



Administering Telco Service & Analytics Manager Applications

V4.2
Document ID: TMGN-03-4.2-01
Date Published: 3.5.04

© 1997–2004 edocs® Inc. All rights reserved.

edocs, Inc., One Apple Hill Dr., Natick, MA 01760

The information contained in this document is the confidential and proprietary information of edocs, Inc. and is subject to change without notice.

This material is protected by U.S. and international copyright laws. edocs and eaPost are registered in the U.S. Patent and Trademark Office.

No part of this publication may be reproduced or transmitted in any form or by any means without the prior written permission of edocs, Inc.

eaSuite, eaDirect, eaPay, eaCare, eaAssist, eaMarket, and eaXchange are trademarks of edocs, Inc.

All other trademark, company, and product names used herein are trademarks of their respective companies.

Printed in the USA.

Preface

In This Section

Using this Manual.....	iv
Finding the Information You Need	x
If You Need Help.....	xiv

Using this Manual

Welcome to Administrating Telco Service & Analytics Manager Applications.

This manual covers the different tasks that administrators need to carry out when working with solutions built using edocsb Telco Service Manager (TSM) and Telco Analytics Manager (TAM).

Before You Get Started

You should be familiar with the following:

- Your application architecture
- Programming Java and Java Server pages
- Designing or working with databases
- eXtended Markup Language (XML)

Who Should Read this Manual

This manual is primarily for administrators of deployed solutions. However, there are other topics covered in this manual that may interest other members of the project development team.

- **Administrators**

You will find the information you need to manage a deployed solution. The information in this manual starts with information about the basic administration tasks. You will also find detailed information about other administration tasks, such as purging information, fine-tuning specific application settings, running the CID2CBU and CustDim loaders, or interpreting logs to pinpoint problems.

- **Developers**

You will find information about running and administering components of your development environment. You may especially be interested in the basic administration tasks and the detailed administration information dealing with various components, such as the way notifications are managed and how the CID2CBU Loader handles the synchronization of changes. The chapter on the CustDim Loader has information about how the administrator uses and troubleshoots the loading of Customer Dimensions.

You may also want to consult the Administration Tool Reference for the comprehensive list of administration tools and their commands.

- **Project Architect**

You will find information about the different components which may require administration or intervention by administrators. There are some features that may be of interest when designing your solution. For instance, you need to be aware of how requests are managed and how the solution handles persistent action managers. When administering analytical applications, you need to know about the CustDim loader as it is run manually to load the CBU. You may also be interested in the various ways of auditing the application which are explained in detail.

You should also look at the sections covering distributed architectures.

- **Project Manager**

You will find information about the configuration and processes of the runtime environment you need to take into account when planning the development of your solution. There is important information about how the solution handles requests. It also explains how to reload reference information during runtime. Other information includes sections about the loaders and synchronization of information in the CBU. You can also look into the auditing and logging features of the solution.

How this Manual is Organized

This manual contains the following chapters:

- **Overview of Administrating**

This chapter covers the basics of managing Telco Service Manager (TSM) solutions:

- The role of an administrator
- Different components and third-party software these components use
- Configuration files and their location

- **Basic Administration Tasks**

This chapter covers the basic administration tasks:

- Starting and stopping the Synchronizer connector
- Starting and stopping the OSS connector
- Starting and stopping the Approval Sequencer
- Starting and stopping the CID2CBU Loader
- Starting and stopping the Custdim Loader

- **Administrating Telco Service Manager (TSM)**

This chapter covers the following administration tasks:

- Managing the Synchronizer connector
- Managing requests
- Managing trouble tickets
- Managing persistent action managers
- Reloading reference data
- Auditing applications

- **Administrating Telco Analytics Manager (TAM)**

This chapter covers the following administration tasks:

- Managing notifications
- Working with the CID2CBU Loader
- Working with the CustDim Loader
- Managing Reports

- Auditing applications

- **Monitoring Telco Service & Analytics Manager Applications**

This chapter covers the following administration tasks:

- Using the logger
- Integrating the logger into a supervision platform

- **Auditing Telco Service & Analytics Manager Applications**

This chapter covers the following administration tasks:

- Activating user events
- Creating reports

- **Managing Distributed Architectures**

This chapter covers administrating solutions in distributed architectures:

- Configuring shared directories
- Managing sessions

- **Administration Tool Reference**

This appendix is an administration tool reference guide. It covers the location, configuration and use of the administration tools

The administration tools covered include:

- Synchronizer connector tools
- OSS connector tools
- Approval Sequencer tools
- CID Administration tool
- CBU Administration Tool
- NOTIFYORG Administration Tool
- CID2CBU Loader Administration Tools
- CustDim Loader Administration Tools

What Typographical Changes and Symbols Mean

This manual uses the following conventions:

TYPEFACE	MEANING	EXAMPLE
<i>Italics</i>	Manuals, topics or other important items	Refer to <i>Developing Connectors</i> .
Small Capitals	Software and Component names	Your application uses a database called the CID.
Fixed Width	File names, commands, paths, and on screen commands	Go to <code>//home/my file</code>

Obtaining edocs Software and Documentation

You can download edocs software and documentation directly from Customer Central at <https://support.edocs.com>. After you log in, click on the Downloads button on the left. When the next page appears, you will see a table displaying all of the available downloads. To search for specific items, select the Version and/or Category and click the Search Downloads button. If you download software, an email from edocs Technical Support will automatically be sent to you (the registered owner) with your license key information.

If you received an edocs product installation CD, load it on your system and navigate from its root directory to the folder where the software installer resides for your operating system. You can run the installer from that location, or you can copy it to your file system and run it from there. The product documentation included with your CD is in the Documentation folder located in the root directory. The license key information for the products on the CD is included with the package materials shipped with the CD.

Finding the Information You Need

The product suite comes with comprehensive documentation set that covers all aspects of building solutions based on the edocs Telco Service & Analytics Manager. You should always read the release bulletin for late-breaking information.

Getting Started

If you are new to the edocs Telco Solutions, you should start by reading *Introducing Telco Service & Analytics Manager Applications*. This manual contains an overview of the various components along with the applications and their features. It introduces various concepts and components you must be familiar with before moving on to more specific documentation. Once you have finished, you can read the manual which covers different aspects of working with the application. At the beginning of each manual, you will find an introductory chapter which covers concepts and tasks.

Designing Your Solution

While reading *Introducing Telco Service & Analytics Manager Applications*, you should think about how the different components can address your solution's needs.

You can refer to *Developing Telco Service Manager (TSM)* for information about extending the object model, application security, and other design issues. The *CID Reference Guide* also gives you the information about how the information in your solution is managed and stored.

You can refer to *Developing Telco Analytics Manager (TAM)* for information about customizing the database, synchronizing data with TSM, loading data from external invoice files, and other design issues. The *CBU Reference Guide* also gives you the information about how the information in your solution is managed and stored. You should also read the section on integrating TAM with TSM in *Developing Telco Analytics Manager (TAM)*.

You can also read the introduction of *Developing Connectors* for information about integrating your solution.

Installing Telco Service & Analytics Manager Applications

You should start by reading the Release Bulletin. For detailed installation and configuring information, refer to *Installing Telco Service & Analytics Manager Applications*. This manual covers installing applications on one or more computers. It also contains the information you need to configure the different components you install.

You might also refer to *Developing Telco Service & Analytics Manager Applications* and *Developing Connectors* as these manuals contain information on customizing applications and working with other software.

If you are upgrading, be sure to read *Migrating Telco Service & Analytics Manager Applications*.

Building Your Solution

If you are designing and programming your solution, you have several different sources of information. If you are programming the user interface of the solution, you should read *Developing User Interfaces*. You also refer to the *BLM Specification* and *JSPF specification* for detailed information about programming the user interface. For configuring the various components, you refer to *Installing Telco Service & Analytics Manager Applications* and sections in other documents which deal with the component to configure.

If you are designing and programming TAM, you have several different sources of information. If you are programming the user interface of the solution, you should read *Developing Reports*. You also refer to the *QRA API Specification* and the *QRA Configuration File Reference Documentation* for detailed information about the different components you can use to build reports. For configuring the various components, you refer to *Installing Telco Service & Analytics Manager Applications* and sections in other documents which deal with the component to configure.

If you are working with the business logic of your solution, you should read *Developing Telco Service Manager (TSM)*. You can also refer to the *BLM Reference Guide* for more information about the design and structure of the BLM object model. For information about how this information is stored, you should refer to the *CID Reference Guide* along with the *CID Reference* documentation for your database. In order to develop your application, you most likely will need to install and run the Loopback Connector. This component mimics back-end applications for development purposes. For information about installing and running this component, refer to *Using the Loopback Connector*.

If you are working on the data warehouse side of TAM, you should read *Developing Telco Analytics Manager (TAM)*. For more information about the design and structure of the CBU, you should refer to the *CBU Reference Guide* along with the *CBU Reference* documentation for your database. You should also read *Developing Telco Analytics Manager (TAM)* for information about synchronizing data between the TAM and *Telco Service Manager (TSM)*. In this manual, you will also find information about loading data in both the CBU and the CID.

For more information about integrating your application, you should read *Building Connectors* to learn how Telco Service & Analytics Manager applications work with different software.

Integrating Your Solution

If you are involved in configuring your solution to work with Operation Support Software (OSS), you should read *Building Connectors*. This manual helps you understand the integration architecture and shows you how to build connectors to connect to today's market-leading OSS software. You can also read *Using the Loopback Connector* for information about a connector built for development purposes. Other manuals you can refer to for information about configuring your application include *Introducing Telco Service & Analytics Manager Applications*, *Developing Telco Analytics Manager (TAM)*, and *Developing Telco Service Manager (TSM)*.

Managing Telco Service & Analytics Manager Applications

If you are responsible for managing Telco Service & Analytics Manager applications, you should read the *Installing Telco Service & Analytics Manager Applications* for information about configuring various components and information about working with different application servers. *Administering Telco Service & Analytics Manager Applications* covers what you need to know about managing your solution at runtime. For information about OSS systems, you should read *Building Connectors*.

If You Need Help

Technical support is available to customers who have valid maintenance and support contracts with edocs. Technical support engineers can help you install, configure, and maintain your edocs application.

edocs provides global Technical Support services from the following Support Centers:

US Support Center

Natick, MA
Mon-Fri 8:30am – 8:00pm US EST
Telephone: 508-652-8400

Europe Support Center

London, United Kingdom
Mon-Fri 9:00am – 5:00 GMT
Telephone: +44 20 8956 2673

Asia Pac Rim Support Center

Melbourne, Australia
Mon-Fri 9:00am – 5:00pm AU
Telephone: +61 3 9909 7301

Customer Central

<https://support.edocs.com>

Email Support

<mailto:support@edocs.com>

When you report a problem, please be prepared to provide us the following information:

- What is your name and role in your organization?
- What is your company's name?
- What is your phone number and best times to call you?
- What is your e-mail address?
- In which edocs product did a problem occur?
- What is your Operating System version?
- What were you doing when the problem occurred?
- How did the system respond to the error?
- If the system generated a screen message, please send us that screen message.

- If the system wrote information to a log file, please send us that log file.

If the system crashed or hung, please tell us.

Contents

Preface	iii
----------------	------------

Overview of Administrating	21
-----------------------------------	-----------

About Administrating	22
Overview of the Application Architecture	23
About Configuration Files and their Location	24

Basic Administration Tasks	25
-----------------------------------	-----------

About the Basic Tasks	26
Starting and Stopping the Synchronizer Connector	27
Starting and Stopping the OSS Connector	28
Starting and Stopping the Approval Sequencer	29
Starting and Stopping the CID2CBU Loader	30
Starting and Stopping the CustDim Loader	31
Starting and Stopping the Job Processor	32

Administrating Telco Service Manager (TSM)	33
---------------------------------------------------	-----------

About Administrating TSM	34
Managing the Synchronizer Connector	35
Checking the Synchronizer Connector	35
Purging the Synchronizer Queue	37
Managing Requests	38
About Requests	38
Viewing the Requests	39
Cancelling Requests	40
Resubmitting Requests	42
Purging Requests	43
Managing Jobs	44
About Jobs	44
Purging Jobs	44
Resubmitting Jobs	45
Cancelling Jobs	45
Managing Persistent Action Managers	47
About Persistent Action Managers	47
Purging Persistent Action Managers	48
Managing Trouble Tickets	49
About Trouble Tickets	49
Purging Trouble Tickets	49
Managing the SmartLink (ISF) Message Cache	50
About the SmartLink (ISF) Message Cache	50
Working with the SmartLink (ISF) Message Cache Structure	51
Purging the SmartLink (ISF) Message Cache	52

Reloading Reference Data	53
About the BLM Cache	53
Limits to Using the Reference Data Reload Feature	54

Administrating Telco Analytics Manager (TAM) **57**

About Administrating TAM	58
Managing Notifications	59
About Managing Notifications	59
Notification Status Codes	60
Viewing Notifications	60
Cancelling Notifications	62
Purging Notifications	63
Resubmitting Notifications	64
Managing Organization Notifications	66
Managing the CustDim Loader	69
About Working with the CustDim Loader	69
Running the CustDim Loader	69
Monitoring the CustDim Loader	71
Recovering from Errors	73
Managing Saved Reports	75
About Managing Reports	75
Purging Reports	75

Monitoring Telco Service & Analytics Manager Applications **77**

About Monitoring Applications using Logs	78
Logger Events	80
Event Types	81
Severity Levels	82
Event Modules	83
Overview of Using Logs	84
Determining the Goals	84
Determining the Components to Monitor	84
Determining the Need for Dynamic Configuration	85
Determining the Level of Information	86
Determining the Output Format	86
Configuring the Logger	87
Logger Configuration Files	88
Setting the Basic Properties	89
Specifying the Events to Log	91
Configuring the Logger Output	93
Examples of Logger Files	103
Using Logs with a Supervision Platform	105
Adapting the Output Format	105
Using Event Attributes	106
Using Event Codes	107
Filtering by Source Component	108
Examples of Filtering by Component	109

Auditing Telco Service & Analytics Manager Applications	111
About Auditing	112
Activating User Events	113
Generating Reports	114
Purging User Events	115
Managing Distributed Architectures	117
About Distributed Architectures	118
Installing and Configuring for Distributed Architectures	119
Managing Sessions	120
Administration Tool Reference	121
About the Administration Tools	122
CID Administration Tool	123
Configuring the CID Administration Tool	123
cidAdminTool Syntax	125
CID2CBU Loader Administration Tools	133
Configuring the CID2CBU Loader	133
cid2cbuloader Syntax	139
cid2cbuloaderadm Syntax	140
CBU Administration Tool	143
Configuring the CBU Administration Tool	143
cbuAdminTool Syntax	145
Synchronizer Connector Administration Tools	147
agentstart Syntax	147
agentadm Syntax	148
OSS Connector Administration Tools	151
ossstart Syntax	151
ossadm Syntax	152
NOTIFYORG Administration Tool	153
Configuring the NOTIFYORG Administration Tool	153
notifyorg Syntax	154
Approval Sequencer Administration Tools	155
approvalsequencer Syntax	155
approvalsequenceradm Syntax	156
CustDim Loader Administration Tools	157
Configuring the CustDim Loader Database Connection	157
custdim Syntax	158
custdimadm Syntax	159
SmartLink (ISF) Message Cache Administration Tool	161
Configuring the SmartLink (ISF) Message Cache Administration Tool	161
messagecacheAdminTool Syntax	163
Report Manager Administration Tool	165
Configuring the Report Manager Administration Tool	165
savedreportAdminTool Syntax	167
Job Service Administration Tools	168
Configuring the Job Service Administration Tool	169
jobAdminTool Syntax	171
jobprocessorstart Syntax	173
jobprocessoradm Syntax	174

CHAPTER 1

Overview of Administrating

In This Section

About Administrating	22
Overview of the Application Architecture	23
About Configuration Files and their Location	24

About Administrating

edocs Telco Service & Analytics Manager applications are designed to run without user intervention or maintenance. To ensure that your system runs smoothly and meets the availability and performance demands your users expect, you have to perform some maintenance and administrative tasks.

Along with this set of tasks, you can also monitor the behavior of your application to make sure it is running without problems. You can use the message logger and the user event logger to pinpoint problems or analyze user actions.

Administrating your application also includes managing some of the system settings and properties. For example, establishing and maintaining system security and data integrity are all part of managing applications.

