the Message command from the Help action bar menu, or you can enter the MESSAGES command or the MSGS command in the Command area of most panels. The Messages - Entry pop-up window is displayed.

**Figure 82: Messages - Entry pop-up window**

```
Specify the numeric portion of the BMCLUI message you wish to view.
Then press Enter.
BMCLUI Message number . . _______ (blank for selection list)
```

In the **BMCLUI Message number** field, you can enter blanks or the numerical digits of a BMCLUI message number:

- If you enter blanks in the field, the Messages - List panel is displayed.

**Figure 83: Messages - List panel**

```
File   Help
-----------------------------------------------
Log Analyzer                     Messages - List             Row 1 to 19 of 146
Command ===> ________________________________ Scroll ===> CSR
Enter / to view help for the message.
- 000501E Chainer error, RC=________, reason=________
- 000502E Syntax error, invalid usage of delimiter _________
- 000503E Syntax error, expecting a command, found __________
- 000504W Processing terminating after previous error
- 000505E Syntax error, multiple _________ keywords
- 000506E Syntax error, multiple _________ commands
- 000507E Syntax error, ANALYZE command should precede _________
- 000509E Syntax error, invalid keyword format _________
- 000510E Syntax error, keyword _________ has null value
- 000511E Syntax error, keyword _________ has invalid value _________
- 000512E Syntax error, keyword _________ has invalid offset value _________
- 000513E Syntax error, keyword _________ has invalid length value _________
- 000514E Syntax error, keyword _________ has too many operands
- 000515E I/O error reading __________
- 000516E Syntax error, unexpected end of data encountered while parsing _________
- 000517E Syntax error, service _________ failed, RC=________
- 000518E Syntax error, error found in PDS member _________
- 000519E Syntax error, keyword expected, found _________
```

On the Messages - List panel, enter a slash (/) to the left of a message number. The Message **nnnnnnn - Help** pop-up window is displayed to provide information about the message.

**Figure 84: Message **nnnnnnn - Help** pop-up window**

```
Message 000501E - Help
```
If you enter a valid number (representing the numerical characters of a Log Analyzer message) in the BMCLUI Message number field, the Message $nnnnnn$ - Help pop-up window is displayed for the specified message (and the Messages - List panel is bypassed).

To display the Message $nnnnnn$ - Help pop-up window (bypassing the Messages - Entry pop-up window and the Messages - List panel), you can enter the MESSAGE $nnnnnn$ command or the MSGS $nnnnnn$ command in the Command area of most panels. In these commands, $nnnnnn$ is optional and represents the numerical digits of the BMCLUI error message. For example, to view information about message BMCLUI186003E, enter MSG 186003.

To display a limited list of messages on the Messages - List panel, you can use a masked value for $nnnnnn$. For example, entering MSGS 18%34* could result in the following messages:

- BMCLUI189340E
- BMCLUI189341E
- BMCLUI189342E
- BMCLUI189343E

Troubleshooting analysis jobs

This section describes common problems and questions that are related to Log Analyzer analysis jobs.

An analysis job runs too long

An analysis job can run for a long time because it can be processing a large number of log records.

You can take the following actions to reduce the amount of time that the analysis job takes to run:
Specify a narrower time range by using the START keyword and the STOP keyword on the INTERVAL control statement.

Specify a smaller set of log record types to include in the analysis.

Explicitly provide log data set names to avoid delays that could be caused by waiting for access to the RECON data sets.

**An analysis job consumes too many resources**

The analysis job consumes as much memory and CPU cycles as it needs and as are available.

Depending on the number of input log records and on the specified filter criteria, this resource consumption could adversely affect other jobs that are running in the system.

You can take the following actions to limit resource usage during execution of the analysis job:

- Implement workload manager parameters, dispatch priorities, or both to ensure that the process receives the desired percentage of resources.

- Specify a narrower time range by using the START keyword and the STOP keyword on the INTERVAL control statement.

- Start out by producing only the Analysis Summary report and the LUOW Summary report (do not produce other reports). Use these reports to narrow the time range and add filters to focus on areas of interest before you request more detailed reports or produce extract and index files.

**Storage compression has failed**

To ensure that an adequate amount of virtual storage remains available for analysis processing, Log Analyzer can initiate storage compression.

If storage compression cannot release storage, the job step fails with a user abend:

```
10.56.48 JOB14179 $HASP373 GCBJOB STARTED - INIT 1 - CLASS Q - SYS SYSP
10.56.48 JOB14179 IEF403I GCBJOB - STARTED - TIME=10.56.48
10.57.20 JOB14179 @BMCLUI000655E LUISTOR0 abnormally terminated - code=U4090 reason=16
10.57.20 JOB14179 10.57.21 0.00.32 0.00.01 0016 GCBJOB ANALYZE
10.57.20 JOB14179 IEF404I GCBJOB - ENDED - TIME=10.57.20
BMCLUI000641I Storage compression entered due to record limits
BMCLUI000645I Storage compression exited
```
If the analysis job experiences a storage compression failure, you might be able to modify Log Analyzer control statements to relieve the problem. The following actions often provide storage relief:

- Request less log data to be processed by reducing the number of log data sets that are selected for processing, by adding a limiting time range, or by taking both of these actions.

- Limit the list of log record types and subtypes to be included as candidates for processing. If the LOGRECORDCODES=ALL keyword is specified or used by default, you can specify the LOGRECORDCODES keyword on the FILTER control statement to include specific log record codes of interest and exclude other codes.

- Specify more SELECT filters to refine the search for requested data. To identify filter values that might yield good search results, you can run the analysis job again, using your original control statements but requesting the production of only the Analysis Summary report (by specifying only the SUMMARY keyword on the REPORTS control statement). This report provides enough detail for you to determine the contents of the input log data and find values to use for SELECT filters.

- Request less detail in reports. The Analysis Summary report and the LUOW Summary report (requested by specifying the SUMMARY keyword and the LUOWSUMM keyword on the REPORTS control statement) often provide enough detail for solving a problem. If you need more detail than these reports provide, you can run the analysis job again and specify the LUOWDETAIL keyword to request details for a limited number of LUOWs.

- Do not request production of the index file and the extract file until the volume of data has been limited sufficiently. You can start out by analyzing a large volume of data and requesting only the Analysis Summary report and the LUOW Summary report. These reports can help you to focus the analysis on specific SELECT filters, time ranges, and log data sets. When the analysis is focused, you can run the analysis again and request production of the files.

If these actions do not help, you can contact BMC Customer Support for more advice and recommendations.

**Troubleshooting LUOW and log record selection**

This section describes common problems and questions that are related to selection of logical units of work (LUOWs) and log records.
An expected log record was not selected

If an expected log record was not selected during the analysis job, you can take the following actions:

- Ensure that the record type (the log record code) is specified for inclusion and is not specified for exclusion in the value of the LOGRECORDCODES keyword on the FILTER control statement (or in the log record code PDS member).
- Ensure that SELECT filters do not exclude the record.
  If two SELECT filter values are specified with a Boolean AND operator, the record is selected only if both conditions are true.
- Ensure that the specified time range or log sequence number range is correct, if specified.
- Ensure that SKIP and LIMIT values are correct, if specified.

One or more filters seem to be ignored

If Log Analyzer seems to have ignored a search filter that you have specified with the FILTER control statement, you can take the following actions:

- Ensure that Log Analyzer is producing output other than the Analysis Summary report. If the REPORTS control statement is not specified (or if only the SUMMARY keyword is specified on the REPORTS statement) and the EXTRACT control statement is not specified, Log Analyzer ignores the SELECT keyword on the FILTER control statement.
- If Log Analyzer seems to be ignoring a filter to exclude a log record code, the log record type might be required. For more information, see “Automatically included log record types” on page 274. You can use the LOGRECORDCODES keyword on the REPORTS control statement to exclude the record from the reports.

No log records were selected

Several conditions can cause Log Analyzer to select no records from provided log record sources:

- The requested types of log records (as specified with the LOGRECORDCODES keyword) are not present in the input log data set.
- The value that was specified for the SKIP keyword is past the end-of-file marker in the input log data set.

- The requested time range or log sequence numbers (as specified with the INTERVAL control statement) are not present in the log data set.

- The requested time ranges are not present in the log data set.

In all of these cases, Log Analyzer issues an error message indicating that no log records were selected for processing and ends with condition code 8.

In response, you should specify different values for control statement keywords and rerun the analysis job.

**Log records seem to be missing from an LUOW**

If you request type 01 and 03 records, Log Analyzer does not include type 11, 12, and 13 records (for conversational transactions) automatically, even though these record types could be considered as belonging to a transaction LUOW.

To select these record types, select them explicitly by specifying them as a value of the FILTER LOGRECORDCODES keyword (or by specifying them in a list of log record codes that is included in the analysis). Log Analyzer places them into a separate LUOW.

**Troubleshooting reports and interactive analysis**

This section describes common problems and questions that are related to Log Analyzer batch reports and the interactive analysis of LUOWs through the Log Analyzer ISPF interface.

**An expected log record is not displayed**

If an expected log record is not shown in batch reports, you can take the following actions:

- Ensure that the record was selected for analysis, as explained in “An expected log record was not selected” on page 334.
Ensure that the record type (the log record code) is specified for inclusion and is not specified for exclusion in the value of the LOGRECORDCODES keyword on the REPORTS control statement (or in the log record code PDS member).

Check the limit value of the keywords on the REPORTS control statement. If a log record that you expect to see is not included in a report, check the limit (first) value of the keyword that controls the report and make sure that the value is large enough to accommodate the records of interest. The value specifies the number of lines, not the number of records. In particular, the Record Dumps report displays multiple lines for each record, and the specified limit might be too small to include the expected record.

**Analysis results cannot be displayed in the ISPF interface**

If you cannot view the results of an analysis job in the ISPF interface, you can take the following actions:

- Ensure that the analysis job has created an extract file and a matching index file. If the analysis job has created an extract file but not an index file (or if the index file has been lost or damaged), the index file can be created by running a new analysis job that reads the existing extract file as input and writes an output extract file and index file. If the original analysis job did not create an extract file, you must add control statements to create an extract file and an index file and run the analysis job again.

- Ensure that you have provided the data set name of the correct index file on the Report Index File Analysis panel.

- Ensure that the index file was created with a version of Log Analyzer that is compatible with the version of the ISPF interface that you are using. If the versions are incompatible, you can re-create the index file by running a new analysis job that reads the existing extract file as input and writes a new output extract file and index file. The ISPF interface gives you the option to view control statements that were used to create the original files.

**An error occurs during allocation of output data sets**

Log Analyzer allocates all output data sets with the DISP=NEW parameter.

If you receive an error indicating that an output data set already exists or a data set name is a duplicate, ensure that all previously allocated data sets have been deleted or change the data set names in the JCL to ensure allocation of a unique data set name. Then resubmit the analysis job.
Sample and syntax reference

This section provides reference information about the samples provided for the BMC Log Analyzer component for IMS and a summary of the syntax of Log Analyzer control statements.

Samples

Log Analyzer provides samples that you can copy to a user library and modify for your own purposes. The data set names of the libraries that contain the samples depends on how Log Analyzer is installed in your environment. For more information, see the product installation guide.

Samples in the control library

The following table lists JCL samples that are provided in the Log Analyzer control library.

Table 31: Samples in the control library

<table>
<thead>
<tr>
<th>Member</th>
<th>Contents</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUI#CPUM</td>
<td>Password installation job</td>
<td>This member contains sample JCL for executing a job that installs a CPU authorization password. The password is required for execution of a Log Analyzer analysis job.</td>
</tr>
<tr>
<td>LUI#FEXT</td>
<td>Extract file reformat job</td>
<td>This member contains sample JCL for executing the Reformat Extract File utility (program LUIRFEXT).</td>
</tr>
<tr>
<td>LUI#JCL1</td>
<td>Analysis job</td>
<td>This member contains sample JCL for executing a Log Analyzer analysis job. This job obtains input control statements from an in-stream data set, which is specified with the following statement: SYSIN DD *</td>
</tr>
</tbody>
</table>
Member | Contents | Description
--- | --- | ---
LUI#JCL2 | Analysis job | This member contains sample JCL for executing a Log Analyzer analysis job. This job obtains input control statements from an external data set, which is specified with the following statement:

```
SYSIN DD DISP=SHR,DSN=BMCNODE.LUISAMP(LUISAMP1)
```  

LUI#JCL3 | Analysis job | This member contains sample JCL for executing a Log Analyzer analysis job. This job contains statements that allocate a Log Analyzer index file and a Log Analyzer extract file and statements to delete these files if they already exist. The job obtains input control statements from an in-stream data set.

---

**Samples in the sample library**

The following table lists control statement samples and other types of samples that are provided in the Log Analyzer sample library. You can copy a sample to a user library and modify the sample for your own purposes.

<table>
<thead>
<tr>
<th>Member</th>
<th>Contents</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUICI@00</td>
<td>ISPF interface CLIST</td>
<td>This member contains a sample CLIST that you can configure and use to execute the Log Analyzer ISPF interface.</td>
</tr>
</tbody>
</table>
| LUISAMPA | Simple control statements | This member contains control statements for an analysis job that performs the following actions:  
- Processes one explicitly specified SLDS  
- Filters log records based on abended LUOWs  
- Produces the Analysis Summary report  
- Produces the LUOW Summary report that identifies individual abends |
| LUISAMP1 | Simple control statements | This member contains control statements for an analysis job that performs the following actions:  
- Process one explicitly specified SLDS  
- Produces an Analysis Summary batch report |
| LUISAMP2 | Simple control statements | This member contains control statements for an analysis job that performs the following actions:  
- Considers specified types of log records as candidates for selection  
- Filters log records and LUOWs by a specified DBD name  
- Produces selected batch reports |
<table>
<thead>
<tr>
<th>Member</th>
<th>Contents</th>
<th>Description</th>
</tr>
</thead>
</table>
| LUISAMP3 | Simple control statements | This member contains control statements for an analysis job that performs the following actions:  
|          |                           | ■ Considers all types of log records as candidates for selection, excluding any type of checkpoint record  
|          |                           | ■ Filters log records and LUOWs by a specified user ID  
|          |                           | ■ Produces all available batch reports  
| LUISAMP4 | Complex control statements | This member contains control statements for an analysis job that performs the following actions:  
|          |                           | ■ Filters log records and LUOWs by a specified user ID and destination  
|          |                           | ■ Considers log records from various time intervals as candidates for selection  
|          |                           | ■ Allocates input log data sets dynamically by using entries in explicitly specified RECON data sets (for the IMSID systems in which the traffic of interest occurred)  
|          |                           | ■ Reports information (in the LUOW Detail batch report) about a subset of selected records  
| LUISAMP5 | Complex control statements | This member contains control statements for an analysis job that performs the following actions:  
|          |                           | ■ Filters log records and LUOWs by LUNAMEs that match specified masked values  
|          |                           | ■ Produce selected batch reports with a limit on the maximum lines of output  
|          |                           | ■ Reports a specified portion of user data from type 01 and type 03 log records  
| LUISAMP6 | Complex control statements | This member contains control statements for an analysis job that performs the following actions:  
|          |                           | ■ Filters log records and LUOWs by user IDs that match specified masked values  
|          |                           | ■ Skips a specified number of records in the input SLDS  
|          |                           | ■ Considers log records that were created during a specified time range as candidates for selection  
|          |                           | ■ Produces selected batch reports  

Appendix B  Sample and syntax reference 339
<table>
<thead>
<tr>
<th>Member</th>
<th>Contents</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUISAMP7</td>
<td>Complex control statements</td>
<td>This member contains control statements for an analysis job that performs the following actions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Filters log records and LUOWs by a specified content string of user data content and by a specified output destination</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Produces selected batch reports</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Snips a specified portion of user data from records</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Reads an extract file that was created by a previously executed analysis job</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Obtains a list of log record types to be included or excluded by reading a specified member of a specified log record code PDS</td>
</tr>
<tr>
<td>LUISAMP8</td>
<td>Complex control statements</td>
<td>This member contains control statements for an analysis job that performs the following actions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Allocates input log data sets dynamically by using entries in explicitly specified RECON data sets (for the IMSID systems in which the traffic of interest occurred)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Creates an index file and an extract file for later use by the Log Analyzer ISPF interface</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Considers log records from various time intervals as candidates for selection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Produces selected batch reports</td>
</tr>
<tr>
<td>LUISAMP9</td>
<td>Simple control statements</td>
<td>This member contains control statements for an analysis job that performs the following actions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Uses a specified SLDS and a specified extract file as input</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Considers all log record types as candidates for selection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Produces selected batch reports</td>
</tr>
<tr>
<td>LUISYNTX</td>
<td>Control statement syntax</td>
<td>This member contains a representation of all control statements, keywords, and valid values that can be specified for an analysis job.</td>
</tr>
<tr>
<td>LUIZAPPL</td>
<td>Application log record types</td>
<td>This member contains a list of log record types that are created during IMS application execution. The list can be used in a member of a log record code PDS to specify the types of log records to consider as candidates for selection.</td>
</tr>
<tr>
<td>LUIZDBAC</td>
<td>Database access log record types</td>
<td>This member contains a list of log record types that are created during IMS database access. The list can be used in a member of a log record code PDS to specify the types of log records to consider as candidates for selection.</td>
</tr>
<tr>
<td>LUIZEXCL</td>
<td>Exclusion list of log record types</td>
<td>This member contains a list of log record types that are excluded. The list can be used in a member of a log record code PDS to specify the types of log records to exclude from consideration as candidates for selection.</td>
</tr>
</tbody>
</table>
### Control statement summary

The following figure shows the valid control statement commands, keywords, and values that you can specify in the SYSIN data set for a Log Analyzer analysis job.

The required portion of the command or keyword is shown in uppercase bold type; the optional portion of the command or keyword is shown in lowercase regular type. The default value (if any) for a keyword is underscored.

This syntax is include in member LUISYNTX of the sample library. For details about how to use this syntax, see “Working with Log Analyzer jobs” on page 55.

**Figure 85: Log Analyzer control statement commands, keywords, and values**

```
* 
ANalyze
   AnalyzeOLDS=NO | YES
   DB2LOG=db2LogDataSet | DB2BSDS=db2BootstrapDataSet
   ELAPSEDTIME=YES | NO
   IMSID=ALL | imsid | (imsid1, imsid2, ...)
   INPExt=extractDataSet
   IPRJRNl=journalDataSet | IPRRPREFIX=journalPrefix
   LIMIT=999999999 | nnnnnnnnn
   LogRecordCodesPDS=logCodeDataSet
   MAXLogs=100 | 999 | nnn | (nnn, abendCode) | (nnn, ReturnCode)
   MQExt=mgExtractdataSet
   RECON=reconDataSet | (reconDataSet1, reconDataSet2, reconDataSet3)
   SKIP=0 | nnnnnnnnn
   SLDS=(sldsDataSet.Rnn.unit.volser)
   TIMEZone=LOCAL | GMT | ORIGINAL | (+hhmm) | (-hhmm)
* 
INTERval
   START=FIRST | yyyyddd/hhmmssth | logSequenceNumber
   STOP=LAST | yyyyddd/hhmmssth | logSequenceNumber
* 
FILTER
   CONTent0103=DATA | ALL
   LogRecordCodes=ALL | nn | n* | ¬nn | nnn | nnn* | ¬nnnn | member | (nn, n*, ¬nn, nnnn, nnn*, ¬nnnn, member, ...)
   SESelect=((subkeyword=value) AND (subkeyword=value))
```
OR (subkeyword NOT=value))

* valid operands for ETIME: < > => >= <= =

* SELECT subkeywords:

  - ABENDS=YES
  - APPC_LTERM=lterm
  - CICSTRAN=transaction
  - CLIENTID=clientId
  - CONTENT=(C' string', logrec, offset, length
  - DBD=(dbdName, dsId | dbdName,areaName | dbdName,areaName,rba)
  - DEST=destination
  - ETIME <operand> ssss | hh:mm:ss.nnnnnn | mm:ss.nnnnnn | ss.nnnnnn
  - LTERM=lterm
  - LUNAME=luName
  - MPR=jobName
  - NODE=node
  - OIMS=originatingIms
  - ORIGIN=source
  - ORIGINPort=nnnn
  - PSB=psbName
  - PST=pstId
  - RDBD=(ddbName)
  - TMEMBER=tmemberName
  - TPIPE=tpipeName
  - TRAN=transaction
  - UDBD=(dbdName, dsId | dbdName,areaName | dbdName,areaName,rba)
  - UOW1=’characterImsID16characterUowID’,
  - USERID=userId

* REPORT

  - ABEND=ALL | nnnnnnnnn
  - APPCHECK =ALL | expressions
  - DEADLOCK=ALL | nnnnnnnnn | expressions
  - DUMPForm=BOTH | CHAR | HEX
  - DUMPRec=((ALL | nnnnnnnnn), offset, length)
  - DUMPVert=((ALL | nnnnnnnnn), offset, length)
  - LogRecordCodes=ALL | nn | n* | ¬nn | nnn | nnn* | ¬nnnn | member |
  - (nn,n*, ¬nn,nnnn,nnn*, ¬nnnn,member, ...)
  - LuowDETAIL=ALL | nnnnnnnnn | (LUOW|LuowId,LUOW|LuowId,...)
  - LuowDETAILData=(offset, length)
  - LuowSUMM=ALL | nnnnnnnnn
  - LuowSeq=ALL | nnnnnnnnn
  - OFFSET=HEX | DECIMAL | DEC
  - ORPHANS=ALL | nnnnnnnnn
  - RBASTATS=ALL | TOTALS
  - SUMMARY=ALL | nnnnnnnnn

* EXTRACT

  - DCB=dataSetName
  - DAClass=className
  - DSN=dataSetName
  - LIKE=dataSetName
  - MGMTClass=className
  - PRISPace=nnnnnnnn
  - SECSPace=nnnnnnnn
  - SpaceUnits=CYL | TRK
  - STORClass=className
  - VOLS=volser
  - UNIT=unitName

* INDEX File

  - DAClass=className
  - DSN=dataSetName
Control statement summary

```plaintext
MGMTClass=className
PRISPace=nnnnnnnn
SECSpace=nnnnnnnn
S PaceUnits=CYL | TRK
STORClass=className
VOLSer=volser

WORKFile
  MGMTClass=className
  STORClass=className
  DATAClass=className
  VOLSer=volser
  UNIT=unitName
  PRISPace=nnnnnnnn
  SECSpace=nnnnnnnn
  S PaceUnits=CYL | TRK

END
```

Appendix B  Sample and syntax reference 343
Log record and LUOW reference

This appendix provides information about the log records and logical units of work (LUOWs) that the BMC Log Analyzer component for IMS handles.

IMS log record types

Log Analyzer supports all types of log records that conform to IMS log record conventions.

Log Analyzer has mapped most IMS and BMC log records. A mapped record has a recognized log record code. Log Analyzer assigns a type and description that identify the record in batch reports and on interface panels. If Log Analyzer has not mapped a type of log record, the string `unmapd` is displayed as the short name of the record.

The following table lists the types of IMS log records that Log Analyzer has mapped, the type of LUOW to which the record is associated, and (for nontransaction types of LUOWs) the label that Log Analyzer uses for the type of LUOW in reports.

---

**Note**

Information about log record types is as current as possible, as of the time that this document was prepared for publication. Log record types and layouts can be changed at any time at the discretion of the owner of the log record.

Information about log records that Log Analyzer has mapped is also available in the Log Analyzer ISPF interface. To access this information, enter the LC global command from any panel or select **Log code help** from the Help action bar choice.
### Table 33: IMS log record types that Log Analyzer has mapped

<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>MSGIN</td>
<td>IMS input message</td>
<td>Transaction</td>
<td>A type 01 record is written when a data communication routine places transaction data in a message queue buffer. All filters except DBD, PSB, ABENDS, ETIME, MPR, and PST filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>02</td>
<td>CMDI</td>
<td>Condensed command - Type I</td>
<td>Nontransaction (command)</td>
<td>A type 02 record is written at successful completion of a /LOG command or a command that updates restart data. USERID and NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;command&quot;) type of LUOW.</td>
</tr>
<tr>
<td>03</td>
<td>MSGOT</td>
<td>IMS output message</td>
<td>Transaction</td>
<td>A type 03 record is written when a DL/I routine places transaction data in a message queue buffer. All filters except DBD, PSB, ABENDS, ETIME, MPR, and PST filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>04</td>
<td>RSR</td>
<td>RSR tracking</td>
<td>Nontransaction (rsr)</td>
<td>A type 04 record is written by Remote Site Recovery (RSR) for usage with remote IMS systems. Log Analyzer relates this type of record to a nontransaction (&quot;rsr&quot;) type of LUOW.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------</td>
<td>-------------</td>
<td>-----------------------------</td>
<td>----------</td>
</tr>
</tbody>
</table>
| 06              | ACTN           | IMS internally initiated action | Nontransaction (action) | A type 06 record is written when any of the following events occurs:  
- A forced-end-of-volume (FEOV) command was issued.  
- IMS was started or stopped.  
- One of the following events has occurred in an IMS XRF complex:  
  - IRLM failed.  
  - The VTAM TPEND exit routine was entered.  
  - A /SWITCH command was processed.  
- A /START command connected IMS to VTAM.  
- Data sharing was quiesced.  
Log Analyzer relates this type of record to a nontransaction ("action") type of LUOW. |
| 07              | APPLT          | Application terminate         | Transaction              | A type 07 record is written when an application program is terminated. PSB, MPR, and PST filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW.  
If the application terminates abnormally, the abend code is reported on the LUOW Summary report and the LUOW Detail report.  
This log record type is included in the ABEND report. |
<p>| 08              | APPLC          | Application start             | Transaction              | A type 08 record is written at startup of an application program. PSB and CICSTRAN filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW. |</p>
<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>09</td>
<td>BSTAT (SUMM, DETL)</td>
<td>Sequential buffering statistics</td>
<td>Nontransaction (stats)</td>
<td>A type 09 record is written at termination of an application program that might be using sequential buffering. Log Analyzer relates this type of record to a nontransaction (&quot;stats&quot;) type of LUOW.</td>
</tr>
<tr>
<td>0A</td>
<td>CPICI</td>
<td>CPI-CI driven program start/terminate</td>
<td>Transaction</td>
<td>A type 0A record is written at startup or termination of a CPI-communications driven program. It is similar to a type 07 or 08 record. Log Analyzer relates a type 0A record to a transaction LUOW if possible. If not, Log Analyzer displays the record as unmapped.</td>
</tr>
<tr>
<td>0F</td>
<td>LGLOG</td>
<td>Logical logger</td>
<td>Nontransaction (lelogrec)</td>
<td>A type 0F record is written by the IMS Logical Logger to track DECB contents. Log Analyzer relates this type of record to a nontransaction (&quot;lelogrec&quot;) type of LUOW.</td>
</tr>
<tr>
<td>10</td>
<td>SVIOL</td>
<td>Security violation</td>
<td>Nontransaction (sec viol)</td>
<td>A type 10 record is written when a security violation occurs. USERID and NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;sec viol&quot;) type of LUOW.</td>
</tr>
<tr>
<td>11</td>
<td>CONVS</td>
<td>Start conversation</td>
<td>Nontransaction (conv)</td>
<td>A type 11 record is written at startup of a conversational program. USERID and NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;conv&quot;) type of LUOW.</td>
</tr>
<tr>
<td>12</td>
<td>CONVE</td>
<td>End conversation</td>
<td>Nontransaction (conv)</td>
<td>A type 12 record is written at termination of a conversational program. USERID and NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;conv&quot;) type of LUOW.</td>
</tr>
</tbody>
</table>