Administrating applications involves:

- **Basic Administration Tasks**
Starting and stopping connectors, testing connections, monitoring your application and so on
- **Administrating Telco Service Manager (TSM)**
Administrating the CID, managing message and notification queues, and so on.
- **Administrating Telco Analytics Manager (TAM)**
Administrating the CBU, managing notifications, running the CID2CBU Loader, and so on.
- **Managing Distributed Architectures**
Managing session serialization

When administrating applications, you must remember that it is part of a large and complex system that works with several different systems and software. Not only do you need to be familiar with how this software works, you need to understand how your application interacts with these systems and software.

Overview of the Application Architecture

Your TSM has the following components:

- Personalization Manager
- CSS Engine
- Customer Interaction Datastore (CID)
- SmartLink (ISF)
- Customer Analytics Warehouse

Each of these components works with and communicate with different software applications. The software packages that these components work with include:

- **Web servers**

The Web server is the main entrance to MyWeb channel. The Web server hosts Web sites and their different files as well as managing the communication between Web browsers and the application server. Web servers can also host Personalization Manager files.

- **Application servers**

The application server creates and manages dynamic content of applications. It manages the communication between the Web server (and other channels) and databases. The application server serves as the foundation of the CSS Engine and can also host Personalization Manager and JSPF files.

- **Databases**

Databases contain the information your application manages. The CID contains the core application information.

- **Operational Support Systems (OSS)**

The OSS contain the information your TSM uses to enable users to manage their accounts. The information managed by **TSM** in the CID comes from OSS. SmartLink (ISF) manages the transfer of information between the CID and OSS.

About Configuration Files and their Location

During installation, there are two identical sets of application configuration files. This is done to respect the requirements of the J2EE Web Application Archive (WAR) file specifications and to help you easily deploy your application.

However, administrating may involve modifying some of the configuration files.

The configuration files are found in:

- `<home_dir>/classes/nmycfg`
This directory contains the reference set of core configuration files. These files are the default set of configuration files and are used when developing applications.
- `<home_dir>/channels/WEB-INF/classes/nmycfg`
This directory contains the Channel configuration files. These files are the configuration files that are deployed when using the J2EE WAR file.

The location of deployed Channel configuration files depends on how your application server handles and deploys WAR files. To indicate the location of deployed Channel configuration files, this manual uses `<app_dir>`.

For more information about WAR and configuration files, refer to *Installing Telco Service & Analytics Manager Applications*, *Developing Telco Service Manager (TSM)* and *Developing Telco Analytics Manager (TAM)*.

CHAPTER 2

Basic Administration Tasks

In This Section

About the Basic Tasks	26
Starting and Stopping the Synchronizer Connector	27
Starting and Stopping the OSS Connector	28
Starting and Stopping the Approval Sequencer	29
Starting and Stopping the CID2CBU Loader.....	30
Starting and Stopping the CustDim Loader.....	31
Starting and Stopping the Job Processor.....	32

About the Basic Tasks

In order to help you, this chapter covers the basic tasks of administrating your TSM.

These tasks include:

- Starting and stopping the Synchronizer connector
- Starting and stopping the OSS connector
- Starting and stopping the Approval Sequencer
- Starting and stopping the CID2CBU Loader
- Starting and Stopping the CustDim Loader
- Starting and Stopping the Job Processor

You do not start and stop your *Telco Service & Analytics Manager* application. As the application is hosted by your application server, starting and stopping your application is starting and stopping your application server.

For more information, refer to your application server's documentation.

Starting and Stopping the Synchronizer Connector

The Synchronizer connector manages the communication between the CID and OSS systems. This agent manages the requests in the CID and their transmission to the OSS.

You use a set of administration tools to start and stop the Synchronizer connector. The administration tools are:

- `agentstart`
- `agentadm`

These administration tools are located in `<home_dir>/bin`.

For more information about configuring and using connectors, refer to *Developing Connectors*.

To start the Synchronizer connector

- 1 Go to `<home_dir>/bin`.
- 2 Run `agentstart`. Use the syntax:
`agentstart <connector_name>`

The connector is loaded and it starts the processes it needs.

To stop the Synchronizer connector

- 1 Go to `<home_dir>/bin`.
- 2 Run `agentadm`. Use the syntax:
`agentadm <host> <port> shutdown`

When finished, it displays a message.

Starting and Stopping the OSS Connector

The OSS connector manages the communication between the middleware and the OSS.

You use a set of administration tools to start and stop the OSS connector. The administration tools are:

- `ossstart`
- `ossadm`

These administration tools are located in `<home_dir>/bin`.

For more information about configuring and using connectors, refer to *Developing Connectors*.

To start the OSS connector

- 1 Go to `<home_dir>/bin`.
- 2 Run `ossstart`. Use the syntax:
`ossstart <connector_name>`

The connector is loaded and it starts the processes it needs.

To stop the OSS connector

- 1 Go to `<home_dir>/bin`.
- 2 Run `ossadm`. Use the syntax:
`ossadm <host> <port> <connector> shutdown`

When finished, it displays a message.

Starting and Stopping the Approval Sequencer

The Approval Sequencer manages the approval processes in the BLM.

You use a set of administration tools to start and stop the Approval Sequencer. The administration tools are:

- `approvalsequencer`
- `approvalsequenceradm`

These administration tools are located in `<home_dir>/bin`.

These tools use the `agent.properties` configuration file to set the properties of the Approval Sequencer. This file is located in `<home_dir>/config/approvalsequencer`.

For more information about working with the Approval Sequencer, refer to *Developing Telco Service Manager (TSM)*.

To start the Approval Sequencer

- 1 Go to `<home_dir>/bin`.
- 2 Run `approvalsequencer`. The Approval Sequencer loads its configuration file and starts the processes it needs. When finished, it displays a message.

To stop the Approval Sequencer

- 1 Go to `<home_dir>/bin`.
- 2 Run `approvalsequenceradm`. Use the syntax:
`approvalsequenceradm <host> <port> shutdown`
When finished, it displays a message.

Starting and Stopping the CID2CBU Loader

The CID2CBU loader processes the notification events in the CID.

You use a set of administration tools to start and stop the CID2CBU loader. The administration tools are:

- `cid2cbuloader`
- `cid2cbuloaderadm`

These administration tools are located in `<home_dir>/bin`.

These tools use the `cid2cbuloader.properties` configuration file to set the properties of the CID2CBU loader. This file is located in `<home_dir>/config/cid2cbuloader`.

To start the CID2CBU loader

- 1 Go to `<home_dir>/bin`.
- 2 Run `cid2cbuloader`.

The CID2CBU loader loads its configuration file and starts the processes it needs. When finished, it displays a message.

To stop the CID2CBU loader

- 1 Go to `<home_dir>/bin`.
- 2 Run `cid2cbuloaderadm`. Use the syntax:
`cid2cbuloaderadm <host> <port> shutdown`

When finished, it displays a message.

Starting and Stopping the CustDim Loader

The CustDim Loader is an SmartLink (ISF) connector which loads the Customer Dimensions in the CBU.

You use a set of administration tools to start and stop the CustDim Loader. The administration tools are:

- `custdimloader`
- `custdimloaderadm`

These administration tools are located in `<home_dir>/bin`.

To start the CustDim Loader

- 1 Go to `<home_dir>/bin`.
- 2 Run `custdimloader`. Use the syntax:

```
custdimloader custdimloader
```

The connector is loaded and it starts the processes it needs.

To stop the CustDim loader

- 1 Go to `<home_dir>/bin`.
- 2 Run `custdimloaderadm`. Use the syntax:

```
custdimloaderadm <host> <port> shutdown
```

This command waits for the completion of the current input file and all of its associated processing.

Starting and Stopping the Job Processor

The Job Processor manages the processing of jobs such as reports submitted for asynchronous processing.

You use a set of administration tools to start and stop the Job Processor. The administration tools are:

- `jobprocessorstart`
- `jobprocessoradm`

These administration tools are located in `<home_dir>/bin`.

To start the Job Processor

- 1 Go to `<home_dir>/bin`.
- 2 Run `jobprocessorstart`. Use the syntax:
`jobprocessorstart <job_type_name>`
The job processor is loaded and begins processing.
For example, to run the Report Processor, use the syntax:
`jobprocessorstart ReportProcessor`

To stop the Job Processor

- 1 Go to `<home_dir>/bin`.
- 2 Run `jobprocessoradm`. Use the syntax:
`jobprocessoradm <host> <port> shutdown`
This command waits until all processing is finished before stopping the Job Processor.

CHAPTER 3

Administering Telco Service Manager (TSM)

In This Section

About Administering TSM	34
Managing the Synchronizer Connector	35
Managing Requests	38
Managing Jobs	44
Managing Persistent Action Managers	47
Managing Trouble Tickets	49
Managing SmartLink (ISF) Message Cache	50
Reloading Reference Data	53

About Administrating TSM

When managing your application, you may have to work with several different components and administration tools. For instance, you may have to monitor your TSM, your database and other software, administrate the CID, manage message and notification queues, and check logs.

This section explains the different administration tasks and the tools you use to carry them out.

Administrating involves:

- Managing the Synchronizer
- Managing requests
- Managing trouble tickets
- Managing jobs
- Managing persistent action managers
- Managing the SmartLink (ISF) message cache
- Reloading reference data
- Monitoring your application with the logger
- Auditing application activity using user events

This section does not show you how to install and configure the various administration tools. For more information on installing and configuring your tools, refer to *Installing Telco Service & Analytics Manager Applications* and the Appendixes in this manual.

Managing the Synchronizer Connector

The Synchronizer connector takes the requests from the CID and create messages to be sent to the OSS connector.

After getting a request from the CID, the Synchronizer connector translates the request into a message. This connector puts this message in a directory for the OSS connector.

Managing the Synchronizer connector involves:

- Checking the Synchronizer connector
- Purging the Synchronizer queue

Checking the Synchronizer Connector

You use the `agentadm` administration tool to manage the Synchronizer connector.

This administration tool is located in `<home_dir>/bin`.

You use this tool to:

- Check the connector status
- Check the Runtime mode
- Display the current settings

For more information about the Synchronizer connector, refer to *Developing Connectors*.

This table describes the different running modes of the Synchronizer Connector.

Synchronizer Connector Modes

MODE	DESCRIPTION
Normal	The default mode in which the connector scans the CID at a regular, fixed time intervals, extracts the requests and sends them to the middleware backbone
Paused	The connector stops scanning the CID for requests, and waits for a restart or recover command before it resumes
Recover	The connector makes a final attempt to send requests (with the status 'TransportFailed') that have not been sent due to a transport problem. When finished, it goes back to the normal mode. The connector goes into this mode when all transport queues are down

To check the status

- 1 Go to `<home_dir>/bin`.
- 2 Run `agentadm`. Use the syntax:
`agentadm <host> <port> getstatus`

The administration tool displays the current status of the connector.

To check the runtime mode of the Agent

- 1 Go to `<home_dir>/bin`.
- 2 Run `agentadm`. Use the syntax:
`agentadm <host> <port> getmode`

The administration tool displays the current runtime mode of the connector.

To display the current settings

- 1 Go to `<home_dir>/bin`.
2. Run `agentadm`. Use the syntax:
`agentadm <host> <port> info`

The administration tool displays the current settings of the connector.

Purging the Synchronizer Queue

You can empty the Synchronizer message queue when required.

Purging the Synchronizer queue also depends on your system's middleware and configuration. For instance, if your application uses MQSeries, you must use and follow the MQSeries procedure to purge queues.

By default, the Synchronizer uses a file queue for messages. The files that correspond to messages are put in a directory for processing by the OSS connector. This directory corresponds to the Synchronizer's `sync2oss` queue. To purge this queue, you delete the messages in this directory.

To purge the sync2oss queue

- 1 Go to `<home_dir>/<var>/data/sync2oss`.
- 2 Delete the message files.

If you use a middleware package to exchange messages with the OSS, refer to your product documentation for more information about purging messages.

Managing Requests

Managing requests involves:

- Viewing the pending requests
- Viewing a request detail
- Cancelling requests
- Resubmitting requests
- Purging requests

You use the administration tool to manage requests in the CID.

About Requests

When a user makes a change that impacts one of the legacy systems (such as add or remove service), the request is stored in the `REQUEST` table. The parameters of the request are stored in the `REQUEST_PARAM` table.

Requests can be:

- **Elementary**
This type of request is a single request.
- **Composite**
This type of request is a set of requests that have to be submitted for consistency.

When you change or cancel a composite request, you change all of the requests.

Pending requests are requests with one of the following statuses:

- NOT YET SUBMITTED
- SUBMISSION IN PROGRESS
- TO BE APPROVED
- TRANSPORT FAILED
- ACKNOWLEDGED
- SUBMITTED

This table describes the different request status codes and their names.

Request Status Codes

REQUEST STATUS CODE	REQUEST STATUS NAME
1	TO BE APPROVED
2	NOT YET SUBMITTED
3	SUBMITTED
4	DONE
5	FAILED
7	DENIED
8	SUBMISSION IN PROGRESS
10	TRANSPORT FAILED
12	ACKNOWLEDGED

Viewing the Requests

You can view a list of requests in the CID. You can also view the detail of a specific request.

To view the pending requests

1 Go to `<home_dir>/bin`.

2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_requests <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

- Oracle: `<instance alias>`
- DB2: `<database alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.

3 Choose 1) `View pending requests` then press Enter. The CID administration tool asks you to enter the number of days the requests have been pending.

4 Do one of the following:

- Enter 0 to display all of the pending requests
 - Enter the number of days the requests have been pending
- 5 Press Enter. The CID administration tool displays a list of pending requests.

To view a request detail

- 1 Go to `<home_dir>/bin`.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_requests <CID> <CID_ADMIN login>  
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
- 3 Choose 2) `View a request in detail` then press Enter. The CID administration tool asks you to enter the ID of the request to display.
- 4 Enter the ID of the request then press Enter. The CID administration tool displays the following pending request information:
 - Request external id
 - Generated by
 - Creation date
 - Action requested
 - Status
 - Status change date
 - Reason for failure

Cancelling Requests

You can cancel specific requests or you can cancel all of the requests. Cancelling a request means you change its status to `FAILED`. To remove the request from the CID, you need to purge the `FAILED` requests.

To cancel a request

- 1 Go to `<home_dir>/bin`.
- 2 Run the CID administration tool. Use the syntax:


```
cidAdminTool admin_requests <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

- Oracle: <instance alias>
- DB2: <database alias>
- SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.

- 3 Choose 3) Set a pending request to failed then press Enter. The CID administration tool asks you to enter the ID of the request to cancel by setting its status to FAILED.
- 4 Enter the ID of the request then press Enter.
- 5 At the prompt, press Y to confirm.

The CID administration tool cancels the request. When finished, the administration tool displays a confirmation message.

To cancel all requests

- 1 Go to <home_dir>/bin.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_requests <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

- Oracle: <instance alias>
- DB2: <database alias>
- SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.

- 3 Choose 4) Set all pending requests to failed then press Enter. The CID administration tool asks you to enter the number of days the requests have been pending that you want to set to FAILED.
- 4 Do one of the following:
 - Enter 0 to set all of the pending requests
 - Enter the number of days the requests have been pending

The CID administration tool displays a list of pending requests.
- 5 At the prompt, press Y to confirm.

The CID administration tool cancels the requests. When finished, the administration tool displays a confirmation message.

Resubmitting Requests

You can resubmit specific requests. In general, you do not need to resubmit requests. This can happen when unforeseen problems or application failures occur. For instance, if the Synchronizer connector fails and restarts, you may have requests in the CID with the status `SUBMISSION IN PROGRESS`. You have to resubmit these request by changing their status to `NOT YET SUBMITTED`.

Only requests having the `REQUEST_STATUS_CODE` of 8 (`SUBMISSION IN PROGRESS`) can be resubmitted.

To resubmit a request

- 1 Go to `<home_dir>/bin`.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_requests <CID> <CID_ADMIN login>  
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where `<CID>`:

 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
- 3 Choose 5) Reset a request being submitted to not yet submitted then press Enter. The CID administration tool asks you to enter the ID of the request to resubmit.
- 4 Enter the ID of the request then press Enter.
- 5 At the prompt, press Y to confirm.
- 6 The CID administration tool changes the request status. When finished, the administration tool displays a confirmation message.

Purging Requests

You can remove requests from the CID that are not pending requests. You can remove request having the following statuses:

- TO BE APPROVED
- DONE
- FAILED

To purge requests

1 Go to `<home_dir>/bin`.

2 Run the CID administration tool. Use the syntax:

```
cidAdminTool purge_requests <CID> <CID_ADMIN login>
<CID_ADMIN password> <days> [<status>,<status>,...] [-quiet]
[-datablocksize:####]
```

where <CID>:

- Oracle: `<instance alias>`
- DB2: `<database alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool removes all of the requests that:

- Have the specified statuses
- Are older than the specified number of days

When finished, the administration tool displays a confirmation message.

Managing Jobs

Managing jobs involves:

- Purging jobs in the CID
- Resubmitting Jobs
- Cancelling Jobs

You use the administration tool to manage jobs in the CID.

About Jobs

Jobs are tasks stored in the CID.

These tasks include reports which have been stored for asynchronous execution. When working with reports, some of the reports may be too big or may take too much time to execute. These reports are sent to the Background Report Processor as jobs to be executed asynchronously.

Purging Jobs

To purge jobs

1 Go to `<home_dir>/bin`.

2 Run the Job administration tool. Use the syntax:

```
jobAdminTool purge_jobs <CID> <CID_ADMIN login> <CID_ADMIN
password> -JobTypeCode:<job-type-code> -NbDays:<days> [-
StatusCodes:<job-status>[,<job-status>...]] [-
ServiceCode:<service-code>] [-quiet] [-DataBlockSize:#####]
```

where `<CID>`:

- Oracle: `<instance alias>`
- DB2: `<database alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The Job administration tool removes all of the jobs that:

- Have the specified statuses
- Are older than the specified number of days

When finished, the administration tool displays a confirmation message.

Resubmitting Jobs

To resubmit a job

- 1 Go to <home_dir>/bin.
- 2 Run the Job administration tool. Use the syntax:

```
jobAdminTool reset_in_progress_jobs <CID> <CID_ADMIN login>
<CID_ADMIN password> -JobTypeCode:<job-type-code> -
NbHours:<hours> [-ServiceCode:<service-code>] [-quiet] [-
DataBlockSize:####]
```

where <CID>:

- Oracle: <instance alias>
- DB2: <database alias>
- SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The Job administration tool changes the status of IN PROGRESS jobs to TO BE PROCESSED for the jobs having:

- The specified job type code
- The IN PROGRESS status for the number of hours specified
- The specified service code

When finished, the administration tool displays a confirmation message.

Cancelling Jobs

Cancelling a job involves changing its status to ERROR.

To cancel a job

- 1 Go to <home_dir>/bin.
- 2 Run the Job administration tool. Use the syntax:

```
jobAdminTool fail_in_progress_jobs <CID> <CID_ADMIN login>
<CID_ADMIN password> -JobTypeCode:<job-type-code> [-
ReasonCode:<fail-reason-code>] -NbHours:<hours> [-
ServiceCode:<service-code>] [-quiet] [-DataBlockSize:####]
```

where <CID>:

- Oracle: <instance alias>

- DB2: <database alias>
- SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The Job administration tool changes the status of `IN PROGRESS` jobs to `ERROR` for the jobs having:

- The specified job type code
- The `IN PROGRESS` status for the number of hours specified
- The specified service code

When finished, the administration tool displays a confirmation message.

Managing Persistent Action Managers

Managing persistent action managers involves:

- Purging persistent action managers from the CID

You use the administration tool to manage persistent action managers in the CID.

About Persistent Action Managers

Shopping Carts correspond to BLM Action Managers. These action managers do exactly what their name implies, they manage the actions in the current context. An action manager is considered a shopping cart when it holds a set of actions to submit at the same time. The types of action managers are:

- Action Managers
- Persistent Action Managers

The context of your application determines which type of action manager you use. In general, you use Action Managers to manage the actions in a specific context or workflow. However these action managers and their contents cannot be saved. If your application requires saving the contents of an action manager, you use Persistent Action Managers. The saved Persistent Action Managers can be backup copies of the shopping cart or used as a shopping cart template.

When saving persistent action managers as Shopping Carts and templates, you should keep in mind the following:

- There is no limit to the number of saved persistent action managers
- There is no history of saved persistent action managers

Shopping Cart templates have the following:

- Name
- Description
- Category
- Additional information to be used as criteria to retrieve it

The shopping carts are saved in the same table as the backup shopping carts. This information is in the Shopping Cart Package tables in the CID:

- `PERSISTENT_ACTIONMGR_CATEGORY` Table
- `PERSISTENT_ACTIONMGR` Table

This table describes the default Persistent Action Manager categories and their codes.

Persistent Action Manager Categories

CODE	NAME	NOTES
1	Backup	Backup category to save Persistent Action Managers during a session
2	Contract	For contract templates

Purging Persistent Action Managers

You can remove persistent action managers from the CID. When purging the persistent action managers from the CID, you specify the category of persistent action managers to purge. Make sure you specify the correct category when purging the CID.

To purge persistent action managers

1 Go to `<home_dir>/bin`.

2 Run the CID administration tool. Use the syntax:

```
cidAdminTool purge_persistent_action_managers <CID>  
<CID_ADMIN login> <CID_ADMIN password> <days> <category> [-  
quiet] [-datablocksize:####]
```

where `<CID>`:

- Oracle: `<instance alias>`
- DB2: `<database alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool removes all of the persistent action managers that:

- Are older than the specified number of days
- Have the specified category

The category corresponds to the value in the `PAM_CATEGORY_CODE` column of the `PERSISTENT_ACTIONMGR_CATEGORY` table.

When finished, the administration tool displays a confirmation message.

Managing Trouble Tickets

Managing trouble tickets involves:

- Purging trouble tickets in the CID

You use the administration tool to manage trouble tickets in the CID.

About Trouble Tickets

Trouble tickets are located in the CID and have an associated status.

Purging Trouble Tickets

You can remove the trouble tickets from the CID.

To purge trouble tickets

1 Go to `<home_dir>/bin`.

2 Run the CID administration tool. Use the syntax:

```
cidAdminTool purge_trouble_tickets <CID> <CID_ADMIN login>
<CID_ADMIN password> <days> <statuscode> <datablocksize> [-
quiet]
```

where `<CID>`:

- Oracle: `<instance alias>`
- DB2: `<database alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool removes all of the trouble tickets that:

- Have the specified statuses
- Are older than the specified number of days

When finished, the administration tool displays a confirmation message.

Managing the SmartLink (ISF) Message Cache

Managing the SmartLink (ISF) message cache involves:

- Creating the cache database structure
- Purging the message cache
- Dropping the cache database structure

You use the administration tool to manage the SmartLink (ISF) message cache.

About the SmartLink (ISF) Message Cache

You use the message cache to store SmartLink (ISF) messages while awaiting the reply from the OSS.

The message cache allows you to:

- Temporary store of integration messages and related data in order to share them among threads and processes
- Cumulatively and concurrently update stored messages for easier reconciliation

When working with messages, the connector receives a message then calls a back end API to request additional information. This API call is performed in a non blocking manner, which means the connector does not wait for the reply. Depending on your integration architecture, you may need to store the message while awaiting the reply from the backend system. As soon as the connector receives this reply from the back end, it restores the persisted message, does the required task (reconcile, update, and so on). When finished, the connector either persist the message again for another request/reply or sends the updated message then removes the persisted message.

The SmartLink (ISF) message cache has the following components:

- Message Cache database
- Message Cache database administration tool
- Message cache APIs
- Message cache processors

Working with the SmartLink (ISF) Message Cache Structure

Working with the SmartLink (ISF) Message Cache structure involves:

- Creating the structure
- Dropping the structure

To create the structure

- 1 Go to `<home_dir>/bin`.
- 2 Run the SmartLink (ISF) Message Cache administration tool. Use the syntax:


```
messagecacheAdminTool create_messagecache_structure
<InstanceName> <ADMIN login> <ADMIN password> [-
cachename:name | -sqlfilename:sql filename] [-quiet]
where <db_instance>:
```

 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

When running this tool, you specify at least one of the following:

`-cachename` the name of the cache

`-sqlfilename` the name of the file containing custom SQL script

When finished, the administration tool displays a confirmation message.

To drop the structure

- 1 Go to `<home_dir>/bin`.
- 2 Run the SmartLink (ISF) Message Cache administration tool. Use the syntax:


```
messagecacheAdminTool drop_messagecache_structure
<InstanceName> <ADMIN login> <ADMIN password> [-
cachename:name | -sqlfilename:sql filename] [-quiet]
where <InstanceName>:
```

 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

When running this tool, you specify at least one of the following:

`-cachename` the name of the cache

`-sqlfilename` the name of the file containing custom SQL script

When finished, the administration tool displays a confirmation message.

Purging the SmartLink (ISF) Message Cache

To purge messages

1 Go to `<home_dir>/bin`.

2 Run the SmartLink (ISF) Message Cache administration tool. Use the syntax:

```
messagecacheAdminTool purge_messagecache <InstanceName>  
<ADMIN login> <ADMIN password> <days> [-cachename:name | -  
sqlfilename:sql filename] [-quiet] [<datablocksize>]
```

where `<InstanceName>`:

- Oracle: `<instance alias>`
- DB2: `<database alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

When running this tool, you specify at least one of the following:

`-cachename` the name of the cache

`-sqlfilename` the name of the file containing custom SQL script

When finished, the administration tool displays a message.

The SmartLink (ISF) Message Cache administration tool removes all of the messages that are older than the specified number of days

When finished, the administration tool displays a confirmation message.

Reloading Reference Data

In general, your reference data should not change frequently. But on occasion, you may need to change this data. You can use the reference data reload feature to program your application to reload reference data without having to take your application off line.

You cannot use this feature to remove reference data and it cannot be used to add new types of reference data. This feature is for reloading modified and new reference information only. If you modify the data structure or remove reference information, you must stop and restart your application server.

Reloading reference data involves:

- Specifying the caching policy of BLM objects
- Creating the batch files to reload the data
- Programming JSPs to use this feature

About the BLM Cache

The BLM cache is a global cache for all user sessions. All of the cached BLM objects are one of following types:

- `GLOBAL`
The object is cached the entire life of the BLM host process.
- `RELOADABLE`
The object is cached and can be updated using the reference data reload feature.
- `HTTP_REQUEST`
The object is cached and updated for each HTTP request.

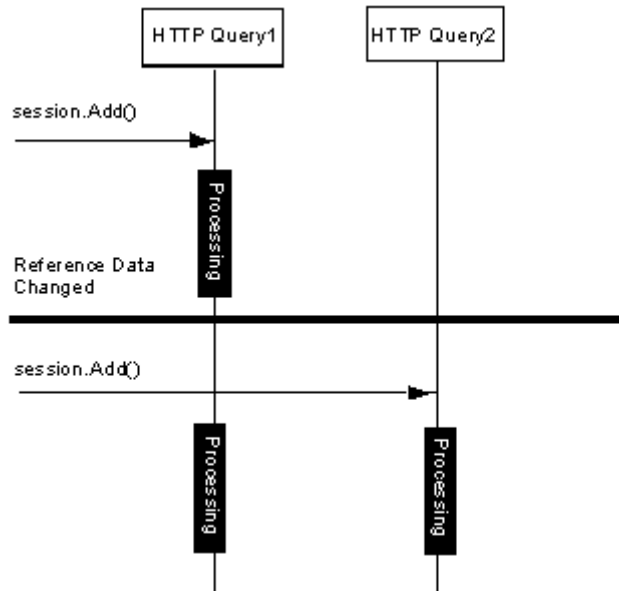
The `policy.properties` configuration file contains the list of BLM objects and their assigned type. This file is located in `<home_dir>/classes/nmycfg/blm/util`.

The `VERSIONS` table contains information about the reference data. This table contains the version of the reference data and its associated timestamp.

Every time a JSP calls the `session.Add()` method, the BLM checks the `VERSIONS` table to see if the timestamp has changed. If the reference data has changed, the BLM updates all of the `RELOADABLE` objects in the cache and starts the session.

Limits to Using the Reference Data Reload Feature

Concurrent user sessions may cause some problems and you should design your application to take the following sequence of events:



In this sequence of events, the HTTP Query2 causes your TSM to update the BLM cache because the reference data changed. In this case, the HTTP Query 1 cannot be notified and the BLM cache might not contain data needed by the query. This is especially important if your data reload contains modified reference information. This situation does not cause problems when you reload only new reference information, because the threads do not need to know about new information.

In order to avoid this situation, you can:

- Use this feature only to load modified or new information in existing reference tables
 - Block access to the impacted service during reloads (redirect to another JSP, display a message)
 - Limit the frequency of reloads
 - Reload your data during off-peak hours
- 1 Create a batch file to extract and load information into the CID.
 - 2 At the end of the transaction, do the following in the `VERSIONS` table:
 - Insert your reference data version reference
 - Insert the timestamp using the `SYSDATE` method

```
update versions set ITEM_VERSION="your reference" where  
ITEM_CODE='REFERENCE_DATA'
```

```
update versions set ITEM_TIMESTAMP=sysdate where  
ITEM_CODE='REFERENCE_DATA'
```

Do not insert or change the information in the `STRUCTURE` data. This data is for internal use only and must not be modified.

To program a JSP using the reference data reload feature

- 1 Open and validate the session
- 2 Use the `ObjectRefMgr.getReferenceDataTimestamp()` method to return the time stamp of the current reference data.
- 3 Store the timestamp in the `HTTP` session.
- 4 When the thread handles an `HTTP` request within the current session, retrieve the timestamp and compare it with the stored timestamp.
 - If the timestamps differ, the reference data has changed and the BLM cache has been updated. You can change the workflow (display a message, reset the workflow, and so on).
 - If the timestamps are identical, the application continues normally.

For more information about programming JSPs, refer to *Developing User Interfaces*.

CHAPTER 4

Administering Telco Analytics Manager (TAM)

In This Section

About Administering TAM	58
Managing Notifications	59
Managing the CustDim Loader	69
Managing Saved Reports.....	75

About Administrating TAM

TAM is designed to run with a minimum amount of user intervention or maintenance.

To ensure that your system runs smoothly and meets the availability and performance demands your users expect, you have to perform some maintenance and administrative tasks. Along with this set of tasks, you can also monitor the behavior of your application to make sure it is running without problems. You can use the message logger to pinpoint problems.

Administrating TAM involves:

- **Managing Notifications**
Some changes in the CID require the CBU to be updated. When this happens, a notification event occurs. You need to make sure that these events are handled correctly by the CID2CBU loader.
- **Working with the CID2CBU loader**
The CID2CBU loader handles the update of information in the CBU with information in the CID.
- **Working with the CustDim Loader**
The information in the CBU contains information from various sources. You use the CustDim loader to load information into the CBU either the first time you set it up or periodically as required.
- **Managing Saved Reports**
Your TAM may use the Report Manager to allow users to save reports. If it does, you need to be able to manage these reports. You need to be able to remove files from directories and purge files.

Managing Notifications

Managing notifications involves:

- Viewing pending notifications
- Viewing a notification detail
- Cancelling notifications
- Resubmitting notifications
- Purging notifications
- Managing synchronization of Organizations

You use the administration tools to manage the notifications.

About Managing Notifications

Changes to specific objects generate notification events. A notification event means that the information in the CID has changed and that the data in the CBU must be updated.

These events are stored in the CID `NOTIFICATION` table. This table serves as the notification queue. This queue is polled at specified intervals by the CID2CBU loader and processes the entries in FIFO (First in/First Out).

Managing notifications involves:

- Viewing pending notifications
- Viewing a notification detail
- Cancelling notifications
- Resubmitting notifications
- Purging notifications
- Managing synchronization of Organizations

TAM comes with administration tools to help you carry out these tasks. For more information about the administration tools and their syntax, refer to the *Administration Tool Reference* chapter in this manual.

Notification Status Codes

This table describes the different notification status codes and their names.

Notification Status Codes

STATUS CODE	NAME
2	NOT YET NOTIFIED
4	NOTIFICATION DONE
5	NOTIFICATION FAILED
8	NOTIFICATION IN PROGRESS

Viewing Notifications

You can view a list of the failed and in progress notifications in the CID. You can also view the detail of a specific notification.

To view failed notifications

- 1 Go to `<home_dir>/bin`.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>  
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```


where `<CID>`:
 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.

- 3 Choose `1) View failed notifications` then press Enter. The CID administration tool asks you to enter the number of days the notifications have been set to FAILED.
- 4 Do one of the following:
 - Enter `0` to display all of the pending notifications
 - Enter the number of days the notifications have been set to failed
- 5 Press Enter. The CID administration tool displays a list of failed notifications.

To view in progress notifications

- 1 Go to <home_dir>/bin.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

 - Oracle: <instance alias>
 - DB2: <database alias>
 - SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
- 3 Choose 2) List notifications 'in progress' then press Enter. The CID administration tool asks you to enter the number of days the notifications have been IN PROGRESS.
- 4 Do one of the following:
 - Enter 0 to display all of the notifications in progress
 - Enter the number of days the notifications have been set to IN PROGRESS
- 5 Press Enter. The CID administration tool displays a list of notifications.

To view a notification detail

- 1 Go to <home_dir>/bin.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

 - Oracle: <instance alias>
 - DB2: <database alias>
 - SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
- 3 Choose 3) View Notification detail then press Enter. The CID administration tool asks you to enter the ID of the notification to display.
- 4 Enter the ID of the notification then press Enter. The CID administration tool displays the following information:
 - Notification ID

- Creation date
- Modification date
- Type
- Notified object type
- First name
- Last name/Business:
- Status

Cancelling Notifications

You can cancel specific notifications or you can cancel all of the notifications. Cancelling a notification means you change its status to failed. To remove the notification from the CID, you need to purge the failed notifications.

To cancel a notification

- 1 Go to `<home_dir>/bin`.
 - 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>  
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
 - 3 Choose 5) Set a notification 'in progress' to 'failed' then press Enter. The CID administration tool asks you to enter the ID of the notification to cancel by setting its status to failed.
 - 4 Enter the ID of the notification then press Enter.
 - 5 At the prompt, press Y to confirm.
- The CID administration tool cancels the notification. When finished, the administration tool displays a confirmation message.

To cancel all notifications

- 1 Go to `<home_dir>/bin`.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

- Oracle: <instance alias>
- DB2: <database alias>
- SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.

- 3 Choose 7) Set all notifications 'in progress' requests to 'failed' then press Enter. The CID administration tool asks you to enter the number of days the notifications have been in progress that you want to set to FAILED.

- 4 Do one of the following:

- Enter 0 to process all of the notifications in progress
- Enter the number of days the requests have been set to in progress

The CID administration tool displays a list of notifications to cancel.

- 5 At the prompt, press Y to confirm.

The CID administration tool cancels the notifications. When finished, the administration tool displays a confirmation message.

Purging Notifications

You can remove notifications from the CID. You can purge notifications that have the following statuses:

- FAILED
- DONE

To purge notifications

- 1 Go to <home_dir>/bin.

- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool purge_notifications <CID> <CID_ADMIN login>
<CID_ADMIN password> <days> [<status>,<status>,...] [-quiet]
[-datablocksize:####]
```

where <CID>:

- Oracle: <instance alias>
- DB2: <database alias>

- SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool removes all of the notifications that:

- Have the specified statuses
- Are older than the specified number of days

If no statuses are specified, the tool removes `FAILED` and `DONE` notifications.