IMS log record types

BMC Log Analyzer for IMS User Guide
<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
</table>
| 13             | CONVC          | Conversation control block   | Nontransaction (conv)        | A type 13 record is written to contain conversation control blocks (CCBs) for any of the following events:  
   - Logon for a static non-ISC terminal  
   - Signon for an ETO user  
   - Static ISC allocation  
   USERID and NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction ("conv") type of LUOW.                                                                                                                                                                                                                          |
<p>| 14             | SWDIS          | Switch line disconnect       | Nontransaction (dialline)    | A type 14 record is written at disconnection of a dial line. Log Analyzer relates this type of record to a nontransaction (&quot;dialline&quot;) type of LUOW.                                                                                                                                                                                                                                                                                 |
| 15             | SWCON          | Switch line connect          | Nontransaction (dialline)    | A type 15 record is written at connection of a dial line. Log Analyzer relates this type of record to a nontransaction (&quot;dialline&quot;) type of LUOW.                                                                                                                                                                                                                                                                                 |
| 16             | SIGN           | Sign on; Sign off            | Nontransaction (signonof)    | A type 16 record is written at successful completion of a /SIGN command. USERID and NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;signonof&quot;) type of LUOW.                                                                                                                                                                                                                             |
| 18             | XCHK           | Extended checkpoint          | Transaction                  | A type 18 record is written when an application program has established an IMS extended checkpoint environment by issuing the XRST call call and the CHKP call. Log Analyzer relates this type of record to a transaction type of LUOW.                                                                                                                                                                                                                             |
| 20             | DBOPN          | Database open                | Nontransaction (open/cls)    | A type 20 record is written when a full-function database is opened. DBD and DSID/area filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;open/cls&quot;) type of LUOW.                                                                                                                                                                                                                             |
| 21             | DBCLS          | Database close               | Nontransaction (open/cls)    | A type 21 record is written when a full-function database is closed. DBD and DSID/area filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;open/cls&quot;) type of LUOW.                                                                                                                                                                                                                             |</p>
<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>CMDII</td>
<td>IMSPLEX command</td>
<td>Nontransaction (command2)</td>
<td>A type 22 record is written at successful completion of an IMSPLEX type 2 command, indicating that the command is recoverable across a restart. Log Analyzer relates this type of record to a nontransaction (&quot;command2&quot;) type of LUOW.</td>
</tr>
<tr>
<td>24</td>
<td>DBERR (DBERW)</td>
<td>Database error</td>
<td>Nontransaction (db error)</td>
<td>A type 24 record is written when a buffer handler detects an I/O error on a database. DBD filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;db error&quot;) type of LUOW.</td>
</tr>
<tr>
<td>25</td>
<td>EEQE</td>
<td>EEQE created; EEQE deleted</td>
<td>Nontransaction (eeqe)</td>
<td>A type 25 record is written when an EEQE is created or deleted for a database. DBD filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;eeqe&quot;) type of LUOW.</td>
</tr>
<tr>
<td>26</td>
<td>IOTOL</td>
<td>I/O toleration buffer</td>
<td>Nontransaction (iot)</td>
<td>A type 26 record is written when an I/O toleration buffer is created. DBD filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;iot&quot;) type of LUOW.</td>
</tr>
<tr>
<td>27</td>
<td>XTEND</td>
<td>Database extended</td>
<td>Nontransaction (db exten)</td>
<td>A type 27 record is written when a database data set is extended. DBD and DSID/area filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;db exten&quot;) type of LUOW.</td>
</tr>
<tr>
<td>28</td>
<td>PH1DC</td>
<td>Phase 1 DC</td>
<td>Transaction</td>
<td>A type 28 record is written when the IMS restart facility updates sequence numbers of input messages for transactions from STSN devices. These transactions operate in response mode and are not Fast Path transactions. NODE filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>30</td>
<td>APRFX</td>
<td>Additional message prefix</td>
<td>Transaction</td>
<td>A type 30 record is written when a message prefix is changed. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>-------------</td>
<td>-----------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>31</td>
<td>MSGGU (GUDLI, GUCOM)</td>
<td>Message GET-UNIQUE</td>
<td>Transaction</td>
<td>A type 31 record is written when a get-unique (GU) is issued for a message. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>32</td>
<td>REJCT</td>
<td>Message reject</td>
<td>Transaction</td>
<td>A type 32 record is written when a message is rejected, causing an application program abend. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>33</td>
<td>FREE</td>
<td>Message free</td>
<td>Transaction</td>
<td>A type 33 record is written when the IMS queue manager releases a message. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>34</td>
<td>CANCL</td>
<td>Message cancel</td>
<td>Transaction</td>
<td>A type 34 record is written when a message is canceled. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>35</td>
<td>MGENQ (ENQ, REENQ)</td>
<td>Message enqueue</td>
<td>Transaction</td>
<td>A type 35 record is written when a message is enqueued or re-enqueued. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>36</td>
<td>MGDEQ (MGDEL, MGSAV)</td>
<td>Message dequeue</td>
<td>Transaction</td>
<td>A type 36 record is written when a message is dequeued, saved, or deleted. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>37</td>
<td>XFER</td>
<td>Message queue sync</td>
<td>Transaction</td>
<td>A type 37 record is written to indicate any of the following events:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- The sync point coordinator has determined that all resource managers have completed phase 1 commit processing.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- The DBCTL sync point processor received a phase 2 commit request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- One or more output messages have been transferred from a queue block to a permanent destination.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------</td>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>38</td>
<td>QRET</td>
<td>Queue return</td>
<td>Transaction</td>
<td>A type 38 record is written to indicate any of the following events:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>■ An input message was returned to an input queue because the application program terminated abnormally.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>■ The DBCTL sync point processor received an abort request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>■ A protected conversation is in doubt, and the input message has been moved to an RRE until the unit of work is aborted or committed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>■ A message has been returned to its original anchor block after QCF has terminated abnormally.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>39</td>
<td>QREL</td>
<td>Queue release</td>
<td>Transaction</td>
<td>A type 39 record is written when an output queue is released. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>3A</td>
<td>QFIXF</td>
<td>Queue record free</td>
<td>Nontransaction (qfixfree)</td>
<td>A type 3A record is written when a queue record was freed. Log Analyzer relates this type of record to a nontransaction (&quot;qfixfree&quot;) type of LUOW.</td>
</tr>
<tr>
<td>3B</td>
<td>QFIXR</td>
<td>Queue invalid message record</td>
<td>Nontransaction (qfixerr)</td>
<td>A type 3B record is written when an invalid message record or a nonrecoverable message response is detected during validation of a queue. Log Analyzer relates this type of record to a nontransaction (&quot;qfixerr&quot;) type of LUOW.</td>
</tr>
<tr>
<td>3C</td>
<td>QFIXB</td>
<td>Queue destination change</td>
<td>Nontransaction (qfixberr)</td>
<td>A type 3C record is written when a control block is changed during validation of a queue. Log Analyzer relates this type of record to a nontransaction (&quot;qfixberr&quot;) type of LUOW.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------</td>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>3D</td>
<td>QFIXQ</td>
<td>Queue block altered</td>
<td>Transaction or nontransaction (qfixqblk)</td>
<td>A type 3D record is written when a QBLK record was changed during validation of a queue. For RELB actions, Log Analyzer relates this type of record to a transaction LUOW. For other actions, Log Analyzer relates this type of record to a nontransaction (&quot;qfixqblk&quot;) type of LUOW.</td>
</tr>
<tr>
<td>3E</td>
<td>MSGCH</td>
<td>Message chain update</td>
<td>Transaction</td>
<td>A type 3E record is written when a message chain is updated. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>3F</td>
<td>UOWER</td>
<td>UOWE release</td>
<td>Nontransaction (uowe rel)</td>
<td>A type 3F record is written when a unit of work is released. Log Analyzer relates this type of record to a nontransaction (&quot;uowe rel&quot;) type of LUOW.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>-------------------</td>
<td>-----------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>40</td>
<td>CHKPT</td>
<td>Checkpoint</td>
<td>Nontransaction (chkpoint)</td>
<td>A type 40 record is written when an IMS system checkpoint is taken. Log Analyzer relates this type of record to a nontransaction (&quot;chkpoint&quot;) type of LUOW. Type 40 record subtypes include those that have the following meanings: ■ 4004—checkpoint SMB ■ 4009—checkpoint CPT ■ 400A—checkpoint CPM ■ 400B—checkpoint CTM ■ 400C—checkpoint CVB ■ 400F—checkpoint LCB ■ 4010—checkpoint CRB ■ 4011—checkpoint TCM ■ 4070—checkpoint MSDB begin ■ 4071—checkpoint MSDB ECNT ■ 4072—checkpoint MSDB header ■ 4073—checkpoint MSDB page fixed ■ 4074—checkpoint MSDB pageable ■ 4079—checkpoint MSDB end ■ 4080—checkpoint Fast Path begin ■ 4081—checkpoint Fast Path ECNT ■ 4082—checkpoint Fast Path EMHB ■ 4083—checkpoint Fast Path RCTE ■ 4084—checkpoint FP DMCB/DMAC ■ 4085—checkpoint Fast Path MTO buffer ■ 4086—checkpoint Fast Path DMHR/DEDB ■ 4087—checkpoint Fast Path ADSC ■ 4088—checkpoint Fast Path IEEQE ■ 4089—checkpoint Fast Path end</td>
</tr>
<tr>
<td>41</td>
<td>BATCH</td>
<td>Batch checkpoint</td>
<td>Transaction</td>
<td>A type 41 record is written when an IMS batch program or BMP program issues a checkpoint. Log Analyzer relates this type of record to a transaction LUOW if possible. If not, Log Analyzer displays the record as unmapped.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------</td>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 42              | LOGBF           | Log buffer control           | Nontransaction (logbuffr)     | A type 42 record is written when any of the following events occurs:  
  ■ IMS has switched from OLDS to another.  
  ■ A checkpoint was taken.  
  ■ A shutdown checkpoint was taken.  
  Log Analyzer relates this type of record to a nontransaction ("logbuffr") type of LUOW. |
| 43              | LOGDS           | Log data set control         | Nontransaction (log ds)       | A type 43 record is written when the log manager or the Log Archive utility reports the status of the current OLDS.  
  Log Analyzer relates this type of record to a nontransaction ("log ds") type of LUOW. |
| 45              | LSTAT           | IMS statistics               | Nontransaction (stats)        | A type 45 record is written to report checkpoint statistics. Log Analyzer relates this type of record to a nontransaction ("stats") type of LUOW. |
| 47              | RGN             | Summary of active regions    | Nontransaction (region)       | A type 47 record is written to report the partition specification tables (PSTs) that are in the system after a checkpoint is taken.  
  Log Analyzer relates this type of record to a nontransaction ("region") type of LUOW. |
<p>| 48              | PAD             | Padding record               | Nontransaction (padding)      | A type 48 record is written to pad a variable-length log record. Log Analyzer relates this type of record to a nontransaction (&quot;padding&quot;) type of LUOW. |
| 4C              | DBPSB           | Database/PSB start/stop/lock/unlock | Nontransaction (db/psb) | A type 4C record is written when an action that is related to a full-function database occurs. Actions can include backouts, write errors, program stops, database starts and stops, locks, and so on. Log Analyzer relates this type of record to a nontransaction (&quot;db/psb&quot;) type of LUOW. |</p>
<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>DBDSG</td>
<td>Database update</td>
<td>Transaction</td>
<td>A type 50 record is written when an application program updates a full-function database. DBD, UDBD, and DSID filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW if possible. If not, Log Analyzer displays the record as unmapped. Type 50 record subtypes include those that have the following meanings: ■ 5050—successful undo/redo update ■ 5051—unsuccessful update ■ 5052—undo KSDS insert</td>
</tr>
<tr>
<td>53</td>
<td>CI-CA</td>
<td>CI/CA and Space Manager</td>
<td>Transaction</td>
<td>A type 53 record is written when an alternate IMS system is tracing a CI split on an active IMS. DBD filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW if possible. If not, Log Analyzer displays the record as unmapped</td>
</tr>
<tr>
<td>55</td>
<td>ESAP</td>
<td>External subsystem</td>
<td>Transaction</td>
<td>A type 55 record is written to track external subsystem information. Log Analyzer relates this type of record to a transaction LUOW.</td>
</tr>
<tr>
<td>56</td>
<td>TPCP</td>
<td>External subsystem</td>
<td>Transaction</td>
<td>A type 56 record is written when any of the following events occurs: ■ An event has occurred to change the status of a transaction that is executing in an IMS external subsystem. ■ An event has occurred to change the status of a connection between IMS and the CCTL. ■ An event has occurred during IMS sync point processing. DBD filters apply to this type of record. Log Analyzer relates this type of record to a transaction LUOW if possible. If not, Log Analyzer displays the record as unmapped</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------</td>
<td>------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5607</td>
<td>TPCP</td>
<td>External subsystem</td>
<td>Transaction</td>
<td>A type 5607 record is written when a unit of recovery has started. For more information, see the type 56 record.</td>
</tr>
<tr>
<td>5612</td>
<td>TPCP</td>
<td>External subsystem</td>
<td>Transaction</td>
<td>A type 5612 record is written to indicate the end of a two-phase unit of recovery. For more information, see the type 56 record.</td>
</tr>
<tr>
<td>57</td>
<td>DBUR</td>
<td>Database update state record</td>
<td>Nontransaction (statechg)</td>
<td>A type 57 record is written to indicate whether an IMS system is updating a database data set in an RSR environment. DBD filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;statechg&quot;) type of LUOW.</td>
</tr>
<tr>
<td>59</td>
<td>FP</td>
<td>Fast Path database update</td>
<td>Transaction</td>
<td>A type 59 record is written when a Fast Path database is updated and when various events occur during Fast Path processing. DBD and area filters apply to this type of record. Log Analyzer relates a type 59 record that has subcode 36, 37, 38, or 50 to a transaction LUOW if possible. If not (or if the record has a different subcode), Log Analyzer displays the record as unmapped.</td>
</tr>
<tr>
<td>5E</td>
<td>SB</td>
<td>Sequential buffer handler requests</td>
<td>Nontransaction (sb-handl)</td>
<td>A type 5E record is written when a function of the sequential buffer handler is called. DBD and DSID/area filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;sb-handl&quot;) type of LUOW.</td>
</tr>
<tr>
<td>5F</td>
<td>DL/I</td>
<td>DL/I call trace</td>
<td>Transaction</td>
<td>A type 5F record is written for the specified PSB when the DL/I call trace facility is active. Log Analyzer relates this type of record to a transaction type of LUOW.</td>
</tr>
<tr>
<td>63</td>
<td>SESSN</td>
<td>Session initiate/terminate</td>
<td>Nontransaction (session)</td>
<td>A type 63 record is written when a log session is initiated or terminated. USERID and NODE filters apply to this type of record. Log Analyzer relates a type 63 record to a nontransaction (&quot;session&quot;) type of LUOW.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
<td>-------------</td>
<td>-----------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>64</td>
<td>MSCDS</td>
<td>MSC discard</td>
<td>Nontransaction (msc disc)</td>
<td>A type 64 record is written when an inconsistency is found in MSC processing. Log Analyzer relates this type of record to a nontransaction (&quot;msc disc&quot;) type of LUOW.</td>
</tr>
</tbody>
</table>
| 66             | 3600           | 3600 record | Nontransaction (3600)        | A type 66 record is written during preparation to enqueue or dequeue a message that is associated with any of the following elements:  
- 3614, FINANCE, or SLU P nodes  
- MSC links  
- ISC sessions  
NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction ("3600") type of LUOW. |
| 67             | TRACE          | Communications trace | Nontransaction (trace) | A type 67 record is written when various events occur and service tracing is in effect. Log Analyzer relates this type of record to a nontransaction ("trace") type of LUOW.  
Type 67FF records are included in LUOWs that are involved in deadlocks. Log Analyzer provides deadlock information in the details for this record type. |
<p>| 69             | 3275           | 3275 record | Nontransaction (3275)        | A type 69 record is written when an unauthorized 3275 terminal has dialed into a line that is specified as VERIFY=YES. Log Analyzer relates this type of record to a nontransaction (&quot;3275&quot;) type of LUOW. |
| 6C             | MSCDC          | MSC connect; MSC disconnect | Nontransaction (msc)       | A type 6C record is written when an MSC partner system is started or disconnected. Log Analyzer relates this type of record to a nontransaction (&quot;msc&quot;) type of LUOW. |</p>
<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>6D</td>
<td>FAST</td>
<td>Fast DB recovery</td>
<td>Nontransaction (fast db)</td>
<td>A type 6D record is written when any of the following events occur in an XRF environment:&lt;br&gt;■ A Fast database recovery task has received a timeout or is in a wait state.&lt;br&gt;■ XRF surveillance is started or stopped.&lt;br&gt;■ The active subsystem received a write error.&lt;br&gt;■ A /CHANGE command was used to change the interval or timeout values for the active subsystem. Log Analyzer relates this type of record to a nontransaction (&quot;fast db&quot;) type of LUOW.</td>
</tr>
<tr>
<td>6E</td>
<td>HARP</td>
<td>HARP session</td>
<td>Nontransaction (harp)</td>
<td>A type 6E record is written when an SNA command is processed. Log Analyzer relates this type of record to a nontransaction (&quot;harp&quot;) type of LUOW.</td>
</tr>
<tr>
<td>70</td>
<td>OLC</td>
<td>Online change</td>
<td>Nontransaction (olc)</td>
<td>A type 70 record is written at successful completion of a command sequence for the online change /MODIFY command, indicating an update to the IMS.MODSTAT data set. Log Analyzer relates this type of record to a nontransaction (&quot;olc&quot;) type of LUOW.</td>
</tr>
<tr>
<td>71</td>
<td>TCF</td>
<td>TCF</td>
<td>Nontransaction (tcf)</td>
<td>A type 71 record is written to record Timer Controlled Facility (TCF) events. Log Analyzer relates this type of record to a nontransaction (&quot;tcf&quot;) type of LUOW.</td>
</tr>
<tr>
<td>72</td>
<td>USER</td>
<td>User create, delete, modify</td>
<td>Nontransaction (usercre)</td>
<td>A type 72 record is written during the creation, deletion, or modification of ETO user structures for dynamic terminals. USERID and NODE filters apply to this type of record. Log Analyzer relates this type of record to a nontransaction (&quot;usercre&quot;) type of LUOW.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>-------------</td>
<td>-----------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>DA</td>
<td>DLP</td>
<td>DELTA PLUS</td>
<td>Nontransaction (delta+)</td>
<td>A type DA record is written by the BMC DELTA PLUS product. Log Analyzer relates this type of record to a nontransaction (&quot;delta+&quot;) type of LUOW.</td>
</tr>
<tr>
<td>DE</td>
<td>DLA (VTF)</td>
<td>DELTA IMS</td>
<td>Nontransaction (delta, vtf, vtfchk)</td>
<td>A type DE record is written by the BMC DELTA product for IMS or by the DELTA VIRTUAL TERMINAL product for IMS. USERID and NODE filters apply to this type of record if the record contains a user ID. Log Analyzer relates this type of record to a nontransaction type of LUOW.</td>
</tr>
<tr>
<td>EA</td>
<td>ETA</td>
<td>Extended Terminal Assist Plus</td>
<td>Nontransaction</td>
<td>A type EA record is written by the BMC EXTENDED TERMINAL ASSIST PLUS product. USERID and NODE filters apply to this type of record if the record contains a user ID. Log Analyzer relates this type of record to a nontransaction type of LUOW.</td>
</tr>
<tr>
<td>EF</td>
<td>MAQ</td>
<td>Message Advisor</td>
<td>Nontransaction (maq)</td>
<td>A type EF record is written by the BMC Message Advisor product for IMS. Log Analyzer relates this type of record to a nontransaction (&quot;maq&quot;) type of LUOW.</td>
</tr>
<tr>
<td>F9</td>
<td>MVF9</td>
<td>Mainview</td>
<td>Nontransaction (mainview)</td>
<td>A type F9 record is written by the BMC MainView product for IMS. Log Analyzer relates this type of record to a nontransaction (&quot;mainview&quot;) type of LUOW.</td>
</tr>
<tr>
<td>FA</td>
<td>MVFA</td>
<td>Mainview</td>
<td>Transaction or nontransaction (mainview)</td>
<td>A type FA record is written by the BMC MainView product for IMS. DBD, and RDBD filters apply to this type of record. If a type FA record was created in an IMS TM environment, Log Analyzer relates it to a transaction type of LUOW. If a type FA record was created in a CICS-DB/CTL environment and your level of MainView provides the required support, Log Analyzer relates it to a transaction type of LUOW. If your level of MainView does not provide the required support, Log Analyzer relates this type of record to a nontransaction (&quot;mainview&quot;) type of LUOW.</td>
</tr>
</tbody>
</table>
## Non-IMS log record types