When finished, the administration tool displays a confirmation message.

Resubmitting Notifications

You can resubmit specific notifications. In general, you do not need to resubmit notifications. However, unforeseen problems or application failures do occur. When this happens, you may need to resubmit some notifications. For instance, if the connection to the CBU is lost, you may have several notifications that are in progress. This means that the notifications have been read but not yet processed.

To resubmit a notification

- 1 Go to <home_dir>/bin.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>  
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

 - Oracle: <instance alias>
 - DB2: <database alias>
 - SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
- 3 Choose 4) Set a notification 'in progress' to 'not yet notified' then press Enter. The CID administration tool asks you to enter the ID of the notification to resubmit by setting its status to not yet notified.
- 4 Enter the ID of the notification then press Enter.
- 5 At the prompt, press Y to confirm.

The CID administration tool resubmits the notification. When finished, the administration tool displays a confirmation message.

To resubmit all notifications

- 1 Go to <home_dir>/bin.

2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

- Oracle: <instance alias>
- DB2: <database alias>
- SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.

3 Choose 6) Set all notifications 'in progress' to 'not yet notified' then press Enter. The CID administration tool asks you to enter the number of days the notifications have been in progress that you want to resubmit by setting the status to not yet notified.

4 Do one of the following:

- Enter 0 to process all of the notifications in progress
- Enter the number of days the notifications have been in progress

5 At the prompt, press Y to confirm.

The CID administration tool resubmits the notifications. When finished, the administration tool displays a confirmation message.

Managing Organization Notifications

You can manage the synchronization of organization data in the CID and CBU. In general, you do not need to manage the synchronization of organization data. However, if an error occurs when processing notifications, you may need to force the resynchronization of some data. For instance, you may have a create organization notification in the queue along with a modify organization notification that adds a contract to this organization. If the create organization notification fails, then the modify organization adding a contract to this organization is meaningless because the organization does not yet exist in the CBU.

An organization has two flags that allow you to manage the notification and synchronization of changes to organization data in the CID. The flags are:

- `NotificationActivated`
Specifies whether or not changes to the organization generate notification events for the CBU. This option is deactivated by default.
- `NotificationRequested`
Specifies whether or not a notification event has already been generated for changes to this organization and placed in the notification queue. When true, a notification has been generated and placed in the notification queue. When the CBU has been successfully synchronized with the latest organization change, this flag is reset to false.

You use the `notifyorg` tool to force the synchronization of organization data in the CID and CBU.

This tool does the following:

- Generates a notification event for each organization with a `NotificationRequested` flag set to 1.
- For each organization with a `NotificationRequested` flag set to 1, calls the notification logic to determine if an update organization event must be generated.

For more information about the default notification logic and how to write your own, refer to *Developing Telco Analytics Manager (TAM) - Customizing Notifications*.

Managing the synchronization for organizations involves:

- Resetting organization `NotificationRequested` flags
- Running the `notifyorg` tool to force synchronization of organization data

Before running the `notifyorg` tool to force synchronization, you need to make sure that there are no failed notifications in the queue.

To force the synchronization of an organization

- 1 Go to `<home_dir>/bin`.
 - 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
 - 3 Choose 8) Force synchronization of an organization then press Enter. The CID administration tool asks you to enter the ID of the notification. The NotificationRequested flag of the organization of this notification will be set to true.
 - 4 Enter the ID of the notification then press Enter.
 - 5 At the prompt, press Y to confirm.
- The CID administration tool modifies the organization's flag. When finished, the administration tool displays a confirmation message.

To force the synchronization of all organizations

- 1 Go to `<home_dir>/bin`.
- 2 Run the CID administration tool. Use the syntax:

```
cidAdminTool admin_notifications <CID> <CID_ADMIN login>
<CID_ADMIN password> [-quiet] [-datablocksize:####]
```

where <CID>:

 - Oracle: `<instance alias>`
 - DB2: `<database alias>`
 - SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool displays a menu.
- 3 Do one of the following:
 - Choose 9) Force synchronization of all organizations with 'failed' notifications
 - Choose 10) Force synchronization of all organization with notifications 'in progress'

- 4 Press Enter. The CID administration tool asks you to enter the number of days the notifications have been set to their current status.
- 5 Do one of the following:
 - Enter 0 to process all of the notifications
 - Enter the number of days the notifications have been set to their current status
- 6 At the prompt, press Y to confirm.

The CID administration tool modifies the organization's flag. When finished, the administration tool displays a confirmation message.

Before you run the `notifyorg` administration tool, make sure you purge all previous notifications with a `failed` or `in progress` statuses.

To synchronize organization information in the CBU

- 1 Go to `<home_dir>/bin`.
- 2 Run the `notifyorg` administration tool. Use the syntax:

```
notifyorg [-datablocksize:####]
```

The `notifyorg` administration tool synchronizes the organization information in the CBU. The organizations in the CID with active Notification Requested flags are updated in the CBU. When finished, the tool displays a summary of the organizations it processed.

Managing the CustDim Loader

Managing the CustDim Loader involves:

- Starting the CustDimLoader
- Stopping the CustDim Loader
- Troubleshooting

You use the administration tools to manage the CustDim Loader.

About Working with the CustDim Loader

The CustDim Loader is an SmartLink (ISF) connector.

For more information about working with the SmartLink (ISF) and its connectors, refer to *Developing Connectors*.

Running the CustDim Loader

To start the CustDim Loader

- 1 Go to `<home_dir>/bin`.
- 2 Run `custdimloader`. Use the syntax:

```
custdimloader custdimloader
```

The connector is loaded and it starts the processes it needs.

To pause the CustDim Loader

- 1 Go to `<home_dir>/bin`.
- 2 Run `custdimloaderadm`. Use the syntax:

```
custdimloaderadm <host> <port> pause
```

This command forces the connector to pause the extraction of new messages from all its inbound queues (invoice file for the splitter, message for the transformer and message for the CID loader and CBU loader)

- 3 Run `custdimloaderadm` to resume processing. Use the syntax:

```
custdimloaderadm <host> <port> resume
```

This command forces the connector to resume the extraction of new messages from all its inbound queues.

To stop the CustDim loader

- 1** Go to <home_dir>/bin.
- 2** Run `custdimloaderadm`. Use the syntax:

```
custdimloaderadm <host> <port> shutdown
```

This command waits for the completion of the current input file and all of its associated processing.

Monitoring the CustDim Loader

In order to help you pinpoint problems, the CustDim loader logs events in the configured log output.

Statistical log entries occur:

- When the connector ends its initialization
- When the connector starts processing invoice files
- When the connector loads customer dimensions in the CID
- When the connector loads customer dimensions in the CBU
- When the connector shuts down because of no more data to process or because a shutdown command has been received
- At specified periodic intervals configured in the connector startup command file

They contain statistical information about the processing of the connector, including:

- Number of invoices files to process
- Number of invoices generated per invoice file
- Number of invoices successfully inserted in the CID
- Number of invoices successfully inserted in the CBU

For more information about logs, refer to *Monitoring Telco Service & Analytics Manager Applications* in this manual.

CustDim Loader Statistics

LOG ENTRY	DESCRIPTION
[list entries processed = x]	Total number of entries in lists already processed or currently being processed
[files to split = x]	Total number of input files sent to splitter
[split files = x]	Total number of input files successfully split
[cut out invoices = x]	Total number of cut out invoices
[transformed invoices = x]	Total number of successfully transformed invoices
[accounts in CID = x]	Total number of invoices successfully inserted in the CID
[accounts in CBU = x]	Total number of invoices successfully inserted in the CBU

CID Statistics

LOG ENTRY	DESCRIPTION
[organizations = x]	Total number of organizations inserted in the CID
[levels = x]	Total number of levels inserted in the CID
[members = x]	Total number of members inserted in the CID
[billing accounts = x]	Total number of billing accounts inserted in the CID
[contracts = x]	Total number of contracts inserted in the CID

CBU Statistics

LOG ENTRY	DESCRIPTION
[organizations = x]	Total number of organizations inserted in the CBU
[levels = x]	Total number of levels inserted in the CBU
[members = x]	Total number of members inserted in the CBU
[billing accounts = x]	Total number of billing accounts inserted in the CBU
[contracts = x]	Total number of contracts inserted in the CBU

Recovering from Errors

Handling Connector Process Crashes

When running the CustDim Loader, your connector process may crash unexpectedly. When this happens, the current state of the connector is lost and the current processing of an invoice flow is interrupted.

In order to recover from a crash, no specific action is required. Restart your connector and the processing of your invoice flow will start again.

Remember that recovering from a connector process does not damage or corrupt the CBU or CID databases. This means that running the invoice load in CID and CBU is not a problem.

Handling Splitting Errors

When running the CustDim Loader, the invoice file being processed may contain corrupted data. When this happens, errors may occur during the splitting phase.

When such errors occur, the CustDim Loader generates a message in the splitting error queue. This message identifies the file which generated the error.

When you are finished fixing the error, you must reprocess the file.

Handling Transformation Errors

When running the CustDim Loader, the transformation of the invoice format may generate some errors. When this happens, the legacy invoice message is put in the transformation error queue.

You can do the following:

- Correct the problem in the invoice message and resubmit it for transformation by copying the message into the transformation retry queue
- Get a new copy of the invoice file and retry the whole invoice flow

Handling CID Update Errors

When running the CustDim Loader, errors may occur when updating the CID. When this happens, the invoice message is put in the CID Update error queue.

You can do the following:

- Correct the problem in the invoice message and resubmit it for processing by copying the message into the CID Update retry queue
- Get a new invoice file containing the corrected invoice and retry the whole invoice flow

Handling CBU Update Errors

When running the CustDim Loader, errors may occur when updating the CBU. When this happens, the invoice message is put in the CBU Update error queue.

You can do the following:

- Correct the problem in the invoice message and resubmit it for processing by copying the message into the CBU Update retry queue
- Get a new invoice file containing the corrected invoice and retry the whole invoice flow

Managing Saved Reports

About Managing Reports

When creating and saving reports, the Report Manager creates temporary files. These files are:

- `.part` files

These temporary files correspond to report files that are being created.

- `.lock` files

These temporary files are used to control concurrent use of a specific file.

Under normal circumstances, these temporary files are automatically managed and purged. However, you may need to purge these temporary files.

Managing reports involves:

- Purging reports

Purging Reports

Purging reports involves:

- Purging corrupted reports
- Purging all reports
- Purging the cache

To purge corrupted reports

1 Go to `<home_dir>/bin`.

2 Run the Report Manager administration tool. Use the syntax:

```
savedreportAdminTool purgeCorrupted <repositoryCode>
```

where `<repositoryCode>` is the directory declared in the Report Manager configuration file.

The Report Manager administration tool removes all of the report files that:

- Are older than the specified validity period
- Are older than the specified `timetolive`
- Are temporary files which have not been validated (with a `.part` extension)

When finished, the administration tool displays a confirmation message.

To purge reports

1 Go to `<home_dir>/bin`.

2 Run the Report Manager administration tool. Use the syntax:

```
savedreportAdminTool purgeAllReports <repositoryCode>
[<keyCode>]
```

where:

`<repositoryCode>` is the directory declared in the Report Manager configuration file.

`<keyCode>` is the optional key code corresponding to a specific directory/user

The Report Manager administration tool removes all of the report files and removes the directory.

When finished, the administration tool displays a confirmation message.

To purge the cache

1 Go to `<home_dir>/bin`.

2 Run the Report Manager administration tool. Use the syntax:

```
savedreportAdminTool purgeCache <repositoryCode>
```

where:

`<repositoryCode>` is the directory declared in the Report Manager configuration file.

The Report Manager administration tool removes all files and removes the directory.

When finished, the administration tool displays a confirmation message.

CHAPTER 5

Monitoring Telco Service & Analytics Manager Applications

In This Section

About Monitoring Applications using Logs	78
Logger Events	80
Event Types	81
Severity Levels	82
Event Modules	83
Overview of Using Logs	84
Configuring the Logger.....	87
Examples of Logger Files.....	103
Using Logs with a Supervision Platform.....	105
Filtering by Source Component.....	108

About Monitoring Applications using Logs

You can use the system logger to create logs while your application is running. These logs are very helpful when you have to monitor system activity and are helpful in solving problems and pinpointing system or architecture bottlenecks.

The common logger features are available for the different product components and each component can have its own specific logger configuration and output. You can log events that occur in the:

- Presentation Layer - includes the Personalization Manager and CSS Engine
- Synchronizer - The Synchronizer connector, SmartLink (ISF), CSS Engine
- Approval Sequencer
- CID2CBU loader
- Job Processor
- CustDim Loader
- NotifyOrg
- Connectors - Loopback and Template Connectors
- QRA
- WFS

Using logs to monitor your application involves the following:

- The capability to define meaningful alerts based on logs the application generates.
- The assumption that your application works fine until an alert has been raised.
- Your application does not only deal with core components.

This also means that you use a supervision platform or tool to gather information and generate alerts. The enhanced system logger helps you make your application an integral part of a comprehensive monitoring system.

There are also different professionals who supervise applications and who can use these logs to monitor applications:

- **Developers**

They can define what should be logged and monitored. They also determine the type of alert to generate and define possible corrective actions.

- **Administrators**

They are responsible for monitoring the application and define and integrate system logs and the monitoring system.

For more information about the standard log messages and their meaning, refer to the *Logger Message Reference Documentation*.

Logger Events

ATTRIBUTE	DESCRIPTION
Date/Time	ISO-8601 date string to identify the time
Thread id	To identify the thread that hosts the event generator
Unique id	To identify the event
Session id	To identify the user session
Type	To classify the impact of event
Severity	To classify the issue level
Module	To identify the event technical source
Code	To classify and describe what occurs
Description	To classify and describe what occurs – linked to code
Debug information	To provide additional technical information

Event Types

EVENT TYPE	DESCRIPTION
INIT	Covers application/module initialization processes
STATE	Covers application/module state changes
EXCEPTION	Covers internal exceptions
SESSION	Covers user session life cycle
REQUEST	Covers request life cycles
MESSAGE	Covers lifecycle in the integration framework
OBJECT	Covers object handling
RESOURCE	Covers component events
DATA	Covers customer data handling in the CID
NONE	Unclassified events

Severity Levels

SEVERITY LEVEL	DESCRIPTION
FATAL	Events that have an impact on the availability of the application
ERROR	Events that cause a given application workflow to not work properly
WARN	Events that may impact the behavior of the application
INFO	Events that record successful basic system actions for supervision
DEBUG	Level 0: no debug information. Level 3: for activation in a production environment within a working day. Level 5: for activation in a production environment for a limited period. Level 7: for activation in a very limited way with one or two concurrent users.

Event Modules

EVENT MODULE	DESCRIPTION
AGT	Any agent – synchronizer, connector, sequencer
BLM	Business Logic Manager
DAL	Data Access Layer
SmartLink (ISF)	SmartLink (ISF) Framework and all of its sub modules
JSPF	JSP Framework
JSP	Java Server Pages
LOG	The logger platform
NIL	No module involved
QRA	Query, Reporting, and Analysis Engine
UTL	Internal utility components
WFS	Web File System

Event Code/description:

- Events are coded to ensure the description, for a given case, is always the same whatever the event source is.
- A code is associated with a description, which is the actual event message.
- The code is the reference to detect what occurs.

DEBUG LEVEL	NOTES
0	Minimum debug information
3	Events with debug level 3
5	Events with debug level 3 or 5
7	Events with debug level 3, 5, or 7

Overview of Using Logs

When using logs to monitor your application, you have to decide on the following:

- The goal for logs
- Which components to monitor
- The need for dynamic configuration
- The information to generate
- The output format

Determining the Goals

You have to determine what exactly you want to do with the logs. Once you decide this, you can easily determine what needs to be done in order to configure and set up the logger.

- If you work with a supervision platform that monitors a text-based source, you should use the circular file (rolling file) driver. We recommend this because even though there are backups, logs are always generated into the main output, with a constant file name.
- If you want to backup and route the generated logs regularly, you should use the time stamped file (daily file) driver. There are several backup capabilities.
- If you want to check what occurs in the application, you should use the standard output.

Determining the Components to Monitor

You can get events from several product layers:

- Presentation Layer
This layer covers modules that run in the application server.
- Integration Layer
- System components

The components to monitor depend on your application features and architecture. For instance, if your application does not use the order validation feature, you do not use the Approval sequencer. Therefore you do not need to worry about monitoring its logs. The same is true with connectors. If you do not synchronize your backend systems with a connector, there is no need to work with connector logs.

Determining the Need for Dynamic Configuration

You can configure the logger dynamically. This means you can change the logger configuration without restarting the component. Using this feature involves a thread that reloads the logger configuration regularly.

You may need to use this feature in the following environments:

- Production environment

In this environment, activating the dynamic reload is trade-off between the need to increase the amount of logged information when an issue occurs and having an additional thread running. This also depends on the supervision process and if it covers the capability to change the logger configuration and get more information.

- Application acceptance test / Pre-integration environment

In these environments, dynamic configuration is helpful to speed up tasks. As issues may occur frequently, using this feature helps get the additional information you need to pinpoint a problem and find a solution quickly.

- Development

When developing, dynamic configuration is helpful to continuously adapt the log to the components being worked on or tested.

Default Settings:

- The presentation layer logger does not use dynamic configuration.
- Standalone components (synchronizer, connector, approval sequencer) use dynamic configuration with a configuration reload every 5 minutes.

Determining the Level of Information

In order to obtain information you can use, you need to determine the type of information to log.

You use the following event attributes to make sure the logs contain the information you need:

- **Event Type**

Basically, you could decide to log all event types, but it really depends on your supervision process.

For instance, you may not want to focus on init/stop phases. In that case, there is no need to get 'INIT'-types events.

- **Event Severity**

The logger configuration offers capability to define the minimum severity level you want to log.

If you select 'ERROR', the log only contains events with severity 'ERROR' or 'FATAL'.

For instance, you log for a monitoring platform and are trying to pinpoint some problems. In this case, you set the minimum severity level to 'WARN'.

Determining the Output Format

The format of your output depends of the way you plan on using the logs. For instance, you most likely use different output formats when working with a system console and a supervision platform.

When determining your format, keep in mind:

- The logger generates one line per event
- You can configure the pattern code
- You can arrange attributes the way you want and you can use any allowed character separator
- You can decide to log all attributes or not.

Configuring the Logger

You can customize the logger for each component. For example, your presentation layer logger can log only logins and critical errors and your Synchronizer logger tracks debug information.

For each component, you use the following configuration files:

- `logger.properties` to set the basic logger properties
- `log4j.properties` to set the event types, appenders, and severity

Although each component has its own set of configuration files, you configure the components the same way. The instructions in this section apply to configuring the logger for all components.

Logger Configuration Files

The location of the configuration files:

COMPONENT	PATH
Presentation Layer	<home_dir>/classes/nmycfg/util <home_dir>/<channels>/WEB-INF/classes/nmycfg/util
Synchronizer	<home_dir>/config/synchronizers/synchronizer/util
Approval Sequencer	<home_dir>/config/approvalsequencer/util
Connector Template	<home_dir>/config/connectors/loopback/nmycfg/util
Loopback Connector	<home_dir>/config/connectors/connectortemplate/nmycfg/util
Job Processor	<home_dir>/config/jobprocessors/reportprocessor/nmycfg/util
CID2CBU Loader	<home_dir>/config/cid2cbuloader/nmycfg/util
CustDim Loader	<home_dir>/config/custdimloaders/custdimloader/nmycfg/util
Job Services	<home_dir>/config/jobservice/nmycfg/util
WFS	<home_dir>/config/reportmanager/nmycfg/util
Job Admin Tool	<home_dir>/lib/admin/jobs/nmycfg/util
NotifyOrg	<home_dir>/config/notifyorg/nmycfg/util

Setting the Basic Properties

You use the `logger.properties` configuration file to set the following basic properties of each logger:

- Location of the dynamic configuration file
Path of the `log4j.properties` configuration file that contains information on the events to log and their format.
- Frequency to reload the parameter file
The time in seconds to reload the `log4j.properties` configuration file.
- Debug Levels
The default debug level for the event types declared in the `log4j.properties` configuration file. These setting can also be overridden by even type as required.

To set the path of the parameter file

- 1 Go to the `/util` directory containing the `logger.properties` configuration file.
- 2 Open `logger.properties`.
- 3 For the `logger.log4j.properties` setting, enter the name of the `log4j.properties` configuration file.
- 4 Save your changes.

The two files must be in the same path.

The `log4j.properties` configuration file is found using the `CLASSPATH`.

To set the reload frequency

- 1 Go to the `/util` directory containing the `logger.properties` configuration file.
- 2 Open `logger.properties`.
- 3 Change the `logger.log4j.reloadDelay` setting to one of the following:
 - The number of seconds to wait before reloading the `log4j.properties` configuration file
 - 0 to deactivate reloading
- 4 Save your changes.

By default, the logger configuration disables the dynamic reloading of the configuration for Presentation Layer.

This is due to J2EE specifications that recommend not creating threads, except when explicitly required.

To specify the debug level

- 1 Go to the `/util` directory containing the `logger.properties` configuration file.
- 2 Open `logger.properties`.
- 3 Under `Default DEBUG Level`, enter default debug level for event types. Use they syntax:
`debug.level=<DEBUG level>`
- 4 Save your changes.

The `logger.log4j.reloadDelay` setting determines when your changes will be taken into account. This setting specifies when the logger reloads its configuration files. If you have disabled the reloading of configuration files, you need to reactivate it.

To specify the debug level by event type

- 1 Go to the `/util` directory containing the `logger.properties` configuration file.
- 2 Open `logger.properties`.
- 3 Under ***DEBUG level by event type***, find the event type you want to set the debug level for.
- 4 Remove the `#` comment marker in front of the event type.
- 5 Set the debug level for the event type. This setting overrides the default debug level declared for event types. Use the syntax:
`debug.level.<event type code>=<DEBUG level>`
where
`<event type code>` corresponds to the event types declared in the `log4j.properties` file.
- 6 Save your changes.

The `logger.log4j.reloadDelay` setting determines when your changes will be taken into account. This setting specifies when the logger reloads its configuration files. If you have disabled the reloading of configuration files, you need to reactivate it.

Specifying the Events to Log

You use the `log4j.properties` configuration file to set the types of events to log.

By specifying different types of events to log, the logs contain only information about the different types of events you want to track.

For each event type, you specify:

- The minimum severity level
- Debug level when required
- The target for the generated events (output)

To specify the types of events to log

- 1 Go to the `/util` directory containing the `log4j.properties` configuration file.
- 2 Open `log4j.properties`.
- 3 Under `EVENTS`, enter the event to log. Use the syntax:

<Event Type> = <Severity Level> <Output Driver #1>, <Output Driver #2>, ...

<Event Type> is one of the following event type patterns:

EVENT TYPE	DESCRIPTION
All types	log4j.category.nmy
INIT	log4j.category.nmy.INIT
STATE	log4j.category.nmy.STATE
EXCEPTION	log4j.category.nmy.EXCEPTION
SESSION	log4j.category.nmy.SESSION
REQUEST	log4j.category.nmy.REQUEST
MESSAGE	log4j.category.nmy.MESSAGE
OBJECT	log4j.category.nmy.OBJECT
RESOURCE	log4j.category.nmy.RESOURCE
DATA	log4j.category.nmy.DATA
NONE	log4j.category.nmy.NONE

<Severity Level> is the minimum security level you want the event type to be logged with:

SEVERITY LEVEL	LEVELS OF EVENTS LOGGED
FATAL	FATAL
ERROR	FATAL, ERROR
WARN	FATAL, ERROR, WARN
INFO	FATAL, ERROR, WARN, INFO
DEBUG	FATAL, ERROR, WARN, INFO, DEBUG

<Output Driver> is the output driver to use:

DRIVER	CODE
Standard output driver	CON

Rolling file output driver	ROL
Daily file output driver	DAY

You enter as many lines in the `EVENTS` section as there are event types to track.
 To disable an event type, enter a line `<Event Type>=INFO, DISABLED`

Configuring the Logger Output

You use the `log4j.properties` configuration file to set the properties of the output drivers.

For each output driver, you specify:

- Target of the driver
- Driver specific properties
- Format of the log

About the Standard Output Driver

This output driver sends log information to your application's standard output (stdout.)
 This log information is included in all other standard application output.

To configure a standard output driver

- 1 Go to the `/util` directory containing the `log4j.properties` configuration file.
- 2 Open `log4j.properties`.
- 3 Go to the `#STANDARD OUTPUT DRIVER SETTINGS` section.
- 4 Set `log4j.appender.CON.Target` to one of the following:
 - `System.out` to redirect output to the standard output
 - `System.error` to redirect output to the error output.
- 5 Set the `log4j.appender.CON.layout.ConversionPattern` to specify the log format.
- 6 Save your changes.

```
log4j.appender.CON=org.apache.log4j.ConsoleAppender
log4j.appender.CON.Target=System.out
log4j.appender.CON.layout=com.netonomy.util.logger.LoggerLayout
log4j.appender.CON.layout.ConversionPattern=%{DATE_TIME}; {EVENT_ID}; {SEVERITY}; {MODULE}; {THREAD_ID}; {EVENT_TYPE}
}; {ERROR_CODE}; {DESCRIPTION}
```

About the Rolling File Output Driver

This output driver saves log information in a text file.

When using this driver, you specify the following:

- The name and location of the log file
- Maximum size of the text file.

When this limit is reached, the logger saves the file as `<log_filename>.1` and starts a new log file. When there is more than one saved logger file, the logger increments the filename of the saved log files. This means that the `<log_filename>.1` is the latest saved log file.

- Number of backup copies.

Determines the number of backup copies the logger keeps. When this limit is reached, the logger deletes the oldest backup copy.

Make sure you specify the correct file size and the number of backup copies.

If you do not, you may lose some log information as the logger automatically deletes the oldest log files.

To configure a rolling file output driver

- 1 Go to the `/util` directory containing the `log4j.properties` configuration file.
- 2 Open `log4j.properties`.
- 3 Go to the `#ROLLING FILE OUTPUT DRIVER SETTINGS` section.
- 4 Modify the following settings:

SETTING	DESCRIPTION
<code>log4j.appender.ROL.File</code>	Full path and filename of the log file
<code>log4j.appender.ROL.MaxFileSize</code>	<p>Maximum size of the log file.</p> <p>Use the syntax: <code><size><unit></code></p> <p>Units can be either:</p> <p>KB MB GB</p> <p>For a log file with a maximum size of 4MB, enter:</p> <p><code>log4j.appender.ROL.MaxFileSize=4MB</code></p>

- 1 Save your changes.

Saved copies are named `<log_filename>.<number>` and incremented when the logger creates a new copy.

This means that `<log_filename>.1` is the latest saved log file.

Default Rolling File Output Settings

By default, the Rolling file output driver has the following configuration:

- MaxFileSize: 4 MB
- MaxBackupIndex: 5
- File: depends on the product layer

COMPONENT	ROLLING FILE DRIVER OUTPUT FILE DEFAULT LOCATION
Presentation Layer	<code><home_dir>/var/logs/nmy_application.log</code>
Synchronizer	<code><home_dir>/var/logs/nmy_synchronizer.log</code>
Connector Template	<code><home_dir>/var/logs/nmy_connectortemplate.log</code>
Loopback Connector	<code><home_dir>/var/logs/nmy_loopback.log</code>
Approval Sequencer	<code><home_dir>/var/logs/nmy_sequencer.log</code>

Example of Rolling File Output Driver Settings

In this example, the logger:

- Saves logs in the `nmy_application.log` file
- Keeps a maximum number of 5 backup copies:
 - `nmy_application.log.1`
 - `nmy_application.log.2`
 - `nmy_application.log.3`
 - `nmy_application.log.4`
 - `nmy_application.log.5`

- Has a maximum file size of 4MB

```
log4j.appender.ROL=org.apache.log4j.RollingFileAppender
log4j.appender.ROL.File=!NMY_VAR_DIR!/logs/nmy_application.log
log4j.appender.ROL.MaxFileSize=4MB
log4j.appender.ROL.MaxBackupIndex=5
log4j.appender.ROL.layout=com.netonomy.util.logger.LoggerLayout
log4j.appender.ROL.layout.ConversionPattern=%d{DATE_TIME}; {EVENT_ID}; {SEVERITY}; {MODULE}; {THREAD_ID}; {EVENT_TYPE}
%; {ERROR_CODE}; {DESCRIPTION}
```

About the Time Stamp Log File Output Driver

This output driver saves log information in a text file.

When using this driver, you specify the following:

- The name and location of the log file
- The interval before creating a backup file. When this interval occurs, the logger saves the file as `<log_filename>.<date>` and starts a new log file.

There is no limit to the number of backup copies of the log file.

You must manage the backup log files, as the logger does not automatically delete them.

To configure a time stamp log file driver

- 1 Go to the `/util` directory containing the `log4j.properties` configuration file.
- 2 Open `log4j.properties`.
- 3 Go to the `#DAILY FILE OUTPUT DRIVER SETTINGS` section.
- 4 Set `log4j.appender.DAY.File` to the full path and log file name.
- 5 Set `log4j.appender.DAY.DatePattern` to one of the following:

SETTING	DESCRIPTION	BACKUP FILE EXTENSION
'.' 'yyyy-MM	Monthly logs that begin at the start of the month	<log_filename>.YYYY-MM
'.' 'yyyy-ww	Weekly logs that start at the beginning of each week	<log_filename>.YYYY-WW
'.' 'yyyy-MM-dd	Daily logs that start at midnight	<log_filename>.YYYY-MM-DD
'.' 'yyyy-MM-dd-h	Daily logs that start at noon and midnight	<log_filename>.YYYY-MM-DD-AM/PM
'.' 'yyyy-MM-dd-HH	Hourly logs that start on the hour	<log_filename>.YYYY-MM-DD-HH
'.' 'yyyy-MM-dd-HH-mm	Minute logs that start every minute	<log_filename>.YYYY-MM-DD-HH-MM

- 1 Set the `log4j.appender.CON.layout.ConversionPattern` to specify the log format.
- 2 Save your changes.