The following table lists the types of non-IMS log records that Log Analyzer has mapped, the type of LUOW to which the record is associated, and the label that Log Analyzer uses for the type of LUOW in reports.

### Table 34: Non-IMS log record types that Log Analyzer has mapped

<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0060</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0060 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect has sent a <strong>prepare read</strong> message to IMS. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0061</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0061 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect has sent a <strong>u-xit entry</strong> message to IMS. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0062</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0062 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect has sent a <strong>u-xit exit</strong> message to IMS. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0065</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0065 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect has sent a message to IMS. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------</td>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>0066</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0066 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS sent a message to IMS Connect. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0069</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0069 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect has sent a timeout message to IMS. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0071</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0065 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect has sent a event abend message to IMS. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0072</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0065 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect has sent a event end message to IMS. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
<td>-----------------------------------</td>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>0073</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0073 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that an IMS Connect read-socket event has occurred (meaning that data came from a client in the TCP/IP network). If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0074</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 0074 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that IMS Connect sent a message to the client in the TCP/IP network. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0091</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 091 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that an IMS Connect DRDA command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0092</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 092 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that an IMS Connect DRDA reply command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
</tbody>
</table>
### Log record types

<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0093</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 93 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that an IMS Connect <strong>PSB strt alloc</strong> command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0094</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 94 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that an IMS Connect <strong>PSB end alloc</strong> command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0097</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 97 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that an IMS Connect <strong>enter rt exit</strong> command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0098</td>
<td>IMSConnect</td>
<td>Energizer for IMS Connect</td>
<td>Transaction</td>
<td>type 98 (IMSCONNECT) record is created by the BMC Energizer for IMS Connect product to indicate that an IMS Connect <strong>return from rt exit</strong> command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td>LUOW type (label on reports)</td>
<td>Comments</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------</td>
<td>-------------</td>
<td>-----------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>0105</td>
<td>IMSConnect</td>
<td>Energizer <em>for IMS Connect</em></td>
<td>Transaction</td>
<td>type 105 (IMSCONNECT) record is created by the BMC Energizer <em>for IMS Connect</em> product to indicate that an IMS Connect msg to ODBA command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0106</td>
<td>IMSConnect</td>
<td>Energizer <em>for IMS Connect</em></td>
<td>Transaction</td>
<td>type 106 (IMSCONNECT) record is created by the BMC Energizer <em>for IMS Connect</em> product to indicate that an IMS Connect msg from ODBA command event has occurred. If Energizer journal records are available to the analysis job, Log Analyzer incorporates this record type in transaction LUOWs.</td>
</tr>
<tr>
<td>0002</td>
<td>DB2 record</td>
<td>DB2 undo</td>
<td>Transaction</td>
<td>type 0002 record indicates a DB2 unit-of-recovery undo event. If DB2 log records are available, Log Analyzer incorporates selected records in LUOWs to provide information about DB2 data and timings within IMS transactions.</td>
</tr>
<tr>
<td>0020</td>
<td>DB2 record</td>
<td>DB2 control</td>
<td>Transaction</td>
<td>type 0020 record indicates a DB2 unit-of-recovery control event. If DB2 log records are available, Log Analyzer incorporates selected records in LUOWs to provide information about DB2 data and timings within IMS transactions.</td>
</tr>
<tr>
<td>0400</td>
<td>DB2 record</td>
<td>DB2 redo</td>
<td>Transaction</td>
<td>type 0400 record indicates a DB2 unit-of-recovery redo event. If DB2 log records are available, Log Analyzer incorporates selected records in LUOWs to provide information about DB2 data and timings within IMS transactions.</td>
</tr>
</tbody>
</table>
Log records types in transaction-type LUOWs

The following table lists the types of records that Log Analyzer associates with transaction-type LUOWs.

Table 35: Log record types in transaction-type LUOWs

<table>
<thead>
<tr>
<th>Log record code</th>
<th>Log record type</th>
<th>Description</th>
<th>LUOW type (label on reports)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0600</td>
<td>DB2 record</td>
<td>DB2 undo/redo</td>
<td>Transaction</td>
<td>type 0600 record indicates a DB2 unit-of-recovery combined undo/redo event. If DB2 log records are available, Log Analyzer incorporates selected records in LUOWs to provide information about DB2 data and timings within IMS transactions.</td>
</tr>
<tr>
<td>01</td>
<td>MSGIN</td>
<td>IMS input message</td>
<td></td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>MSGOT</td>
<td>IMS output message</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>APPLT</td>
<td>Application terminate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>APPLC</td>
<td>Application start</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0A</td>
<td>CPICI</td>
<td>CPI-CI driven program start/terminate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>XCHK</td>
<td>Extended checkpoint</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>PH1DC</td>
<td>Phase 1 DC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>APRFX</td>
<td>Additional message prefix</td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>MSGGU (GUDLI, GUCOM)</td>
<td>Message GET-UNIQUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>REJCT</td>
<td>Message reject</td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>FREE</td>
<td>Message free</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>CANCL</td>
<td>Message cancel</td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>MGENQ (ENQ, REENQ)</td>
<td>Message enqueue</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>MGDEQ (MGDEL, MGSAV)</td>
<td>Message dequeue</td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>XFER</td>
<td>Message queue sync</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>QRET</td>
<td>Queue return</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log record code</td>
<td>Log record type</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>QREL</td>
<td>Queue release</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D&lt;sup&gt;a&lt;/sup&gt;</td>
<td>QFIXQ</td>
<td>Queue block altered</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3E</td>
<td>MSGCH</td>
<td>Message chain update</td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>BATCH</td>
<td>Batch checkpoint</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>DBDSG</td>
<td>Database update</td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>CI-CA</td>
<td>CI/CA and Space Manager</td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>ESAP</td>
<td>External subsystem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>TPCP</td>
<td>External subsystem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>FP</td>
<td>Fast Path database update</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5F</td>
<td>DL/I</td>
<td>DL/I call trace</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FA</td>
<td>MainView</td>
<td>MainView</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<sup>a</sup> In some cases, a type 3D record is associated with a non-transaction LUOW.

## Nontransaction and other LUOW types

The following table lists the types of nontransaction and other LUOWs that Log Analyzer can create and the types of log records that are associated with them.

Reports and panels identify these types of LUOWs with the indicated label. For more information, see the description of indicated type of log record in “IMS log record types” on page 345.

### Table 36: Nontransaction and other LUOW types

<table>
<thead>
<tr>
<th>Log record code</th>
<th>LUOW label</th>
<th>Log record description</th>
</tr>
</thead>
<tbody>
<tr>
<td>02</td>
<td>command</td>
<td>Condensed command - Type I</td>
</tr>
<tr>
<td>04</td>
<td>rsr</td>
<td>RSR tracking</td>
</tr>
<tr>
<td>06</td>
<td>action</td>
<td>IMS internally initiated action</td>
</tr>
<tr>
<td>09</td>
<td>sb stats</td>
<td>Sequential buffering statistics</td>
</tr>
<tr>
<td>10</td>
<td>sec viol</td>
<td>Security violation</td>
</tr>
<tr>
<td>11, 12</td>
<td>conv</td>
<td>Start conversation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>End conversation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Conversation control block</td>
</tr>
</tbody>
</table>

Appendix C  Log record and LUOW reference 367
<table>
<thead>
<tr>
<th>Log record code</th>
<th>LUOW label</th>
<th>Log record description</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>dialline</td>
<td>Switch line disconnect</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>Switch line connect</td>
</tr>
<tr>
<td>16</td>
<td>signonof</td>
<td>Sign on; Sign off</td>
</tr>
<tr>
<td>18</td>
<td>ext chk</td>
<td>Extended checkpoint</td>
</tr>
<tr>
<td>20</td>
<td>open/cls</td>
<td>Database open</td>
</tr>
<tr>
<td>21</td>
<td></td>
<td>Database close</td>
</tr>
<tr>
<td>22</td>
<td>command2</td>
<td>IMSPLEX command</td>
</tr>
<tr>
<td>24</td>
<td>db error</td>
<td>Database error</td>
</tr>
<tr>
<td>25</td>
<td>eeqe</td>
<td>EEQE created; EEQE deleted</td>
</tr>
<tr>
<td>26</td>
<td>iot</td>
<td>I/O toleration buffer</td>
</tr>
<tr>
<td>27</td>
<td>db exten</td>
<td>Database extended</td>
</tr>
<tr>
<td>3A</td>
<td>qfixfree</td>
<td>Queue record free</td>
</tr>
<tr>
<td>3B</td>
<td>qfixrerr</td>
<td>Queue invalid message record</td>
</tr>
<tr>
<td>3C</td>
<td>qfixberr</td>
<td>Queue destination change</td>
</tr>
<tr>
<td>3D</td>
<td>qfixqblk</td>
<td>Queue block altered</td>
</tr>
<tr>
<td>3F</td>
<td>uowe rel</td>
<td>UOWE release</td>
</tr>
<tr>
<td>40</td>
<td>chkpoint</td>
<td>Checkpoint</td>
</tr>
<tr>
<td>42</td>
<td>logbuffr</td>
<td>Log buffer control</td>
</tr>
<tr>
<td>43</td>
<td>log ds</td>
<td>Log data set control</td>
</tr>
<tr>
<td>45</td>
<td>stats</td>
<td>IMS statistics</td>
</tr>
<tr>
<td>47</td>
<td>region</td>
<td>Summary of active regions</td>
</tr>
<tr>
<td>48</td>
<td>padding</td>
<td>Padding record</td>
</tr>
<tr>
<td>4C</td>
<td>db/psb</td>
<td>Database/PSB start/stop/lock/unlock</td>
</tr>
<tr>
<td>57</td>
<td>statechg</td>
<td>Database state change record</td>
</tr>
<tr>
<td>5E</td>
<td>sb-handl</td>
<td>Sequential buffer handler requests</td>
</tr>
<tr>
<td>63</td>
<td>session</td>
<td>Session initiate/terminate</td>
</tr>
<tr>
<td>64</td>
<td>msc disc</td>
<td>MSC discard</td>
</tr>
<tr>
<td>66</td>
<td>3600</td>
<td>3600 record</td>
</tr>
<tr>
<td>67</td>
<td>trace</td>
<td>Communication trace records</td>
</tr>
<tr>
<td>69</td>
<td>3275</td>
<td>3275 record</td>
</tr>
<tr>
<td>6C</td>
<td>msc</td>
<td>MSC connect; MSC disconnect</td>
</tr>
</tbody>
</table>
### Log record types that are included automatically

In an analysis job, Log Analyzer can include one or more log record codes automatically if any of the following conditions is true:

<table>
<thead>
<tr>
<th>Log record code</th>
<th>LUOW label</th>
<th>Log record description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6D</td>
<td>fast db</td>
<td>Fast DB recovery</td>
</tr>
<tr>
<td>6E</td>
<td>harp</td>
<td>HARP session</td>
</tr>
<tr>
<td>70</td>
<td>olc</td>
<td>Online change</td>
</tr>
<tr>
<td>72</td>
<td>usercre</td>
<td>User create, delete, modify</td>
</tr>
<tr>
<td>0F</td>
<td>lelogrec</td>
<td>Logical logger</td>
</tr>
<tr>
<td>DA</td>
<td>delta+</td>
<td>various DELTA PLUS records</td>
</tr>
<tr>
<td>DE</td>
<td>delta</td>
<td>various DELTA IMS records</td>
</tr>
<tr>
<td>DE</td>
<td>vtf</td>
<td>various DELTA VIRTUAL TERMINAL records</td>
</tr>
<tr>
<td>DE</td>
<td>vtfchk</td>
<td>DELTA VIRTUAL TERMINAL checkpoint log record</td>
</tr>
<tr>
<td>EA</td>
<td>eta</td>
<td>various EXTENDED TERMINAL ASSIST PLUS records</td>
</tr>
<tr>
<td>EA</td>
<td>eta chk</td>
<td>EXTENDED TERMINAL ASSIST PLUS checkpoint record</td>
</tr>
<tr>
<td>EF</td>
<td>maq</td>
<td>Message Advisor</td>
</tr>
<tr>
<td>F9</td>
<td>mainview</td>
<td>Mainview</td>
</tr>
<tr>
<td>varies</td>
<td>misc</td>
<td>LUOW that does not contain a type 01 log record or a type 03 log record</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For information about &quot;misc&quot; types of LUOWs, see “Nontransaction LUOWs” on page 303.</td>
</tr>
<tr>
<td>varies</td>
<td>unmapd**</td>
<td>Unmapped</td>
</tr>
<tr>
<td></td>
<td></td>
<td>An unmapped log record is one for which Log Analyzer has no type code, subcode, or short text. An unmapped record is usually a &quot;user&quot; type of log record.</td>
</tr>
</tbody>
</table>
Log Analyzer is using the default log record code list for one of the following reasons:

— You have omitted the LOGRECORDCODES keyword and have specified the SELECT keyword on the FILTER control statement.

— You have specified the LOGRECORDCODES=ALL keyword and the SELECT keyword on the FILTER control statement and have requested reports other than the Analysis Summary report.

Log Analyzer is using the default log record code list, and you have specified a SELECT subkeyword that operates on a log record code which is not included in this default list.

Log Analyzer is using the explicit log record code list that is specified as the value of the LOGRECORDCODES keyword on the FILTER control statement, and you have specified a SELECT subkeyword that operates on a log record code which has been omitted from your specified list.

The following table lists each log record code that Log Analyzer can include automatically.

### Table 37: Automatically included log record types

<table>
<thead>
<tr>
<th>Log record code</th>
<th>Transaction LUOW</th>
<th>Conditions when this log record type is included automatically</th>
</tr>
</thead>
</table>
| 01              | Yes              | Log Analyzer is using the default log record code list, or one or more of the following SELECT subkeywords are specified on the FILTER control statement: CICSTRAN=(\textit{transaction}) CONTENT=(C' string',01...)
<p>|                 |                  | DEST=(\textit{destination}) or 01DEST=(\textit{destination}) |
|                 |                  | LUNAME=(\textit{luname}) or 01LUNAME=(\textit{luname}) |
|                 |                  | LTERM=(\textit{lterm}) or 01LTERM=(\textit{lterm}) |
|                 |                  | NODE=(\textit{node}) or 01NODE=(\textit{node}) |
|                 |                  | ORIGIN=(\textit{origin}) or 01ORIGIN=(\textit{origin}) |
|                 |                  | PORT#=(\textit{port}) or 01PORT#=(\textit{port}) |
|                 |                  | TMEMBER=(\textit{tmembername}) or 01TMEMBER=(\textit{tmembername}) |
|                 |                  | TPIPE=(\textit{tpipename}) or 01TPIPE=(\textit{tpipename}) |
|                 |                  | TPNAME=(\textit{tpname}) or 01TPNAME=(\textit{tpname}) |
|                 |                  | TRAN=(\textit{tranname}) or 01TRAN=(\textit{tranname}) |
|                 |                  | UOW1='8characterImISID16characterUowID' |
|                 |                  | USERID=(\textit{userid}) or 01USERID=(\textit{userid}) |</p>
<table>
<thead>
<tr>
<th>Log record code</th>
<th>Transaction LUOW</th>
<th>Conditions when this log record type is included automatically</th>
</tr>
</thead>
</table>
| 02              | No               | One or more of the following keywords are specified on the FILTER control statement:  
|                 |                  | SELECT=NODE=(node)  
|                 |                  | SELECT=USERID=(userid) keyword |
| 03              | Yes              | Log Analyzer is using the default log record code list, or one or more of the following SELECT subkeywords are specified on the FILTER control statement:  
|                 |                  | CICSTRAN=(transaction)  
|                 |                  | CONTENT=(C' string',03...)  
|                 |                  | DEST=(destination) or 03DEST=(destination)  
|                 |                  | LUNAME=(luname) or 03LUNAME=(luname)  
|                 |                  | LTERM=(lterm) or 03LTERM=(lterm)  
|                 |                  | NODE=(node) or 03NODE=(node)  
|                 |                  | ORIGIN=(origin) or 03ORIGIN=(origin)  
|                 |                  | PORT#=(port) or 03PORT#=(port)  
|                 |                  | TMEMBER=(tmembername) or 03TMEMBER=(tmembername)  
|                 |                  | TPIPE=(tpipename) or 03TPIPE=(tpipename)  
|                 |                  | TPNAME=(tpname) or 03TPNAME=(tpname)  
|                 |                  | TRAN=(tranname) or 01TRAN=(tranname)  
|                 |                  | USERID=(userid) or 03USERID=(userid)  
|                 |                  | UOW1= '8characterImsID16characterUowID' |
| 07              | Yes              | One of the following conditions exists:  
|                 |                  | - Log Analyzer is using the default log record code list.  
|                 |                  | - The SELECT=ABENDS=YES keyword is specified on the FILTER control statement.  
|                 |                  | - The ABEND keyword is specified on the REPORTS control statement. |
| 08              | Yes              | Log Analyzer is using the default log record code list, or the SELECT=PSB=(psbname) keyword is specified on the FILTER control statement.  
|                 |                  | SELECT=CICSTRAN=(transaction) |
| 10              | No               | One or more of the following keywords are specified on the FILTER control statement:  
|                 |                  | SELECT=NODE=(node)  
<p>|                 |                  | SELECT=USERID=(userid) keyword |</p>
<table>
<thead>
<tr>
<th>Log record code</th>
<th>Transaction LUOW</th>
<th>Conditions when this log record type is included automatically</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement: SELECT=NODE=(node) SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>12</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement: SELECT=NODE=(node) SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>13</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement: SELECT=NODE=(node) SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>16</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement: SELECT=NODE=(node) SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>18</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>20</td>
<td>No</td>
<td>The SELECT=DBD=(dbname...) keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>21</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement: SELECT=DBD=(dbname...) SELECT=USERID=(userid)</td>
</tr>
<tr>
<td>24</td>
<td>No</td>
<td>The SELECT=DBD=(dbname...) keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>25</td>
<td>No</td>
<td>The SELECT=DBD=(dbname...) keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>26</td>
<td>No</td>
<td>The SELECT=DBD=(dbname...) keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>27</td>
<td>No</td>
<td>The SELECT=DBD=(dbname...) keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>28</td>
<td>No</td>
<td>The SELECT=NODE=(node) keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>30</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>Log record code</td>
<td>Transaction LUOW</td>
<td>Conditions when this log record type is included automatically</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>31</td>
<td>Yes</td>
<td>One of the following conditions exists:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ The SELECT=UOW1= ‘8characterImsID16characterUowID’ keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Log Analyzer can also include this record in other cases as needed.</td>
</tr>
<tr>
<td>32</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>33</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>34</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>35</td>
<td>Yes</td>
<td>One of the following conditions exists:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ The SELECT=UOW1= ‘8characterImsID16characterUowID’ keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Log Analyzer can also include this record in other cases as needed.</td>
</tr>
<tr>
<td>36</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>37</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>38</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>39</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>41</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>50</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list, or one or more of the following keywords are specified on the FILTER control statement:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=DBD=(dbdname...)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=UDBD=(dbdname...)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=PSB=(psbname...)</td>
</tr>
<tr>
<td>53</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>56</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list, or one or more of the following keywords are specified on the FILTER control statement:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=DBD=(dbdname...)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=PSB=(psbname...)</td>
</tr>
<tr>
<td>Log record code</td>
<td>Transaction LUOW</td>
<td>Conditions when this log record type is included automatically</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>59</td>
<td>Yes</td>
<td>One of the following conditions exists:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The SELECT=DBD=(dbdname...) subkeyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- For certain subcodes, the SELECT=UOW1='8characterImSID16characterUowID' subkeyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>5F</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list.</td>
</tr>
<tr>
<td>63</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=NODE=(node)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>66</td>
<td>No</td>
<td>The SELECT=NODE=(node) keyword is specified on the FILTER control statement.</td>
</tr>
<tr>
<td>72</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=NODE=(node)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>DE (if the</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement:</td>
</tr>
<tr>
<td>record contains</td>
<td></td>
<td>SELECT=NODE=(node)</td>
</tr>
<tr>
<td>a user ID)</td>
<td></td>
<td>SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>EA (if the</td>
<td>No</td>
<td>One or more of the following keywords are specified on the FILTER control statement:</td>
</tr>
<tr>
<td>record contains</td>
<td></td>
<td>SELECT=NODE=(node)</td>
</tr>
<tr>
<td>a user ID)</td>
<td></td>
<td>SELECT=USERID=(userid) keyword</td>
</tr>
<tr>
<td>FA</td>
<td>Yes</td>
<td>Log Analyzer is using the default log record code list, or one or more of the following keywords are specified on the FILTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>control statement: SELECT=DBD=(dbdname,...)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SELECT=RDBD=(dbdname,...)</td>
</tr>
</tbody>
</table>

**Variable data types**

Log Analyzer provides details about log records (in batch reports and the ISPF interface) and, whenever possible, reports relevant data from within the log record.
Log Analyzer "snips" relevant data by mapping relevant fields in the log record and saving the data from the fields as the record is read. Relevant data depends on the type of log record.

The following table lists and describes some of the variable data fields that Log Analyzer can snip from log records. The short name can be used on batch reports and in the ISPF interface where space is limited. The description is used in the ISPF interface where space is available.