Default Time Stamp Log File Output Settings

By default, the Time Stamp Log File output driver has the following configuration:

- Backup every day at midnight - `DatePattern='.' 'yyyy-MM`
- File: depends on the product layer

COMPONENT	ROLLING FILE DRIVER OUTPUT FILE DEFAULT LOCATION
Presentation Layer	<home_dir>/var/logs/nmy_daily_application.log
Synchronizer	<home_dir>/var/logs/nmy_daily_synchronizer.log
Connector Template	<home_dir>/var/logs/nmy_daily_connectortemplate.log
Loopback Connector	<home_dir>/var/logs/nmy_daily_loopback.log
Approval Sequencer	<home_dir>/var/logs/nmy_daily_sequencer.log

In this example, the logger:

- Saves logs in the `nmy_daily_application.log` file
- Daily log that starts at midnight

```
log4j.appender.DAY=org.apache.log4j.DailyRollingFileAppender
log4j.appender.DAY.File=!NMY_VAR_DIR!/logs/nmy_daily_application.log
log4j.appender.DAY.DatePattern='.'yyyy-MM-dd
log4j.appender.DAY.layout=com.netonomy.util.logger.LoggerLayout
log4j.appender.DAY.layout.ConversionPattern=%{DATE_TIME}; {EVENT_ID}; {SEVERITY}; {MODULE}; {THREAD_ID}; {EVENT_TYPE}
}; {ERROR_CODE}; {DESCRIPTION}
```

About the Output Format

For each driver, you configure the output format. The output format is an arrangement of the event attributes that generates a line in the output.

Each attribute is assigned a pattern you use to define the layout. This means that the format is defined with a string that is a suite of patterns and separators.

Use the syntax:

```
<Driver ConversionPattern setting> = separator  
[EVENT_ATTRIBUTE_PATTERN_1] separator  
[EVENT_ATTRIBUTE_PATTERN_2]...
```

Each of the format definition elements is optional. You can log the event attributes you want.

You cannot use the following characters as layout separators:

- ! (exclamation point)
- : (colon)
- - (dash)

Use the syntax:

```
log4j.appender.<OUTPUT_DRIVER_CODE>.layout.ConversionPattern=  
+{EVENT_OUTPUT_PATTERN1} separator {EVENT_OUTPUT_PATTERN2}...
```

For more information about the attribute patterns, refer to *Output Patterns* in this chapter.

The logger inserts the output as a single line of text in the log file.

If you log DEBUG information, the logger places a block of information in the log between the `<DEBUG_INFO>` and `</DEBUG_INFO>` tags.

Because of this, you should place DEBUG information at the end of your output format. This keeps all of the log information together then lists any associated debug information. Otherwise you may have log information split by a block of debug information.

To specify the output format

- 1 Go to the `/util` directory containing the `log4j.properties` configuration file.
- 2 Open `log4j.properties`.

- 3 Under the `OUTPUT DRIVER` section, set
`log4j.appender.<OUTPUT_DRIVER_CODE>.ConversionPattern` to the
format to use. Use the output format syntax.
- 4 Save your changes.

EVENT ATTRIBUTE PATTERN	DESCRIPTION	NOTES
{DATE_TIME}	Event date and time	The format complies with ISO 8601 standard
{SEVERITY}	Event severity level	<ul style="list-style-type: none"> • FATAL • ERROR • WARN • INFO • DEBUG
{THREAD_ID}	Event thread ID	
{SESSION_ID}	Session ID	<p>Session ID of the user.</p> <p>This is used to identify the user who caused the event to be logged. This ID is the BLM session ID.</p> <p>When there is no BLM session, the value is NONE.</p>
{EVENT_TYPE}	Event type	This is the type of event you can specify
{MODULE}	Event source module	<ul style="list-style-type: none"> • AGT - Any agent – synchronizer, connector, sequencer, CID2CBU loader • BLM - Business Logic Manager • CUS - Custom event • DAL - Data Access Layer • SmartLink (ISF) - SmartLink (ISF) Framework and all of its sub-modules • JSPF - JSPF Framework • JSP - Java Server Pages • LOG - The logger platform • NIL - Unqualified event • UTL - Internal utility components
{EVENT_ID}	Event ID	Unique ID for the specific output driver
{ERROR_CODE}	Event error code	
{DESCRIPTION}	Event description	
{DEBUG_INFO}	Event Debug info	Creates a block of information between the <DEBUG_INFO> and </DEBUG_INFO> tags. This should be used at the end of your debug pattern

Default Format Settings

For product layers and drivers, the default output format is:

```
+{DATE_TIME};{EVENT_ID};{SEVERITY};{MODULE};{THREAD_ID};{EVENT_T  
YPE};{ERROR_CODE};{DESCRIPTION}
```

For the Presentation Layer, the default output format is:

```
+{DATE_TIME};{EVENT_ID};{SESSION_ID};{SEVERITY};{MODULE};{THREAD  
_ID};{EVENT_TYPE};{ERROR_CODE};{DESCRIPTION}
```

Examples of Logger Files

The following examples show the logger configuration files and an example of a logger message.

Example of *logger.properties*

Location of the parameter file	logger.log4j.properties=log4j.properties
Frequency of parameter file reload	logger.log4j.reloadDelay=0

Example of *log4j.properties*

Internal settings DO NOT MODIFY	# DO NOT MODIFY log4j.categoryFactory=com.netonomy.util.logger.LoggerCategoryFactory log4j.rootCategory=INFO,DISABLED
<ul style="list-style-type: none"> Types of events to log All of the events are logged in the ROL log file Exceptions are logged in a DAY log file Unqualified are displayed in the system console 	# EVENTS log4j.category.nmy.INIT=INFO,0,ROL log4j.category.nmy.STATE=INFO,0,ROL log4j.category.nmy.EXCEPTION=INFO,0,ROL, DAY log4j.category.nmy.SESSION=INFO,0,ROL log4j.category.nmy.REQUEST=INFO,0,ROL log4j.category.nmy.MESSAGE=INFO,0,ROL log4j.category.nmy.OBJECT=INFO,0,ROL log4j.category.nmy.RESOURCE=INFO,0,ROL log4j.category.nmy.DATA=INFO,0,ROL log4j.category.nmy.NONE=INFO,0,ROL,CON
Internal settings DO NOT MODIFY	# DO NOT MODIFY log4j.appender.DISABLED=com.netonomy.util.logger.NullAppender
Standard Output Settings	# STANDARD OUTPUT DRIVER SETTINGS log4j.appender.CON=org.apache.log4j.ConsoleAppender log4j.appender.CON.Target=System.out log4j.appender.CON.layout=com.netonomy.util.logger.LoggerLayout log4j.appender.CON.layout.ConversionPattern=%{DATE_TIME};{EVENT_ID};{SESSION_ID};{SEVERITY};{MODULE};{THREAD_ID};{EVENT_TYPE};{ERROR_CODE};{DESCRIPTION}
Rolling File Output Settings	# ROLLING FILE OUTPUT DRIVER SETTINGS log4j.appender.ROL=org.apache.log4j.RollingFileAppender log4j.appender.ROL.File=!NMY_VAR_DIR!/logs/nmy.log log4j.appender.ROL.MaxFileSize=4MB log4j.appender.ROL.MaxBackupIndex=5 log4j.appender.ROL.layout=com.netonomy.util.logger.LoggerLayout log4j.appender.ROL.layout.ConversionPattern=%{DATE_TIME};{EVENT_ID};{SESSION_ID};{SEVERITY};{MODULE};{THREAD_ID};{EVENT_TYPE};{ERROR_CODE};{DESCRIPTION}

Daily File Output Settings	<pre># DAILY FILE OUTPUT DRIVER SETTINGS log4j.appender.DAY=org.apache.log4j.DailyRollingFileAppender log4j.appender.DAY.File=!NMY_VAR_DIR!/logs/nmy_daily_log.log log4j.appender.DAY.DatePattern='.'yyyy-MM-dd log4j.appender.DAY.layout=com.netonomy.util.logger.LoggerLayout log4j.appender.DAY.layout.ConversionPattern=+{DATE_TIME};{EVENT_ID};{SESSION_ID};{SEVERITY};{MODULE};{THREAD_ID};{EVENT_TYPE};{ERROR_CODE};{DESCRIPTION}</pre>
----------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

This example shows the initialization log.

```
+2002-03-06 17:39:23.830;EC6C8259B6ExecuteThread-310;INFO;LOG;ExecuteThread-31;INIT;0012000;Setting logger
configuration file to [path="classes/nmycfg/util/log4j.properties"].
```

```
+2002-03-06 17:39:23.850;EC6C8259B6ExecuteThread-311;INFO;LOG;ExecuteThread-31;INIT;0012001;Set dynamic reload
of logger configuration every [frequency=30] seconds.
```

```
+2002-03-06 17:39:29.889;EC6C8259B6ExecuteThread-312;INFO;DAL;ExecuteThread-31;INIT;0011001;Initializing
datasource [name="cidDatasource"], [driver="JNDI t3://localhost:7001"], [user="N/A"] succeeded.
```

```
+2002-03-06 17:39:52.232;EC6C8259B6ExecuteThread-313;INFO;DAL;ExecuteThread-31;INIT;0010101>Loading component
configuration [name="DAL"] succeeded.
```

```
+2002-03-06 17:40:18.760;EC6C8259B6ExecuteThread-314;INFO;BLM;ExecuteThread-31;INIT;0010101>Loading component
configuration [name="BLM"] succeeded.
```

```
+2002-03-06 17:40:21.635;EC6C8259B6ExecuteThread-315;INFO;JFN;ExecuteThread-31;INIT;0040001>Loading media
application file [path="<home_dir>/channels/WEB-INF/classes/nmycfg/jfn/MyWeb.xml"] succeeded.
```

```
+2002-03-06 17:40:22.255;EC6C8259B6ExecuteThread-316;WARN;BLM;ExecuteThread-31;STATE;0022102;Checking the
validity of the reference data cache. Cache out of date and cleared.
```

This example shows the log when an error occurs when loading a required file.

```
+2003-01-20 18:18:35.261;F2DC8E7ABEEExecuteThread: '12' for queue: 'default'63;1060;INFO;BLM;ExecuteThread:
'12' for queue: 'default';INIT;0010101>Loading component configuration [name="nmycfg.blm.config"] succeeded.
```

```
+2003-01-20 18:21:21.568;F2DC8E7ABEEExecuteThread: '12' for queue: 'default'64;1060;ERROR;DAL;ExecuteThread:
'12' for queue: 'default';DATA;0011404;Executing SQL statement failed [execute type="update"]: [DB error="1"].
```

```
+2003-01-20 18:25:40.728;F2DC9FB90CEExecuteThread: '12' for queue: 'default'0;NONE;INFO;LOG;ExecuteThread: '12'
for queue: 'default';INIT;0012000;Setting logger configuration file to [path="nmycfg/util/log4j.properties"].
```

```
+2003-01-20 18:25:40.768;F2DC9FB90CEExecuteThread: '12' for queue: 'default'1;NONE;INFO;LOG;ExecuteThread: '12'
for queue: 'default';INIT;0012002;Dynamic reload of logger configuration is not activated.
```

```
+2003-01-20 18:25:48.429;F2DC9FB90CEExecuteThread: '12' for queue: 'default'2;NONE;FATAL;DAL;ExecuteThread:
'12' for queue: 'default';INIT;0011002;Initializing datasource [name="jdbc/cidDatasource"], [driver="JNDI "],
[user="N/A"] failed.
```

```
+2003-01-20 18:25:48.650;F2DC9FB90CEExecuteThread: '12' for queue: 'default'3;NONE;FATAL;DAL;ExecuteThread:
'12' for queue: 'default';INIT;0030105;Instantiating Data Access Layer driver
[instance="nmycfg.dal.instances.instance_route"],
[driver="com.netonomy.dal.drivers.impl.sql.jndi.JNDIDataSourceInstance"] failed.
```

```
+2003-01-20 18:25:48.650;F2DC9FB90CEExecuteThread: '12' for queue: 'default'4;NONE;FATAL;DAL;ExecuteThread:
'12' for queue: 'default';INIT;0010002;Initializing component [name="DAL Authentication Module"] failed.
```


Using Logs with a Supervision Platform

When using a supervision platform to manage or analyze logs, you can do the following to maximize the performance and quality of information in the logs:

- Adapt the output format for your platform
- Use certain event attributes
- Take advantage of event codes

Adapting the Output Format

Most supervision platforms offer the possibility to parse text files with pattern matching.

This feature is based on the following:

- In the generated file, there is one event per line
- The line structure is known

You can adapt the output format of log files to make pattern matching easier.

As this is also based on the principle that there is one event per line, you should not log `DEBUG_INFO`.

The default output format settings follow these recommendations.

Using Event Attributes

Some event attributes are very helpful to supervise:

- **Event Id**
Because the ID is unique, you can use this information as a reference when the alert is to be forwarded to tracking systems.
- **Session Id**
This ID allows you to track events which are generated by the same user. This way you can pinpoint and follow the events that are logged. If there is no associated user session id, you can still track the sequence of logged events by using the thread id.
- **Event codes**

Using Event Codes

When using event codes, the main attribute to check is severity. This is because severity gives you an idea as to the impact of the event on the application. But the real impact depends on your application. As most events are coded, you can use this to your advantage when checking logs.

For event codes:

- Each code is associated with a description (the event description attribute value).
- In some descriptions, there are some fixed parts (the description parameters).

Description parameters refer to the dynamic part of the description. But they are always marked with a fixed and documented tag. This part of the event code is the code interface and will not change even if the description does.

An example:

Your deployment relies on a file system that is extremely sensitive and you want to monitor events that deal with your application interacting with this file system.

There are some event codes that are dedicated to file system interaction.

For instance, the code 2004 is associated with the event description "Cannot write to file [path=" <the path> "]."

In this case, the section [path=""] is fixed.

This means you can:

- Monitor logging of event code '2004'.
- If the event occurs, check the 'path' description parameter.

If the path that appears just after the <[path="> tag, you can generate a critical alert for your supervision system and have people work on the issue quickly.

For more information about the standard codes, refer to the *Logger Message Reference Documentation*.

Filtering by Source Component

You can also configure the logger to filter events by component. You can use this feature to help application development and debugging as you focus on event source components.

If you have customized a part of the application and want to check logs for testing or debugging, you can focus on the component you work with, activate full debug information, without getting a huge amount of events from components you are not interested in.

You filter events by specifying event types. In addition to event types, you can filter logged events by using the event component parameter.

You use the syntax:

```
<Event Type>.<Event Source> = <Severity Level>, <Output Driver #1>, <Output Driver #2>,...
```

For example, `log4j.category.nmy.INIT.BLM=INFO,ROL`

- Reduces the logs for INIT-type events to the ones issued from the BLM.
- Logs events using the Rolling File output, and there is no DEBUG event generated.

The event modules you can filter on are:

EVENT COMPONENT	DESCRIPTION
ADM	Administration
AGT	Any agent – synchronizer, connector, sequencer, CID2CBU loader
BLM	Business Logic Manager
CUS	Custom event
DAL	Data Access Layer
SmartLink (ISF)	SmartLink (ISF) Framework and all of its sub-modules
JSPF	JSPF Framework
JSP	Java Server Pages
LOG	The logger platform
NIL	Unqualified event
QRA	Query and Report Analysis

UTL	Internal utility components
WFS	Web File System

Examples of Filtering by Component

- 1 Get all types of event but only ones generated by the BLM – no debug information, rolling file

```
# EVENTS
log4j.category.nmy.INIT.BLM=INFO,0,ROL
log4j.category.nmy.STATE.BLM=INFO,0,ROL
log4j.category.nmy.EXCEPTION.BLM=INFO,0,ROL
log4j.category.nmy.SESSION.BLM=INFO,0,ROL
log4j.category.nmy.REQUEST.BLM=INFO,0,ROL
log4j.category.nmy.MESSAGE.BLM=INFO,0,ROL
log4j.category.nmy.OBJECT.BLM=INFO,0,ROL
log4j.category.nmy.DATA.BLM=INFO,0,ROL
log4j.category.nmy.NONE.BLM=INFO,0,ROL
...
```

- 1 Get only 'INIT' events, but block the ones generated by the DAL – debug level 3, rolling file

```
# EVENTS
log4j.category.nmy.INIT.BLM=INFO,0,ROL
log4j.category.nmy.INIT.JFN=INFO,0,ROL
log4j.category.nmy.INIT.JSP=INFO,0,ROL
log4j.category.nmy.INIT.AGT=INFO,0,ROL
log4j.category.nmy.INIT.ISF=INFO,0,ROL
log4j.category.nmy.INIT.LOG=INFO,0,ROL
log4j.category.nmy.INIT.UTL=INFO,0,ROL

log4j.category.nmy.STATE.BLM=INFO,DISABLED
log4j.category.nmy.EXCEPTION.BLM=INFO,DISABLED
log4j.category.nmy.SESSION.BLM=INFO,DISABLED
log4j.category.nmy.REQUEST.BLM=INFO,DISABLED
log4j.category.nmy.MESSAGE.BLM=INFO,DISABLED
log4j.category.nmy.OBJECT.BLM=INFO,DISABLED
log4j.category.nmy.DATA.BLM=INFO,DISABLED
log4j.category.nmy.NONE.BLM=INFO,DISABLED
...
```

- 1 Get all types of event, but block events generated by the JSPF and the JSP – no debug information, rolling file

```
# EVENTS

log4j.category.nmy.INIT.BLM=INFO,0,ROL
log4j.category.nmy.INIT.DAL=INFO,0,ROL
log4j.category.nmy.INIT.AGT=INFO,0,ROL
log4j.category.nmy.INIT.ISF=INFO,0,ROL
log4j.category.nmy.INIT.LOG=INFO,0,ROL
log4j.category.nmy.INIT.UTL=INFO,0,ROL

log4j.category.nmy.STATE.BLM=INFO,0,ROL
log4j.category.nmy.STATE.DAL=INFO,0,ROL
log4j.category.nmy.STATE.AGT=INFO,0,ROL
log4j.category.nmy.STATE.ISF=INFO,0,ROL
log4j.category.nmy.STATE.LOG=INFO,0,ROL
log4j.category.nmy.STATE.UTL=INFO,0,ROL

log4j.category.nmy.EXCEPTION.BLM=INFO,0,ROL
log4j.category.nmy.EXCEPTION.DAL=INFO,0,ROL
log4j.category.nmy.EXCEPTION.AGT=INFO,0,ROL
log4j.category.nmy.EXCEPTION.ISF=INFO,0,ROL
log4j.category.nmy.EXCEPTION.LOG=INFO,0,ROL
log4j.category.nmy.EXCEPTION.UTL=INFO,0,ROL
...
```

Note:

- You have to define filtering for all couples (Event Type, Event Module) that you are interested in.
- Blocking JSPF and JSP is achieved by not mentioning the modules at all.