Table 38: Variable data types snipped from log records

<table>
<thead>
<tr>
<th>Short name</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>oims</td>
<td>Originating IMSID</td>
<td>IMSID from which the message was originally input</td>
</tr>
<tr>
<td>origims</td>
<td>Originating IMSID</td>
<td>IMSID from which the message was originally input</td>
</tr>
<tr>
<td>dims</td>
<td>Destination IMSID</td>
<td>IMSID in which the message is destined to execute</td>
</tr>
<tr>
<td>destims</td>
<td>Destination IMSID</td>
<td>IMSID in which the message is destined to execute</td>
</tr>
<tr>
<td>origin</td>
<td>Origin</td>
<td>Name of a logical terminal (LTERM) or a transaction that indicates where the message (or messages) originated</td>
</tr>
<tr>
<td>dest</td>
<td>Destination</td>
<td>Name of a logical terminal (LTERM), a transaction, an OTMA TMEMBER, or an APPC transaction program (TPNAME)</td>
</tr>
<tr>
<td></td>
<td>Destination type</td>
<td>Queue of messages that are destined for a terminal or program. The destination type can be CCB, CNT, or SMB</td>
</tr>
<tr>
<td>ims TOD</td>
<td>IMS timestamp</td>
<td>Date and timestamp provided by IMS when this log record was created. The format is: yyyy.ddd.hh.mm.ss.t</td>
</tr>
<tr>
<td>cmd</td>
<td>Command</td>
<td>IMS command that was issued</td>
</tr>
<tr>
<td></td>
<td>First segment?</td>
<td>For a multi-segment message, shows YES or NO</td>
</tr>
<tr>
<td></td>
<td>Last segment?</td>
<td>For a multi-segment message, shows YES or NO</td>
</tr>
<tr>
<td>readDBs</td>
<td>List of databases that were accessed by read-only PSB</td>
<td></td>
</tr>
<tr>
<td>readDBs/upd</td>
<td>List of databases with read and update</td>
<td></td>
</tr>
<tr>
<td>sysid</td>
<td>Sysid</td>
<td>RSR tracking system (RSENAME) that created this record</td>
</tr>
<tr>
<td>stk timestamp</td>
<td>STCK timestamp</td>
<td>Actual timestamp from the STCK instruction</td>
</tr>
<tr>
<td></td>
<td>Reason for record</td>
<td>Reason that this x'06' accounting record was created</td>
</tr>
<tr>
<td>IMSID</td>
<td>IMS system ID</td>
<td></td>
</tr>
<tr>
<td>RSE name</td>
<td>Recovery Service Element (RSE) name that is associated with this log record</td>
<td></td>
</tr>
<tr>
<td>recovery token</td>
<td>Recovery token</td>
<td>Full name that is assigned for this recovery token</td>
</tr>
<tr>
<td>tran</td>
<td>Transaction code</td>
<td>Name of this transaction</td>
</tr>
<tr>
<td>Short name</td>
<td>Description</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>psb</td>
<td>PSB name</td>
<td>Name of this program specification block (PSB)</td>
</tr>
<tr>
<td>program</td>
<td>Program name</td>
<td>Name of the program</td>
</tr>
<tr>
<td>job or jobname</td>
<td>Jobname</td>
<td>Job name associated with this log record</td>
</tr>
<tr>
<td></td>
<td>Stepname</td>
<td>Step name associated with this log record</td>
</tr>
<tr>
<td></td>
<td>Completion code</td>
<td>Return code associated with the completion of this application</td>
</tr>
<tr>
<td>pst</td>
<td>PST number</td>
<td>IMS assigned number associated with this partition specification table (PST)</td>
</tr>
<tr>
<td></td>
<td>Program type</td>
<td>Type of program executed, typically BMP or MPP</td>
</tr>
<tr>
<td></td>
<td>Abend flag</td>
<td>Flag set by IMS during application termination</td>
</tr>
<tr>
<td></td>
<td>Number of messages</td>
<td>Actual number of IMS messages processing during this application scheduling</td>
</tr>
<tr>
<td></td>
<td>Region type</td>
<td>Used by CPI-C communication x'0Axx' log records</td>
</tr>
<tr>
<td></td>
<td>Conversation id</td>
<td>IMS assigned numerical identification assigned to this conversation (CPIC related)</td>
</tr>
<tr>
<td></td>
<td>Conversation owner</td>
<td>Actual owner of this conversation (CPIC related)</td>
</tr>
<tr>
<td></td>
<td>Execution time</td>
<td>Total time to complete this scheduling (CPIC related)</td>
</tr>
<tr>
<td></td>
<td>Tran arrival time</td>
<td>IMS assigned timestamp when this transaction arrived (CPIC related)</td>
</tr>
<tr>
<td></td>
<td>Error type</td>
<td>Type of error encountered</td>
</tr>
<tr>
<td></td>
<td>Error code</td>
<td>Alphanumeric value assigned to this type of error</td>
</tr>
<tr>
<td></td>
<td>Error block number</td>
<td>Alphanumeric value assigned to this error</td>
</tr>
<tr>
<td></td>
<td>BTAM line</td>
<td>BTAM rel. line</td>
</tr>
<tr>
<td></td>
<td>BTAM terminal</td>
<td>BTAM rel. terminal</td>
</tr>
<tr>
<td>cmpnodnm or VTAM node or node</td>
<td>Node name</td>
<td>Name of the inputting IMS terminal</td>
</tr>
<tr>
<td>cnt</td>
<td>CNT</td>
<td>Communication Name Table (CNT), refers to the LTERM (logical terminal name) name associated with this message</td>
</tr>
<tr>
<td>CMPSPQB</td>
<td>SPQB</td>
<td>Subpool Queue Block (SPQB) that refers to the user name associated with this message</td>
</tr>
<tr>
<td>spqb</td>
<td>SMB</td>
<td>Scheduler Message Block (SMB) or transaction name for this message</td>
</tr>
<tr>
<td>lterm</td>
<td>LTERM name</td>
<td>Communication Name Table (CNT), refers to the LTERM (logical terminal name) name associated with this message</td>
</tr>
<tr>
<td>term</td>
<td>Terminal id</td>
<td>Name of the inputting IMS terminal</td>
</tr>
<tr>
<td>Short name</td>
<td>Description</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>line#</td>
<td>Line number</td>
<td>IMS system generation (SYSGEN) line associated with this message</td>
</tr>
<tr>
<td></td>
<td></td>
<td>User name</td>
</tr>
<tr>
<td>userid</td>
<td>Userid</td>
<td>IBM RACF® (or other security package) ID associated with this message</td>
</tr>
<tr>
<td></td>
<td></td>
<td>User checkpoint id</td>
</tr>
<tr>
<td>ddbname</td>
<td>DD name</td>
<td>For dynamic allocation purposes, this is the name associated with the data set for this database</td>
</tr>
<tr>
<td>dmb</td>
<td>DMB name</td>
<td>Data Management Block (DMB) name that is associated with this database</td>
</tr>
<tr>
<td>dmb#</td>
<td>DMB number</td>
<td>Database definition (DBD) generation output number associated with this DMB (Data Management Block)</td>
</tr>
<tr>
<td>dc#</td>
<td>DCB number</td>
<td>Data Control Block (DCB) number associated with this DMB (Data Management Block) output from the DBD (database definition) generation process</td>
</tr>
<tr>
<td>verb</td>
<td>Command verb</td>
<td>Command verb from a Type II command</td>
</tr>
<tr>
<td>keyword</td>
<td>Command keyword</td>
<td>Command keyword from a Type II command</td>
</tr>
<tr>
<td>dbd or name</td>
<td>Database name</td>
<td>Database name associated with this message</td>
</tr>
<tr>
<td>ddbname</td>
<td>Database DD name</td>
<td>For dynamic allocation purposes, this is the name associated with the data set for this database</td>
</tr>
<tr>
<td>dbd/psb name</td>
<td>Database/PSB name</td>
<td>Database name or the PSB name for this log record</td>
</tr>
<tr>
<td></td>
<td>Database org</td>
<td>Chosen database organization (HIDAM, HDAM, etc.)</td>
</tr>
<tr>
<td></td>
<td>Data set org</td>
<td>Chosen data set organization (VSAM, OSAM, and so on)</td>
</tr>
<tr>
<td></td>
<td>Symbolic name</td>
<td>Symbolic name</td>
</tr>
<tr>
<td>ssid</td>
<td>Subsystem ID</td>
<td>Subsystem id where DB extend happened</td>
</tr>
<tr>
<td></td>
<td>LU 6.2 subpool name</td>
<td>For the LU 6.2 protocol, this is the LTERM or User name for this communication</td>
</tr>
<tr>
<td>drrn</td>
<td>DRRN</td>
<td>Disk relative record number (DRRN) of this message (the actual address of the message within the IMS message queue)</td>
</tr>
<tr>
<td></td>
<td>First DRRN</td>
<td>First disk relative record number (DRRN) for this message (the actual address of the message within the IMS message queue)</td>
</tr>
<tr>
<td></td>
<td>Last DRRN</td>
<td>Final disk relative record number (DRRN) for this message (the actual address of the message within the IMS message queue)</td>
</tr>
<tr>
<td>Short name</td>
<td>Description</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>DRRN count</td>
<td>Actual number of disk relative record numbers (DRRN) for this message</td>
<td></td>
</tr>
<tr>
<td>DRRN of msg to move</td>
<td>Disk relative record numbers (DRRN) to be moved for this message</td>
<td></td>
</tr>
<tr>
<td>DRRN of next msg</td>
<td>Disk relative record numbers (DRRN) of the next segment of this IMS message</td>
<td></td>
</tr>
<tr>
<td>freed DRRN or released DRRN</td>
<td>Disk relative record numbers (DRRN) of the IMS message to be released (deleted)</td>
<td></td>
</tr>
<tr>
<td>changed DRRN or updated DRRN</td>
<td>Disk relative record numbers (DRRN) of the queue record to be changed or updated</td>
<td></td>
</tr>
<tr>
<td>DRRN of error record</td>
<td>Disk relative record numbers (DRRN) of the error record</td>
<td></td>
</tr>
<tr>
<td>DRRN of QBLK</td>
<td>Disk relative record numbers (DRRN) of the QBLK entry for the IMS message</td>
<td></td>
</tr>
<tr>
<td>DRRN of next QBLK</td>
<td>Disk relative record numbers (DRRN) of the next QBLK entry for the IMS message</td>
<td></td>
</tr>
<tr>
<td>Bit map</td>
<td>IMS bit map representation of this data</td>
<td></td>
</tr>
<tr>
<td>PCB address</td>
<td>Actual address of the program communication block (PCB) in the IMS buffer pool</td>
<td></td>
</tr>
<tr>
<td>Queue number</td>
<td>IMS-assigned number that represents the internal IMS queue on which this message resides</td>
<td></td>
</tr>
<tr>
<td>Reason codes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Caller type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uow count</td>
<td>UOW count</td>
<td>Total number of Units of Work (UOW)</td>
</tr>
<tr>
<td>uow</td>
<td>UOW id</td>
<td>Identification number assigned to this Unit of Work (UOW)</td>
</tr>
<tr>
<td>checkpoint or checkpoint #</td>
<td>Checkpoint number</td>
<td>For a checkpoint restartable UOW, this is the current checkpoint ID</td>
</tr>
<tr>
<td>IMS release</td>
<td>IMS release</td>
<td>Version, release, and modification (V.R.M) of this IMS system</td>
</tr>
<tr>
<td>chkpt id</td>
<td>Checkpoint id</td>
<td>Checkpoint ID in checkpoint record</td>
</tr>
<tr>
<td>Checkpoint type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Restart type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enhanced restart type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>System type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buffer flag</td>
<td></td>
<td></td>
</tr>
<tr>
<td>batch system flag</td>
<td>Batch system flag</td>
<td>Batch system flag in batch checkpoint</td>
</tr>
<tr>
<td>Short name</td>
<td>Description</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>IRLM subsystem name</td>
<td>Name of the IRLM subsystem</td>
<td></td>
</tr>
<tr>
<td>IRLM subsystem id</td>
<td>Identification of the IRLM subsystem</td>
<td></td>
</tr>
<tr>
<td>ims</td>
<td>IMS subsystem name</td>
<td>IMS subsystem in batch checkpoint</td>
</tr>
<tr>
<td>ims</td>
<td>IMS subsystem id</td>
<td>IMS ID</td>
</tr>
<tr>
<td>Sharing status flag</td>
<td></td>
<td></td>
</tr>
<tr>
<td>record id</td>
<td>Record identifier</td>
<td>Stats log record ID</td>
</tr>
<tr>
<td>dsid</td>
<td>Data set id</td>
<td>Combination of DBD name and data set number</td>
</tr>
<tr>
<td></td>
<td>Data set id 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Update set id</td>
<td></td>
</tr>
<tr>
<td>area</td>
<td>Area name</td>
<td>IMS Fast Path area name</td>
</tr>
<tr>
<td></td>
<td>Area id</td>
<td>Identification number for this Fast Path area</td>
</tr>
<tr>
<td>half sess</td>
<td>Half-sess qualifier</td>
<td>Half session qualifying name</td>
</tr>
<tr>
<td></td>
<td>Other half-sess qualifier</td>
<td>Other half session qualifying name</td>
</tr>
<tr>
<td>modname</td>
<td>Modename used for sess</td>
<td>Mode name used for session</td>
</tr>
<tr>
<td>mod</td>
<td>Module name</td>
<td>Module identifier</td>
</tr>
<tr>
<td>clb</td>
<td>CLB name</td>
<td>Another name for the inputting IMS terminal (communication line block)</td>
</tr>
<tr>
<td></td>
<td>Record type</td>
<td></td>
</tr>
<tr>
<td>link#</td>
<td>Link number</td>
<td>Multiple System Communication (MSC) link number associated with this message</td>
</tr>
<tr>
<td>trace ids</td>
<td>Trace of MS ids</td>
<td>list of IMS IDs in EF log record</td>
</tr>
<tr>
<td></td>
<td>Graphic id</td>
<td></td>
</tr>
<tr>
<td>1st send</td>
<td>Time of first send</td>
<td>Link connection first send time</td>
</tr>
<tr>
<td>2nd send</td>
<td>Time of first receive</td>
<td>Link connection first receive time</td>
</tr>
<tr>
<td>type</td>
<td>Type</td>
<td>Online change record type</td>
</tr>
<tr>
<td>modifyid</td>
<td>Modify id</td>
<td>Online change modify id</td>
</tr>
<tr>
<td>modstat</td>
<td>DDname 1</td>
<td>Online change modstat1 DD</td>
</tr>
<tr>
<td>modstat2</td>
<td>DDname 2</td>
<td>Online change modstat2 DD</td>
</tr>
<tr>
<td>tpipe</td>
<td>TPIPE</td>
<td>Name of the OTMA relate TPIPE for this transmission</td>
</tr>
<tr>
<td>luname</td>
<td>LU name</td>
<td>Logical unit (LU) name that entered data</td>
</tr>
<tr>
<td>cnt/smb</td>
<td>SMB</td>
<td>Scheduler message block (SMB) or transaction name for this message</td>
</tr>
<tr>
<td>Short name</td>
<td>Description</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>CNT</td>
<td>Communication name table (CNT), refers to the logical terminal (LTERM) name associated with this message</td>
<td></td>
</tr>
<tr>
<td>CCB</td>
<td>Conversation control block (CCB) associated with this conversational message</td>
<td></td>
</tr>
<tr>
<td>RRE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>action</td>
<td>IMS action</td>
<td>Various IMS actions (started, stopped, etc.)</td>
</tr>
<tr>
<td>cnvflag1</td>
<td>Flag byte</td>
<td>From type 13 record</td>
</tr>
<tr>
<td>lnrflag</td>
<td>Flag byte</td>
<td>From type 14 and 15 records</td>
</tr>
<tr>
<td>sgnflag</td>
<td>Flag byte</td>
<td>From type 16 record</td>
</tr>
<tr>
<td>xlogflag</td>
<td>Flag byte</td>
<td>From type 18 record</td>
</tr>
<tr>
<td>doclogf1</td>
<td>Flag byte</td>
<td>From type 20 record</td>
</tr>
<tr>
<td>log22_flagl</td>
<td>Flag byte</td>
<td>From type 22 record</td>
</tr>
<tr>
<td>erlgerfl</td>
<td>Flag byte</td>
<td>From type 24 record</td>
</tr>
<tr>
<td>eeqlflg1</td>
<td>Flag byte</td>
<td>From type 25 record</td>
</tr>
<tr>
<td>iotblflg</td>
<td>Flag byte</td>
<td>From type 26 record</td>
</tr>
<tr>
<td>dbxflags</td>
<td>Flag byte</td>
<td>From type 27 record</td>
</tr>
<tr>
<td>l3aplg</td>
<td>Flag byte</td>
<td>From type 3A record</td>
</tr>
<tr>
<td>l3bplg</td>
<td>Flag byte</td>
<td>From type 3B record</td>
</tr>
<tr>
<td>l3cplg</td>
<td>Flag byte</td>
<td>From type 3C record</td>
</tr>
<tr>
<td>l3Dplg</td>
<td>Flag byte</td>
<td>From type 3D record</td>
</tr>
<tr>
<td>qlrluoowe reason codes</td>
<td>Flag byte</td>
<td>From type 3F record</td>
</tr>
<tr>
<td>caprtype</td>
<td>Flag byte</td>
<td>From type 47 record</td>
</tr>
<tr>
<td>slflg1</td>
<td>Flag byte</td>
<td>From type 4C record</td>
</tr>
<tr>
<td>slglgl2</td>
<td>Flag byte</td>
<td>From type 4C record</td>
</tr>
<tr>
<td>s3flg1</td>
<td>Flag byte</td>
<td>From type 63 record</td>
</tr>
<tr>
<td>s2flg2</td>
<td>Flag byte</td>
<td>From type 63 record</td>
</tr>
<tr>
<td>smflag</td>
<td>Flag byte</td>
<td>From type 64 record</td>
</tr>
<tr>
<td>error subtype</td>
<td>Flag byte</td>
<td>From type 67 record</td>
</tr>
<tr>
<td>error code</td>
<td>Flag byte</td>
<td>From type 67 record</td>
</tr>
<tr>
<td>olcflag1</td>
<td>Flag byte</td>
<td>From type 70 record</td>
</tr>
<tr>
<td>olcflag2</td>
<td>Flag byte</td>
<td>From type 70 record</td>
</tr>
<tr>
<td>cmscflg1</td>
<td>Flag byte</td>
<td>From type 6C record</td>
</tr>
<tr>
<td>Short name</td>
<td>Description</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>srvflg1</td>
<td>Flag byte</td>
<td>From type 6D record</td>
</tr>
<tr>
<td>lumflag1</td>
<td>Flag byte</td>
<td>From type 6E record</td>
</tr>
<tr>
<td># psts</td>
<td>Number of psts</td>
<td>Number of PSTs in system</td>
</tr>
<tr>
<td>abendcode</td>
<td>Abendcode</td>
<td>Abend code that was issued if the application has terminated abnormally</td>
</tr>
<tr>
<td>insertDBs</td>
<td></td>
<td>List of databases with inserts</td>
</tr>
<tr>
<td>deleteDBs</td>
<td></td>
<td>List of databases with deletes</td>
</tr>
<tr>
<td>replaceDBs</td>
<td></td>
<td>List of databases with replaces</td>
</tr>
</tbody>
</table>
Index

* comment indicator 61
* masking character 86, 90, 122
% masking character 90
%LUICI@00 command and CLIST 46
+ prompt indicator 49
= logical operator 89
=/> logical operator 89

A

ABEND keyword 110, 162, 202
Abend report 110
ABEND report 202
abends
  maximum logs exceeded 74
  search for abended LUOWs 94
Abends field 162
ABENDS subkeyword 94, 156
ABOUT command 50
About command on action bar 50
action bar commands 49
active OLDS 67, 251, 256
ACTIVEOLDS keyword 67, 256
Additional intervals field 160
After field 52
ALL command 154
Analysis action field 212
Analysis Summary field 162
Analysis Summary report 128, 178
ANALYZE control statement 63, 141
ACTIVEOLDS keyword 67
DB2BSDS keyword 68
DB2LOG keyword 68
ELAPSEDTIME keyword 69
IMSID keyword 70
INPEXT keyword 70
IPRJRNLS keyword 71
IPRPREPFX keyword 72
keyword summary 63
LIMIT keyword 73
LOGRECORDDPDS keyword 73
LRCPS keyword 73
MAXLOGS keyword 74
MQEXT keyword 75, 265
RECON keyword 76
SKIP keyword 76
SLDS keyword 77
TIMEZONE keyword 78
ANALYZE Control Statement panel 141
APPC
  LUNAME search 100
  TPNAME search 105
APPC_LTERM subkeyword 95, 156
APPCHECK keyword 111, 162, 207
Application Checkpoint field 162
Application Checkpoint report 111, 162, 206
Application Checkpoint report options field 162
application program search 103
architecture 38
asterisk on control statements 61
ATTR command 51, 53
Audit field 162
AUDIT keyword 115, 162, 186
AUDIT=
  Subkeywords 116
Auditing report 115, 185
audits 310
B

batch reports 291
Before field 52
BMC Log Analyzer product for IMS 33
BMC System Administration solution for IMS 33
BMC System Communication solution for IMS 33
Boolean operator field 156
Boolean operators 288
BOTH value of DUMPFORM keyword 119
Build JCL panel 139

C

CANCEL command 47, 51
CHAR value of DUMPFORM keyword 119
Checkpoint frequency fields 162
CHKFREQ keyword 162
CICS Tran summary field 184
CICS Transaction List panel 224
CICSTRAN subkeyword 95, 156
CLEAR command 154
CLIENTID subkeyword 95, 156
CMDS command 51
COLORS command 51, 53
Colors command on action bar 50, 53
Command area 50
commands
  action bar 49
  ALL 154
  CLEAR 154
  CREATE 171
  FILTER 215
  FIND 237, 242
  global ISPF 50
  overview 61
  PROMPT 152
  RESET 237, 242
  RFND 237, 242
  SORT 232
  SUBMIT 171, 172
  syntax rules 60
  VIEW 171
COMMANDS command 51, 53
comments on control statements 61
Comparison operator field 162
compression, storage 332
condition codes, analysis job 135
considerations for using Log Analyzer 33
CONTENT option field 155
CONTENT subkeyword 96, 156
CONTENT0103 keyword 84, 155
control statements
  ANALYZE 63, 141
  commands 60, 61
  comments 61
  END 135
  expressions 59
  EXTRACT 128, 168, 297
  FILTER 83, 151
  INDEXFILE 128, 168, 300
  INTERVAL 79, 160, 277
  keywords 60
  Log Analyzer 59
  minimum sets 62
  REPORTS 107, 162
  rules for specifying 59
  samples 59, 341
  syntax 59
  SYSIN DD statement 58
  values 61
  WORKFILE 128, 168
conventions, documentation 16
COPY1 RECON data set 76
Count of nonselected records by code section 185
Count of processed records by code section 184
CPU authorization password 33, 57
CPU cycle usage 35
CREATE command 171
CSU product code 33
CUA Attribute Change Utility pop-up window 53

D

data content search 84
databases
  searching log records 97, 103, 106
  updates against 312
DATACLASS keyword 132, 168
daylight savings time 78
db2 bootstrap data set 68
db2 BSDS Input panel 149
db2 Input panel 148
db2BSDS keyword 68
db2LOG keyword 68, 262
dbd subkeyword 97, 156
dcb data set name (non-SMS) field 168
DCB keyword 132, 168
deadlock analysis 226
DEADLOCK keyword 118, 204, 210
Deadlocks report 118, 203
DEIMAL or DEC value of OFFSET keyword 126
DEST subkeyword 98, 156
Destination List panel 220
destination search 98, 99, 105
Destination summary field 183
DRESERA10 utility 249
diagnostic system dump 58
Display 01 and 03 record data field 162
Display Logo panel at Startup field 53
documentation information 15
DOWN command 48
DSN keyword 132
dump data set 58
DUMPFORM keyword 119, 197
DUMPREC keyword 119, 162, 196
DUMPRERT keyword 120, 162, 196
DURATION field 178
dynamic allocation members 57

E

ELAPSEDTIME keyword 69, 141
END command 47, 51
END control statement 135
Energizer for IMS Connect journal files 71, 72
Energizer for IMS Connect journals 260
error messages 330
ETIME subkeyword 98, 156
examples

Boolean operators 288
complex SELECT filters 286
elapsed time 69
IMSID keyword 255
ISPF interface usage 30
active OLDS 251
active OLDS usage 67
ACTIVEOLDS keyword 256
ANALYZE control statement 63
application checkpoint frequency 114
audit report request 293
auditing user actions 310
batch report request 293
complex JCL 28
CONTENT search 287
content search in 01 and 03 records 84
control statement syntax 59
data search in 01 and 03 records 295
database update activity 312
ending (stop) range for logs and records 82
Energizer for IMS Connect journal 71, 72
EXTRACT control statement 129
extract file creation 297
extract file input 70, 299
extract file used as input 258
filter by search criteria 90
FILTER control statement 83
filter with masked values 286
FIRST and specific stop time 279
format of dumped records 295
generic JCL 56
implicit and explicit specification of logs 257
IMS system selection 70
index file creation 300
INDEXFILE control statement 129
INTERVAL 278
INTERVAL control statement 80
LIMIT keyword 271
limited log records read from log 73
log data set names specified explicitly 77
log record code PDS 73, 275
log record code PDS member 276
log record codes 294
log record types included and excluded 87
log sequence number interval 279
maximum number of logs 74
message-switching transactions 313
minimum control statement sets 62
MSC traffic 313
multiple intervals 280
multiple RECON keywords 253
RECON data set allocation through MDA members 254
RECON data set allocation with named SLDS 253
RECON data set names 76
remote system data extracted 259
reports 28
REPORTS control statement 108
select by filter criteria 90
SELECT DBD 284
SELECT USERID 286
simple JCL 27
SKIP keyword 271
skipped log records in SLDS 76
specific start time and LAST 279
START and STOP with generic time values 278
starting range for logs and records 80
time zone 78
transaction log record code filter 273
WebSphere for MQ extract file 75
WORKFILE control statement 129
Excluded log codes field 152
EXEC statement 57
Exit command on action bar 50
explicit specification of log record sources 250
expressions, control statement 59
Extended Help command on action bar 50, 53
EXTRACT control statement 128, 168, 297

keyword summary 129
EXTRACT Control Statement panel 168
Extract file count field 181
Extract file field 146, 148–150
Extract File Input panel 146
extract files
allocating 128
creation 296
data set name 132
input 70, 257
out-of-space conditions 297
reformat of 299

F

FAQs 329

fields

Abends 162
Additional intervals 160
After 52
Analysis action 212
Analysis Summary 162
Application Checkpoint 162
Application Checkpoint report options 162
Audit 162
Before 52
Boolean operator 156
Checkpoint frequency 162
CICS Tran summary 184
Command 50
Comparison operator 162
CONTENT option 155
DCB data set name (non-SMS) 168
Destination summary 183
Display 01 and 03 record data field 162
Display Logo panel at Startup 53
DURATION 178
Excluded log codes 152
Extract file 146, 148–150
Extract file count 181
FILTER option 151
Format 52
Generic unit 168
Help information 53
IMSIDs 143
IMSIDs encountered 181
Included log codes 152
JCL Actions 138
Job card 139
Jobnames 162
Jobs to include or exclude 162
LIKE data set name (SMS) 168
Limit SYSOUT to n lines 162
Limit to LUOW ID(s) field 162
Log Analyzer option 46
Log record code PDS 141
Log Record Dump 162
Log record dump panel options 52
Log Record Dump report options 162
Log record limit 144
Log Records - Time Sequence panel options 52
Log records read 181
Log time span 178
LUOW Detail 162
LUOW Summary 162
LUOW Time Sequence 162
LUOWs 181
Maximum logs options 143
Number of checkpoints 162
Origin summary 182
Orphans 162
Other IMSIDs encountered 182
Other summary 184
Passed to select process 181
PDS member names 152
Per threshold 162
Primary quantity 168
PSB names 162
PSB summary 183
PSBs to include or exclude 162
RBA Buffer Statistics 162
RECON n 143
Relational Operator 156
Remote IMS connections found 182
Report index file 212
Reports selection 162
Secondary quantity 168
Select a SELECT subkeyword to insert 156
SELECT subkeywords 155
Selected log records 181
Skip to log record 144
SLDS data set name 144
SMS Data class 168
SMS Management class 168
SMS Storage class 168
Space units 168
Specify input 141
Start type 160
Startup Preferences 53
STEPLIB libraries 139
Stop type 160
Suppress display of this panel on startup 46
Suppress elapsed time calculations 141
Time zone 141
Type of Energizer for IMS Connect journal data sets 147
Unresolved associations 181
User ID summary 183
Volume serial(s) 168
File menu on action bar 50
FILTER - Log Record Code panel 152
FILTER - Select Subkeywords panel 155
FILTER command 215
FILTER control statement 83, 151

ABENDS subkeyword 94, 156
APPC_LTERM subkeyword 95, 156
CICSTRAN subkeyword 95, 156
CLIENTID subkeyword 95, 156
CONTENT subkeyword 96, 156
CONTENT0103 keyword 84
DBD subkeyword 97, 156
DEST subkeyword 98, 156
ETIME subkeyword 98, 156
keyword summary 83
LOGRECORDCODES keyword 85
LRC keyword 85
LTERM subkeyword 156
LUNAME subkeyword 100, 156
MPR subkeyword 100, 156
NODE subkeyword 101, 156
OIMS subkeyword 101, 156
ORIGIN subkeyword 102, 156
PORT# subkeyword 102, 156
PSB subkeyword 103, 156
PST subkeyword 103, 156
RDBD subkeyword 103, 156
SELECT keyword 89
TMEMBER subkeyword 104, 156
TPIPE subkeyword 104, 156
TPNAME subkeyword 105, 156
TRAN subkeyword 105
UDBD subkeyword 106
UOW1 subkeyword 106
USERID subkeyword 107, 156
FILTER Control Statement panel 151
Filter LUOWs pop-up window 215
FILTER option field 151
filters
multiple 288
types 284
values 286
FIND command 237, 242
FIRST value of START keyword 80
Format field 52
function keys
definitions 53
preferences 53
Function keys command on action bar 50, 53

G

Generic unit field 168
GMT 78
HELP command 51, 53
Help for help command on action bar 50, 53
Help menu on action bar 50
Help panels 53
HEX value of DUMPFORM keyword 119
HEX value of OFFSET keyword 126

implicit specification of log record sources 252
IMS
  specifying for analysis 70
  version that created log 77
IMS Connect 71, 72
IMS Connect input 260
IMS Connect Input panel 147
IMSID keyword 70, 143, 255
IMSIDs encountered field 181
IMSIDs field 143
IMSIDs, nonunique 35
Included log codes field 152
index files
  allocating 128
  creation 300
  data set name 132
  out-of-space conditions 300
INDEXFILE control statement 128, 168, 300
  keyword summary 129
INDEXFILE Control Statement panel 168
INPEXT keyword 70, 146, 148–150, 257
Insert SELECT Subkeyword pop-up window 156
INTERVAL control statement 79, 160, 277
  keyword summary 80
  START keyword 80
INTERVAL Control Statement panel 160
IPRJRNLS keyword 71, 147, 260
IPRPREF keyword 147
IPRPREFIX keyword 72
IPT product code 33
ISPF interface
  CANCEL command 51
  ISP command 51
UP command 48
VERSION command 51
ISPPROF data set 45

J

JCL
STEPLIB DD statement 57, 139
typical example 39
building through ISPF interface 137
complex example 28
creation 56, 137
EXEC statement 57
generic 56
JOB statement 56, 139
LUITRACE DD statement 58
samples 56
simple example 27
STEPLIB DD statement 57, 139
SYSIN DD statement 58
SYSMDUMP DD statement 58
SYSOUT DD statement 58
JCL Actions field 138
JCL Menu panel 138
Job card field 139
JOB keyword 162
JOB statement 56, 139
Jobnames field 162
Jobs to include or exclude field 162
journals for IMS Connect 71, 72

K

KEYLIST command 53
Keylist Utility for LUI pop-up window 53
keywords

APPCLTERM 95, 156
DUMPVER 120, 162, 196
ABEND 110, 162, 202
ABENDS 94, 156
ACTIVOLD 67, 256
APPCLTERM 95, 156
APPCHECK 111, 162, 207
AUDIT 115, 162, 186
CHKFREQ 162
CICSTAN 95, 156
CLIENTID 95, 156
CONTENT 96, 156
CONTENT103 84, 155
DATACLASS 132, 168
DB2BDS 68
DB2LOG 68, 262
DBD 97, 156
DCB 132, 168
DEADLOCK 118, 204, 210
DEST 98, 156
DSN 132
DUMPFORM 119, 197
DUMPPREC 119, 162, 196
DUMPVER 120, 162, 196
ELAPSEDTIME 69, 141
ETIME 98, 156
IMSID 70, 143
INPEXT 70, 146, 148–150, 257
IPRJRN 71, 147, 260
IPRPREF 147
IPRPREFIX 72
JOB 162
LIKE 133, 168
LIMIT 73, 144, 270
LOGRECORC 73, 141
LOGRECORC 85, 122, 152, 271
LOGRECORC 152, 275
LRC 85, 122
LRCPDS 73
LTERM 156
LUNAME 100, 156
LUOWDETAIL 123, 162, 192
LUOWDETAILDATA 124, 162, 193
LUOWSUMM 125, 162, 189
LUOWTSEQ 125, 162, 199
MAXLOGS 74, 143, 255
MGMTCLASS 133, 168
MPR 100, 156
MQEXT 75, 265
NODE 101, 156
OFFSET 126
OIMS 101, 156
ORIGIN 102, 156
ORPHANS 126, 162, 201
PORT# 102, 156
PRISPACE 133, 168
PSB 103, 156, 162
PST 103, 156
RBASTATS 127, 162
RDBD 103, 156
RECON 76, 143, 253
SECSPACE 134, 168
SELECT 89, 155, 284
SKIP 76, 144, 270
SLDS 77, 144, 251
SPACEUNITS 134, 168
START 80, 160
STOP 82, 160
STORCLASS 134, 168
SUMMARY 128, 162, 179
syntax rules 60
THRESH 162
TIMEZONE 78, 141
TMEMBER 104, 156
TPIPE 104, 156
TPNAME 105, 156
TRAN 105, 156
UDBD 106, 156
UNIT 134, 168
UOW1 106
USERID 107, 156
VOLSER 135, 168

L

LAST value for STOP keyword 82
LIKE data set name (SMS) field 168
LIKE keyword 133, 168
LIMIT keyword 73, 144, 270
Limit SYSOUT to n lines field 162
Limit to LUOW ID(s) field 162
LISTRECS command 235
local time zone 78
Log Analyze
  reports 42
Log Analyzer
batch approach to usage 27
interactive approach to usage 30
allocation of log record sources 249
architecture 38
auditing usage 310
basic tasks 37
batch reports 175
considerations 33
control statements 59
getting started 37
introduction 19
ISPF interface usage 45
ISPF reports 211
JCL 56, 137
load modules 57
log record candidates 269
LUOW selection process 289
messages 330
procedure for usage 44
processes 38
program for executing 57
record selection process 289
reports 175, 211
samples 337
troubleshooting 329
Log Analyzer for IMS panel 46, 53
Log Analyzer option field 46
Log code help command on action bar 50
log data sets
  active OLDS 67
  allocation 249
  finding from RECONs 76
  IMS system that produced 70
  maximum number to select 74
  names 77
  range for selection 80, 82
  SLDS 77
log record code PDS 73, 275
data set creation 276
member creation 276
Log record code PDS field 141
log record codes
  automatic inclusion 274
  filtering by 86, 122
  masked 86, 122
  selecting from a prompt 154
  selection 271
Log Record Detail panel 240
Log Record Dump field 162
Log record dump panel options field 52
Log Record Dump report options field 162
Log record limit field 144
Log Record logrecnum panel 245
log records
  abend search 94
  allocation of sources 249
  automatically included types 369
  candidates 269
  CICS transaction search 95
  client ID search 95
  content search 96
  content searches 84
  database search 97, 103, 106
  destination search 98
  excluding types 85
  including types 85
  limited number read from log 73
  LUNAME search 100
  node search 101
  origin search 102
  port search 102
  program specification block search 103
  range for selection 80, 82
  reporting 122
  search criteria 89
  selection of types 85
  selection process 289
  skipped 76
  TMEMBER search 104
  TPIPE search 104
  TPNNAME search 105
  transaction search 105
  types 345
  types for consideration 85
  types included 73
  types included automatically for USERID searches 107
  user ID search 107
Log Records - Time Sequence panel 242
Log Records - Time Sequence panel options field 52
Log records read field 181
log sequence number 80
Log time span field 178
LOGCODES command 51
logical operators 286
logical-not symbol to exclude log record codes 88
LOGRECORDCODEPDS keyword 73, 141
LOGRECORDCODES keyword 85, 122, 152, 271
LOGRECORDCODESPDS keyword 152, 275
logs, DB2 68
LRC keyword 85, 122
LRCPDS keyword 73
LTERM search 102
LTERM subkeyword 99, 156
LUI product code 33
LUI#xxxx samples 337
LUIMAIN program 57
LUISYNTX sample 341
LUITRACE DD statement 58, 329
LUIxxxx samples 338
LUIZUSER member and panel 45, 54
LUNAME subkeyword 100, 156
LUOW Detail field 162
LUOW Detail report 192
  requesting 123
  requesting data fields 124
LUOW List panel 216
LUOW Lists section 182
LUOW luowid Log Records panel 237
LUOW luowid panel 235
LUOW numbers 304
LUOW Summary field 162
LUOW Summary report 125, 188
LUOW Time Sequence field 162
LUOW types 302
LUOWDETAIL keyword 123, 162, 192
LUOWDETAILDATA keyword 124, 162, 193
LUOWs
  creation 269
  descriptions 367
  search for long-running 98
  selection process 289
LUOWs field 181
LUOWSUMM keyword 125, 162, 189
LUOWTSEQ keyword 125, 162, 199

M
Main Menu panel 46
masking character 86, 90, 122
Maximum logs options field 143
MAXLOGS keyword 74, 143, 255
MDA library 57
memory usage 35
Menu action on action bar 50
Message nnnnnn - Help pop-up window 330
message-switching transactions 313
messages 330
Messages - Entry pop-up window 330
Messages - List panel 330
MESSAGES command 51
Messages command on action bar 50
MGMTCLASS keyword 133, 168
minimum control statements 62
miscellaneous LUOWs 303
MORE indicator 48
MPR subkeyword 100, 156
MPR, filtering by 100
MQ extract data set 75, 265
MQ Extract panel 150
MQEXT keyword 75, 265
MSC traffic
  example 313
  nonunique IMSIDs 35
MSGS command 51

N
navigation in the ISPF interface 47
NODE subkeyword 101, 156
noncataloged log data sets 77
noncataloged SLDS 251
nontransaction LUOWs 303, 367
nonunique IMSIDs 35
NOT= logical operator 89
Number of checkpoints field 162

O
OFFSET keyword 126
OIMS subkeyword 101, 156
OLDS, active 67
Options menu on action bar 50
Origin List panel 221
origin search 99, 102
ORIGIN subkeyword 102, 156
Origin summary field 182
original time zone 78
Orphans field 162
ORPHANS keyword 126, 162, 201
Orphans report 200
Other IMSIDs encountered field 182
other LUOWs 367
Other LUOWs List panel 225
Other summary field 184
OTMA

port search 102
TMEMBER search 104
TPIPE search 104
out-of-space conditions 297, 300

P
panels
ANALYZE Control Statement 141
Build JCL 139
CICS Transaction 224
CUA Attribute Change Utility 53
DB2 BSDS Input 149
DB2 Input 148
Deadlock Victims List 226
Destination List 220
EXTRACT Control Statement 168
Extract File Input 146
FILTER - Log Record Code 152
FILTER - Select Subkeywords 155
FILTER Control Statement 151
Filter LUOWs 215
Help 53
IMS Connect Input 147
INDEXFILE Control Statement 168
Insert SELECT Subkeyword 156
INTERVAL Control Statement 160
JCL Menu 138
Keylist Utility for LUI 53
Log Analyzer for IMS 46, 53
Log Record Detail 240
Log Record logrecnum 245
Log Records - Time Sequence 242
LUIZUSER 45, 54
LUOW List 216
LUOW luowid Log Records 237
Main Menu 46
Message nnnnn - Help 330
Messages - Entry 330
Messages - List 330
MQ Extract 150
Origin List 221
Other LUOWs List 225
PSB List 223
RECON Input 143
Report Index File Analysis 212
Report Index File Summary 229
REPORTS Control Statement 162
Select Log Code 154
SLDS Input 144
Sort pull-down menu 232
Sort type List 232
User Preferences 52
Userid List 222
WORKFILE Control Statement 168
Passed to select process field 181
password 33, 57
PDS member names field 152
Per threshold field 162
PFSHOW command 53
plus sign (+) prompt indicator 49
PORT# subkeyword 102, 156
PREF command 51, 52
Preferences command on action bar 50, 52
Primary quantity field 168
PRISPACE keyword 133, 168
product authorization load library 57
product authorization module 57
profile values 171
profile variables 45, 54
programs
LUIMAIN 57
LUIRFEXT 299
PROMPT command 49, 152
prompt indicator 49
PSB keyword 162
PSB List panel 223
PSB names field 162
PSB subkeyword 103, 156
PSB summary field 183
PSBs to include or exclude field 162
PST subkeyword 103, 156
publications, related 15
R
RBA Buffer Statistics field 162
RBA Buffer Statistics report 210
RBASTATS keyword 127, 162
RDBD subkeyword 103, 156
read-only databases 103
RECON data sets
allocation through MDA library 57
allocation through MDA members 254
finding active OLDS 67
finding log data sets 76
RECON Input panel 143
RECON keyword 76, 143, 253
RECON n field 143
RECON processing modules 57
Record and LUOW Counts section 180
Record Dumps report 195
requesting 119, 120
requesting format options 119, 126
Reformat Extract File utility 299
REGION parameter on EXEC statement 57
related publications 15
Relational Operator field 156
Remote IMS connections found field 182
Report Index File Analysis panel 212
Report index file field 212
Report Index File Summary panel 229
reports
Abend 110
ABEND 202
accessing 175
Analysis Summary 178
Application Checkpoint 206
Auditing 185
batch 175
comparison of batch and ISPF 42
Count of nonselected records by code 185
Count of processed records by code 184
Deadlocks 203
headings 178
ISPF 211
LUOW Detail 192
LUOW Summary 188
Orphans 200
production of 175, 291
RBA Buffer Statistics 210
Record and LUOW Counts section of Analysis Summary 180
Record Dumps 195
Time Sequence 198
REPORTS control statement 107, 162
APPCHECK keyword 111
AUDIT keyword 115
DEADLOCK keyword 118
DUMPFORM keyword 119
DUMPREC keyword 119
DUMPPVERT keyword 120
keyword summary 108
LOGRECORDCODES keyword 122
LRC keyword 122
LUOWDETAIL keyword 123
LUOWDETAILDATA keyword 124
LUOWSUMM keyword 125
LUOWTSEQ keyword 125
OFFSET keyword 126
ORPHANS keyword 126
RBASTATS keyword 127
SUMMARY keyword 128
REPORTS Control Statement panel 162
Reports selection field 162
RESET command 237, 242
resource consumption 35
return codes
-analysis job 135
maximum logs exceeded 74
RFND command 237, 242
Row indicator 48
samples
%LUICI@00 46
control library 337
JCL 337
LUIZUSER 45, 54
sample library 338
scroll commands 48
search for log records 89
Secondary quantity field 168
SECSPACE keyword 134, 168
security access controls 34
Select a SELECT subkeyword to insert field 156
SELECT keyword 89, 155, 284
Select Log Code pop-up window 154
SELECT subkeywords 91
SELECT subkeywords field 155
Selected log records field 181
SKIP keyword 76, 144, 270
Skip to log record field 144
SLDS data set name field 144
SLDS Input panel 144
SLDS keyword 77, 144, 251
SMS Data class field 168
SMS Management class field 168
SMS Storage class field 168
SORT command 232
Sort pull-down menu 232
Sort type List pop-up window 232
Space units field 168
SPACEUNITS keyword 134, 168
Specify input field 141
START keyword 80, 160
Start type field 160
Startup Preferences field 53
STEPLIB DD statement 57, 139
STEPLIB libraries field 139
STOP keyword 82, 160
Stop type field 160
storage compression 332
STORCLASS keyword 134, 168
string search 96
SUBMIT command 171, 172
SUMMARY keyword 128, 162, 179
Suppress display of this panel on startup field 46
Suppress elapsed time calculations field 141
syntax
    rules 59
    samples 341
syntax statement conventions 16
SYSIN DD statement 58
SYSMDUMP DD statement 58
SYSOUT DD statement 58

T

THRESH keyword 162
Time Sequence report 125, 198
time shift, analysis across 78
Time zone field 141
TIMEZONE keyword 78, 141
TMEMBER subkeyword 104, 156
TPIPE subkeyword 104, 156
TPNAME subkeyword 105, 156
Trace facility 329
TRAN subkeyword 105, 156
transaction LUOWs 303, 366
transaction search 95, 98, 105
troubleshooting 329
TSO %LUICI@00 command 46
Type of Energizer for IMS Connect journal data
    sets field 147

U

UDBD subkeyword 106, 156
UNIT keyword 134, 168
unit of work 106
unmapd log record type 345
unresolved association 282
Unresolved associations field 181
UOW-1 numbers 304
UOW1 subkeyword 106
UP command 48
updated databases 106
User ID summary field 183
user ID, searching for log records 107
User Preferences pop-up window 52
user preferences, ISPF 51
Userid List panel 222
USERID subkeyword 107, 156
UTC 78

V

values, syntax rules 61
variable data 304, 374
VERSION command 51
VIEW command 171
VOLSER keyword 135, 168
Volume serial(s) field 168

W

WebSphere for MQ extract data set 75, 265
work files, allocating 128
WORKFILE control statement 128, 168
WORKFILE Control Statement panel 168

X

X’5F’ logical-not symbol to exclude log record
codes 88