CHAPTER 6

Auditing Telco Service & Analytics Manager Applications

In This Section

About Auditing	112
Activating User Events	113
Generating Reports	114
Purging User Events	115

About Auditing

You can monitor the system User Events to track the actions of users. For instance, you can track the events of a user who adds a service to their contract, from the moment they click the submit button to the confirmation message received from the OSS.

User events belong to one of the following categories:

- Session events (login, logout, session expiration)
- Execution features (creation of requests)
- Custom user events
- DO events generated by an OSS
- Organization views
- Reporting

There is a set of predefined user event types in the CID. These user event types are stored in the `USER_TYPE_EVENT` table. User events are logged in the `USER_EVENT` table.

Using User Events involves:

- Activating user events
- Viewing user events
- Managing user events
- Creating custom user events
- Purging the User Event table

Activating User Events

By default, all user events are inactive. Before you can start using user events for an audit, you need to activate the user events you want to track.

User event settings are reference data. If you want to automatically reload this information after changing your settings, refer to *Reloading Reference Data* in *Developing Telco Service Manager (TSM)*.

To activate user events

- 1 Use your database tool to connect to the CID.
- 2 In the `USER_TYPE_EVENT` table, find the user event you want to activate.
- 3 In the `ACTIVATION_FLAG` column, enter 1.
- 4 Save your changes.
- 5 Do one of the following:
 - Use the Reference Data Reload feature to refresh the reference data
 - Restart your application server.

To deactivate user events

- 1 Use your database tool to connect to the CID.
- 2 In the `USER_TYPE_EVENT` table, find the user event you want to activate.
- 3 In the `ACTIVATION_FLAG` column, enter 0.
- 4 Save your changes.
- 5 Do one of the following:
 - Use the Reference Data Reload feature to refresh the reference data
 - Restart your application server.

Generating Reports

The cidAdminTool comes with a command you use to quickly generate logon statistics. If you need to create your own customized reports, you can use the methods in the `com.netonomy.blm.interfaces.event` package to display and create other reports.

To generate a login report

1 Go to `<home_dir>/bin`.

2 Run the `cidAdminTool` administration tool. Use the syntax:

```
cidAdminTool generate_logon_stats <CID> <CID_ADMIN login>
<CID_ADMIN password> <m|q|y> [-quiet]
```

where `<CID>`:

- Oracle: `<instance alias>`
- DB2: `<instance alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port
- `m` generate stats for the last completed month
- `q` generate stats for the last completed quarter
- `y` generate stats for the last completed year

The CID administration tool extracts the information and displays the location and filename of the extracted information. The report files are located in `<home_dir>/<var>/output/admin`.

Purging User Events

You can remove user events that are saved in the CID.

To purge user events

1 Go to `<home_dir>/bin`.

2 Run the CID administration tool. Use the syntax:

```
cidAdminTool purge_user_events <CID> <CID_ADMIN login>  
<CID_ADMIN password> <days> [<code>,<code>,...] [-quiet] [-  
datablocksize:####]
```

where <CID>:

- Oracle: `<instance alias>`
- DB2: `<database alias>`
- SQL Server: `<database host> [:<port>]` If no port is specified, the tool uses the default SQL server port

The CID administration tool removes all of the user events that:

- Have the specified statuses
- Are older than the specified number of days

When finished, the administration tool displays a confirmation message.

If you do not specify a user event code, the CID administration tool purges all of the user events.

CHAPTER 7

Managing Distributed Architectures

In This Section

About Distributed Architectures.....	118
Installing and Configuring for Distributed Architectures	119
Managing Sessions.....	120

About Distributed Architectures

Except for some development computers, you do not install all of the components on a single computer. Your application server, Web server, OSS systems and databases may be on different computers and even may be running different operating systems.

This section covers some of the things you have to keep in mind when working in distributed architectures.

Installing and Configuring for Distributed Architectures

Because components may be installed on several different computers, you need to make sure that they have all of the required access and permissions for shared directories.

You need to make sure that the components can access and has write permission for the following directories:

- `<home_dir>/var` and its subdirectories
- `<home_dir>/share` and its subdirectories

Your TSM uses this directory to store logs, connector message queues, and so on.

Managing Sessions

Your TSM supports session serialization to take advantage of the load balancing and failover features of your application server. You can use your application server's features to:

- Maintain information through HTTP session
- Persist this information in a database, file system, and so on
- Share the session context between application server instances

By default, all of the TSM objects support session serialization.

When customizing your application, you need to make sure you implement the `java.io.Serializable` interface in order to support session serialization. For more information about implementing this interface, refer to your application server's documentation.

If you use a database to store sessions, you must store TSM sessions in one block. You cannot store them in multiple rows.

APPENDIX A

Administration Tool Reference

In This Section

About the Administration Tools	122
CID Administration Tool	123
CID2CBU Loader Administration Tools.....	133
CBU Administration Tool	143
Synchronizer Connector Administration Tools	147
OSS Connector Administration Tools	151
NOTIFYORG Administration Tool	153
Approval Sequencer Administration Tools	155
CustDim Loader Administration Tools.....	157
SmartLink (ISF) Message Cache Administration Tool	161
Report Manager Administration Tool	165
Job Service Administration Tools.....	168

About the Administration Tools

A complete set of administration tools helps you manage the different components of your TSM.

The name of the files depends on your operating system. For Windows, the tools have a `.cmd` extension and for UNIX a `.sh` extension. The behavior of these tools is identical and they have the same command line options.

Make sure that the `CLASSPATH` and `LIB_PATH` in these files include the path to all required files. If not, open the required `.env` files and add the paths where required.

CID Administration Tool

You use the `cidAdminTool` to manage the CID.

This administration tool is located in `<home_dir>/bin`.

For very large databases, this tool comes with a `datablock` parameter for commands that manage tables containing large amounts of data (requests, trouble tickets, and so on.) This parameter lets you set the number of records to process before the tool commits changes. This way, you can handle large amounts of data in smaller blocks of information that are easier to manage and help avoid various issues while interacting with the database. You can also override this setting from the command line when administering the CID.

However, by limiting the number of items the tool may have to submit several changes to the database. For instance, if your `REQUEST` table contains 3,000 requests to purge and your default `datablock` size is 1,000, the tool commits 3 changes. If an error occurs during this process, there is no way to carry out a global rollback of your changes.

Configuring the CID Administration Tool

After installation, you can change the settings of the CID administration tool.

You use the `cid_tools.properties` configuration file to set the properties of the CID administration tool. This file is located in `<home_dir>/lib/admin/cid`.

Configuring the CID Administration tool involves:

- Specifying the JDBC driver
- Specifying the database URL
- Specifying the encoding
- Specifying the data block size

When configuring the CID Administration tool, keep in mind that you can override the data block size when running the tool.

To modify the settings of the CID administration tool

- 1 Go to `<home_dir>/lib/admin/cid`.
- 2 Open `cid_tools.properties`.
- 3 Change the following settings:

SETTING	DESCRIPTION
DRIVER	Class name of the ODBC driver to use
URL	URL of the CID database
ENCODING	Encoding to use
DATABLOCKSIZE	<p>Default number of items to handle before submitting the data.</p> <p>The CID administration tool processes the data in sets corresponding to the specified size until done.</p> <p>This value can be overridden when running the CID administration tool.</p>

Example of cid_tool.properties for Oracle

JDBC driver	<pre># driver: This is the class name of the jdbc driver driver=oracle.jdbc.driver.OracleDriverdriver=oracle.jdbc.driver.OracleDriver</pre>
URL of the database	<pre># url:This is url for the database (typically jdbc:oracle:oci8:@OracleCidName) url=jdbc:oracle:oci8:@CID</pre>
<p>Internal settings DO NOT MODIFY</p> <p>If you need to modify these settings, you must reinstall and enter the new values during installation.</p>	<pre># SQL Base Path SqlBasePath=<home_dir>/data/oracle/ # LOG Path LogPath=<home_dir>/var/logs/admin/ # OUTPUT PATH OutPath=<home_dir>/var/output/admin/ # VERSION STRING Version=X.X.X.XXX</pre>
Encoding	<pre># ENCODING Encoding=ISO-8859-1</pre>
Data Block Size	<pre># DATABLOCKSIZE DataBlockSize=10000</pre>

cidAdminTool Syntax

```
cidAdminTool help | <command> <parameters> [<optional
parameters>] <help>
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<command>	Administration command
<parameters>	Specifies the parameters for the command
[<optional parameters>]	Optional parameters
<command> help	Displays help for the command

COMMAND	DESCRIPTION	PARAMETERS
admin_notifications	Manages notifications stored in the CID notification queue	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>

admin_requests	Manages requests stored in the CID request queue	<p><CID></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>
create_cid_users	Creates the CID_ADM and CID_USR users and their corresponding roles	<p><CID></p> <ul style="list-style-type: none">• Oracle: <instance alias>• SQL Server: <database host> [:<port>];DatabaseName="<db_name>" If no port is specified, the tool uses the default SQL server port <p>For SQL Server, note that the semicolon is a special character of UNIX and Windows shell. The CID parameter should be enclosed by quotation marks.</p> <p>For DB2, the CID users are system users and cannot be created using this tool.</p> <p><login> DBA Login ID</p> <p><password> DBA Login password</p> <p>< new CID_ADMIN login> login for the CID admin account</p> <p><new CID_ADMIN password> password for the CID admin account</p> <p><new CID_USER login> login for the CID user account</p> <p><new CID_USER password> password for the CID user account</p> <p>[<quiet>] Do not display information</p>

create_demo_cid_structure	Creates a CID demo database structure with reference data, but without the demo dataset	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p>[-quiet] Do not display information</p>
create_demo_cid_test	Creates a CID demo database structure with reference data, and populates it with the demo dataset	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p>[-quiet] Do not display information</p>
drop_demo_cid_structure	Drops a CID demo database structure along with the populated data	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p>[-quiet] Do not display information</p>

generate_logon_stats	Extracts logon statistics from CID to a file	<p><CID></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><m q y></p> <ul style="list-style-type: none">• m generate stats for the last completed month• q generate stats for the last completed quarter• y generate stats for the last completed year <p>[-quiet] Do not display information</p>
install_cid	Creates the CID database structure and populates the required reference tables	<p><CID></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><CID_USER login> login for the CID user account</p> <p><CID_USER password> password for the CID user account</p> <p>[-quiet] Do not display information</p>

purge_requests	Purges the requests stored in the CID request queue	<p><CID></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><days> The age of the requests to keep in days. For example, entering 7 purges all of the requests that were not created during the last week.</p> <p>[<status>, <status>...] The status of the requests to purge</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>
----------------	-----------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<p>purge_persistent_action_managers</p>	<p>Purges the persistent action managers (shopping carts) stored in the CID</p>	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><days> The age of the persistent action managers to keep in days. For example, entering 7 purges all of the persistent action managers that were not created during the last week.</p> <p><category> The category of persistent action managers to purge.</p> <p>By default:</p> <ul style="list-style-type: none"> • 1 backup shopping carts • 2 contract templates <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>
<p>purge_trouble_tickets</p>	<p>Purges the trouble tickets stored in the CID</p>	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><days> The age of the trouble tickets to purge in days</p> <p>[<status>,<status>...] The status of the trouble tickets to purge</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>

purge_user_events	Purges the user events stored in the CID	<p><CID></p> <ul style="list-style-type: none"> ▪ Oracle: <instance alias> ▪ DB2: <database alias> ▪ SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><days> The age of the events to keep in days. For example, entering 7 purges all of the events that were not created during the last week.</p> <p>[<event_code>[,<event_code>...]] The code of the type(s) of events to purge. If none entered, all events are purged.</p> <p>The event code corresponds to the the USER_TYPE_EVENT_ID in the USER_TYPE_EVENT table.</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>
-------------------	------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

purge_notifications	Purges the notifications stored in the CID notification queue	<p><CID></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><days> The age of the notifications to keep in days. For example, entering 7 purges all of the notifications that were not created during the last week.</p> <p>[<status>,<status>...] The status of the notifications to purge. If none specified, the tool purges FAILED and DONE notifications.</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>
---------------------	---------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

CID2CBU Loader Administration Tools

You use the following administration tools to administrate the CID2CBU Loader:

- `cid2cbuloader`
- `cid2cbuloaderadm`

These administration tools are located in `<home_dir>/bin`.

Configuring the CID2CBU Loader

The CID2CBU loader acts as a polling process that extracts notifications from the CID and updates the CBU database.

Configuring the CID2CBU loader involves:

- Setting the CID2CBU properties
- Optimizing the CID2CBU properties when required
- Setting the CID2CBU database connection properties

Setting the CID2CBU Loader Properties

You can define several CID2CBU loaders.

The CID2CBU loader uses the `cid2cbuloader.properties` configuration file to set its properties. This file is located in `<home_dir>/config/cid2cbuloader`.

For each CID2CBU loader, you configure a `cid2cbuloader.properties` configuration file.

You can configure the following:

- The user name and password to authenticate the CID2CBU loader to the BLM
- The BLM connection retry mechanism
- The CBU Database connection retry mechanism
- The number of notifications to extract from the notification queue
- The stability delay of a notification before extracting it
- The list of notification types-object types to extract
- The average number of notifications selected still to treat (Queue threshold) for one thread before selecting new ones.
- The sleep time between two executions of the notification selection when the first one returns nothing to treat.
- The sleep time between two verifications of the Queue threshold before selecting new notifications
- The running mode of the CID2CBU loader (one shot or not)
- The number of threads to allocate to process notifications
- The configuration directory path
- The administration port

To configure the CID2CBU loader

- 1 Go to `<home_dir>/config/cid2cbuloader`.
- 2 Open `cid2cbuloader.properties`.
- 3 Enter the following:

PARAMETER	DESCRIPTION
CONFIG_DIR	Full path of the directory containing the cid2cbuloader.properties configuration file
ADM_PORT	Port number you use to administer the CID2CBU loader (this parameter is set by the installer)
NAPPING_TIME	Milliseconds between two executions of the notification selection when the first one returns nothing to process
LOOPING_TIME	Milliseconds between two verifications of the queue threshold before selecting new notifications
Q_THRESHOLD	Average number of notifications selected still to process for one thread before selecting new notifications
BLM_USER	User login to authenticate the CID2CBU loader for the BLM
BLM_PASSWORD	Associated password
BLM_RECONNECT_RETRIES	Number of times the CID2CBU loader tries to reconnect to the BLM
BLM_RECONNECT_DELAY	Milliseconds between two BLM connection retries
CBU_RECONNECT_RETRIES	Number of times the CID2CBU loader tries to reconnect to the CBU
CBU_RECONNECT_DELAY	Milliseconds between two CBU connection retries
NB_NOTIFICATIONS	Number of notifications to extract from the notification queue
NOTIFICATION_STABILITY_DELAY	Milliseconds between a notification creation date-time and current date time to extract it.
NB_THREADS	Number of threads allocated to process notifications
ONE_SHOT	TRUE for one shot mode. This mode is usually for loading an empty CBU. FALSE for agent mode
NOTIFICATION_FILTER	TRUE to process the notification type-object type couple. FALSE to ignore Example: <code>{"CREATE",MEMBER,true}, {"MODIFY",MEMBER,false}</code> The CREATE notification event of a MEMBER is processed by the CID2CBU loader. The MODIFY notification event of a MEMBER is ignored.

- 1 Save your changes.

Example of cid2cbuloader.properties

CID2CBU Loader parameters	<code>CONFIG_DIR="file:///<home_dir>/config/cid2cbuloader/"</code> <code>ADM_PORT=3004</code> <code>NAPPING_TIME=1000</code> <code>LOOPING_TIME=500</code> <code>NB_THREADS=4</code>
BLM connection parameters	<code>BLM_USER="agent"</code> <code>BLM_PASSWORD="agent"</code> <code>BLM_RECONNECT_DELAY=60000</code> <code>BLM_RECONNECT_RETRIES=30</code>
CBU connection parameters	<code>CBU_RECONNECT_DELAY=60000</code> <code>CBU_RECONNECT_RETRIES=30</code>
Queue parameters	<code>Q_THRESHOLD=2</code> <code>NB_NOTIFICATIONS=1000</code> <code>NOTIFICATION_STABILITY_DELAY=5000</code> <code>NOTIFICATION_FILTER=1</code>
Mode	<code>ONE_SHOT=false</code>
Notification filter for notification type-object type couples	<code>NOTIFICATION_FILTER={{"CREATE",ORGANIZATION,true},{"MODIFY",ORGANIZATION,true},{"UPDATE",ORGANIZATION,true}, \</code> <code> {"CREATE",MEMBER,true},{"MODIFY",MEMBER,true}, \</code> <code> {"CREATE",CONTRACT,true},{"MODIFY",CONTRACT,true},</code> <code> {"CREATE",CONTACT,true},{"MODIFY",CONTACT,true}, \</code> <code> {"CREATE",BILLINGACCOUNT,true},{"MODIFY",BILLINGACCOUNT,true}, \</code> <code> {"CREATE",LOGIN,true},{"MODIFY",LOGIN,true}, \</code> <code> {"CREATE",ORGVIEW,true},{"MODIFY",ORGVIEW,true},{"REMOVE",ORGVIEW,true} }</code>

Configuring the CID2CBU Loader Database Connection

You can define the CID2CBU loader database connection parameters. By default, you enter the CID2CBU connection parameters for the CID and CBU during installation. However, you may need to change the connection parameters after installation.

The CID2CBU loader uses the following configuration files:

- `instance_route.properties` configuration file to access the CID.
- `cbu_instance.properties` configuration file to access the CBU.

These files are located in

`<home_dir>/config/cid2cbuloader/nmycfg/dal/instances.`

To configure the database connection

- 1 Go to `<home_dir>/config/cid2cbuloader/nmycfg/cid2cbu.`
- 2 Do one of the following:

- To change the connection to the CID, open `instance_route.properties`.
 - To change the connection to the CBU, open `cbu_instance.properties`
- 3 Enter the following:
 - `DRIVER`: enter the name of the driver to use
 - `URL`: enter the location of the database
 - `USER`: enter the login
 - `PASSWORD`: enter the password
 - 4 Save your changes.

Optimizing the CID2CBU loader

The default CID2CBU loader settings may not correspond exactly to your system architecture and resources. When not optimized for your environment, the CID2CBU loader may not respond to your performance targets for data synchronization.

To optimize the performances of the CID2CBU loader, you have to change the following settings for the Java Virtual Machine (JVM):

- Number of threads
- Allocated memory

By changing these settings, you ensure that the CID2CBU loader is running at optimal performance for your system.

Optimizing the CID2CBU loader involves:

- Determining the system resources available for the CID2CBU loader
- Specifying the number of threads in the `cid2cbuloader.properties` configuration file
- Specifying the allocated memory in the CID2CBU loader administration tool

To optimize the CID2CBU

- 1 Determine the number of processors dedicated to running the CID2CBU loader. The number of processors is referred to as `N`.
- 2 Determine the maximum amount of RAM to use per dedicated processor. The maximum available RAM is referred to as `MR`. By definition, $MR = \text{Total RAM} / N$, although this may be different for your environment.
- 3 Specify the number of threads per processor. Do the following:
 1. Go to `<home_dir>/config/cid2cbuloader`.
 2. Open `cid2cbuloader.properties`.

3. Set the `NB_THREADS` setting to the number of threads the CID2CBU loader uses.

The number of threads should be limited to 1 or 2 per processor.

- 4 Determine the minimum and maximum memory allocated to the JVM. Use the following formulas to determine the values of the JVM command line arguments:

JVM Arguments:

- `MaxJVMSize = (MR) x (N)`
- `MinJVMSize = MaxJVMSize`

If the `MaxJVMSize` is greater than the maximum allowed memory for a process, you can run more than one instance of the CID2CBU loader. The maximum amount depends on your system. For instance, the maximum available RAM for Solaris 8 is 4GB. If you run more than one CID2CBU loader, you need to install it and specify a different administration port and configure it by following these instructions.

If the minimum amount of RAM is not equal to the maximum, enter the minimum amount of guaranteed RAM. By entering the amount of guaranteed RAM, you can prevent problems when the process or system are restarted.

New Object Size Arguments:

- `MaxNewSize=MaxJVMSize/3`
- `NewSize=MaxJVMSize/3`

- 5 Enter the JVM command line arguments to the CID2CBU loader administration tool. Do the following:

1. Go to `<home_dir>/bin`.
2. Open the `cid2cbuloader` file.
3. Change the following Java command line arguments:

DEFAULT	NEW
<code>-ms64m</code>	<code>-XmsMaxJVMSize</code>
<code>-mx128m</code>	<code>-XmxMinJVMSize</code>

1. Add the following Java command line arguments:

```
-XX:MaxNewSize=MaxNewObjectSize  
-XX:NewSize=MaxNewObjectSize
```

Your command line should look like this:

```
$NMY_JAVA_HOME/bin/java -XmsMaxJVMSize -XmxMinJVMSize -  
XX:MaxNewSize=MaxNewObjectSize -XX:NewSize=MaxNewObjectSize  
...
```

cid2cbuloader Syntax

```
cid2cbuloader <connector_name>
```

PARAMETERS	DESCRIPTION
<connector_name>	Name of the connector to start

cid2cbuloderadm Syntax

```
cid2cbuloderadm help | <host> <port> <command> [<parameters>]
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<host>	Specifies the agent host
<port>	Specifies the agent administration port
<command>	Administration command

COMMAND	DESCRIPTION	PARAMETERS	RETURN VALUES
start	Resumes the execution of the connector. Use this command to restart the connector after a stop command.	none	-200 AGENT START
stop	Stops the connector. The connector no longer processes inbound and outbound messages	none	-120 AGENT STOP
shutdown	Terminates the connector	none	-100 AGENT SHUTDOWN
setparameter	Changes the settings of the connector remotely. Use the following syntax: setparameter <parameter>=<value>	AGENTNAPPING AGENTLOOPING ONE_EXECUTION NBRROW QFILL MAXNUMBERRETRY RETRYDELAY	-510 AGENT SET<parameter name>=<parameter value>

getparameter	Restarts the current value of the parameter	AGENTNAPPING AGENTLOOPING ONE_EXECUTION NBRROW QFILL MAXNUMBERRETRY RETRYDELAY Any parameter defined in the properties files	-610 AGENT GET<parameter name>=<parameter value>
getmode	Displays the current running mode	none	-210 AGENT NORMAL -220 AGENT REDUCE -230 AGENT PAUSE
getstatus	Displays the current status	none	130 AGENT NORMAL -140 AGENT REDUCE -150 AGENT PAUSE
force	Forces the connector to run in the specified mode	NORMAL REDUCE RECOVER	-410 AGENT SET MODE=NORMAL -410 AGENT SET MODE=REDUCE -410 AGENT SET MODE=RECOVER For MODE=RECOVER, an additional parameter must be provided: FILTER <FilterName>
info	Displays information about the connector	none	[AGENTNAME:xxx,NBTHRE ADS:xxx,FILL:xxx,NBROW:x xx,NAPPING:xxx,LOOPING: xxx,VERSION:xxx,BUILD:xx x,OS:xxx]
kill	Forces shutdown	none	none
list	Returns all the profiling information	none	none
purge	Purges the profiling information	none	none
stat	Returns the profiling information of the last profiling element	none	none
version	Returns the version of the connector	none	none

CBU Administration Tool

You use the `cbuAdminTool` to manage the CBU.

This administration tool is located in `<home_dir>/bin`.

For very large databases, this tool comes with a `datablock` parameter for commands that manage tables containing large amounts of data (requests, trouble tickets, and so on.) This parameter lets you set the number of records to process before the tool commits changes. This way, you can handle large amounts of data in smaller blocks of information that are easier to manage and help avoid various issues while interacting with the database. You can also override this setting from the command line when administering the CBU.

However, by limiting the number of items the tool may have to submit several changes to the database. For instance, if your table contains 3,000 records to purge and your default `datablock` size is 1,000, the tool commits 3 changes. If an error occurs during this process, there is no way to carry out a global rollback of your changes.

Configuring the CBU Administration Tool

After installation, you can change the settings of the CBU administration tool.

You use the `cbu_tools.properties` configuration file to set the properties of the CBU administration tool. This file is located in `<home_dir>/lib/admin/cbu`.

Configuring the CBU Administration tool involves:

- Specifying the JDBC driver
- Specifying the database URL
- Specifying the encoding
- Specifying the data block size

To modify the settings of the CBU administration tool

- 1 Go to `<home_dir>/lib/admin/cbu`.
- 2 Open `cbu_tools.properties`.
- 3 Change the following settings:

SETTING	DESCRIPTION
DRIVER	the class name of the ODBC driver to use
URL	the URL of the CBU database
ENCODING	the encoding to use
DATABLOCKSIZE	<p>Default number of items to handle before submitting the data.</p> <p>The CBU administration tool processes the data in sets corresponding to the specified size until done.</p> <p>This value can be overridden when running the CBU administration tool.</p>

- 1 Save your changes.

Example of cbu_tool.properties

JDBC driver	<pre># driver: This is the class name of the jdbc driver (typically oracle.jdbc.driver.OracleDriver) driver=oracle.jdbc.driver.OracleDriver</pre>
URL of the database	<pre># url: This is url for the database (typically jdbc:oracle:oci8:@OracleCidName) url=jdbc:oracle:oci8:@CBU</pre>
<p>Internal settings DO NOT MODIFY</p> <p>If you need to modify these settings, you must reinstall Telco Service & Analytics Manager and enter the new values during installation.</p>	<pre># SQL Base Path SqlBasePath=<home_dir>/data/cbu/oracle/ # LOG Path LogPath=<home_dir>/var/logs/cbu_adm/ # OUTPUT PATH OutPath=<home_dir>/var/output/cbu_adm/ # VERSION STRING Version=3.1.0.407</pre>
Encoding	<pre># ENCODING Encoding=ISO-8859-1</pre>
Data Block Size	<pre># DATABLOCKSIZE DataBlockSize=10000</pre>

cbuAdminTool Syntax

```
cbuAdminTool help | <command> <parameters> [<optional
parameters>] <help>
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<command>	Administration command
<parameters>	Specifies the parameters for the command
[<optional parameters>]	Optional parameters
<command> help	Displays help for the command

COMMAND	DESCRIPTION	PARAMETERS
create_cbu_users	Creates the default CBU_ADM and CBU_USR users and their corresponding roles	<p><CBU></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • SQL Server: <database host> [:<port>];DatabaseName="db_name" <p>If no port is specified, the tool uses the default SQL server port</p> <ul style="list-style-type: none"> • For DB2, the CID users are operating system users and cannot be created using this tool. <p><DBA login>DBA Login ID</p> <p><DBA password>DBA Login password</p> <p><CBU_ADMIN login> login for CBU admin account</p> <p><CBU_ADMIN password>: password for CBU admin account</p> <p><CBU_USER login> login for CBU user account</p> <p><CBU_USER password> password for CBU user account</p> <p>[-quiet] Do not display information</p>

install_cbu	Creates the CBU database for production and populates reference tables	<p><CBU></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CBU_ADMIN login> login for CBU admin account</p> <p><CBU_ADMIN password>: password for CBU admin account</p> <p><CBU_USER login> login for CBU user account</p> <p><CBU_USER password> password for CBU user account</p> <p>[-quiet] Do not display information</p>
create_cbu_structure	Creates a demo CBU database structure	<p><CBU></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CBU_ADMIN login> login for CBU admin account</p> <p><CBU_ADMIN password>: password for CBU admin account</p> <p>[-quiet] Do not display information</p>

Synchronizer Connector Administration Tools

You use the following administration tools to administrate the Synchronizer Connector:

- `agentstart`
- `agentadm`

These tools are located in `<home_dir>/bin`.

agentstart Syntax

```
agentstart <connector_name>
```

PARAMETERS	DESCRIPTION
<code><connector_name></code>	Name of the synchronizer connector to start

agentadm Syntax

```
agentadm help | <host> <port> <command> [<parameters>]
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<host>	Specifies the agent host
<port>	Specifies the agent administration port
<command>	Administration command

COMMAND	DESCRIPTION	PARAMETERS	RETURN VALUES
start	Resumes the execution of the connector. Use this command to restart the connector after a stop command.	none	-200 AGENT START
stop	Stops the connector. The connector no longer processes inbound and outbound messages	none	-120 AGENT STOP
shutdown	Terminates the connector	none	-100 AGENT SHUTDOWN
setparameter	Changes the settings of the connector remotely. Use the following syntax: setparameter <parameter>=<value>	AGENTNAPPING AGENTLOOPING ONE_EXECUTION NBRROW QFILL MAXNUMBERRETRY RETRYDELAY	-510 AGENT SET<parameter name>=<parameter value>

getparameter	Restarts the current value of the parameter	AGENTNAPPING AGENTLOOPING ONE_EXECUTION NBRROW QFILL MAXNUMBERRETRY RETRYDELAY Any parameter defined in the properties files	-610 AGENT GET<parameter name>=<parameter value>
getmode	Displays the current running mode	none	-210 AGENT NORMAL -220 AGENT REDUCE -230 AGENT PAUSE
getstatus	Displays the current status	none	130 AGENT NORMAL -140 AGENT REDUCE -150 AGENT PAUSE
force	Forces the connector to run in the specified mode	NORMAL REDUCE RECOVER	-410 AGENT SET MODE=NORMAL -410 AGENT SET MODE=REDUCE -410 AGENT SET MODE=RECOVER For MODE=RECOVER, an additional parameter must be provided: FILTER <FilterName>
info	Displays information about the connector	none	[AGENTNAME:xxx,NBTHRE ADS:xxx,FILL:xxx,NBROW:xx, NAPPING:xxx,LOOPING:xxx, VERSION:xxx,BUILD:xx x,OS:xxx]
kill	Forces shutdown	none	none
list	Returns all the profiling information	none	none
purge	Purges the profiling information	none	none

stat	Returns the profiling information of the last profiling element	none	For each message type, displays the following information: <ul style="list-style-type: none">- name of message type- min processing time- max processing time- total processing time- mean processing time- deviation processing time- nb number of messages- probability the percentage of successful messages processed- first timestamp of the first message processed- last timestamp of the last message processed- throughput number of messages processed per minute
version	Returns the version of the connector	none	none

OSS Connector Administration Tools

You use the following administration tools to administrate the OSS connector:

- `ossstart`
- `ossadm`

These tools are located in `<home_dir>/bin`.

ossstart Syntax

```
ossstart help | <connector>
```

PARAMETERS	DESCRIPTION
<connector>	Name of the connector to start.

ossadm Syntax

```
ossadm help | <host> <port> <command>
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<host>	Specifies the connector host
<port>	Specifies the connector port
<command>	Administration command

COMMAND	DESCRIPTION	PARAMETERS	RETURN VALUES
shutdown	Terminates the ossagent	none	none
kill	Forces shutdown	none	none
stat	Returns the profiling information of the last profiling element	none	For each message type, displays the following information: - name of message type - min processing time - max processing time - total processing time - mean processing time - deviation processing time - nb number of messages - probability the percentage of successful messages processed - first timestamp of the first message processed - last timestamp of the last message processed - throughput number of messages processed per minute

NOTIFYORG Administration Tool

You use the `notifyorg` tool to force the synchronization of organization information in the CBU.

This administration tool is located in `<home_dir>/bin`.

Configuring the NOTIFYORG Administration Tool

After installation, you can change the settings of the `notifyorg` administration tool.

You use the `notifyorg.properties` configuration file to set the properties of the administration tool. This file is located in

`<home_dir>/config/notifyorg/nmycfg/notifyorg`.

Configuring the `notifyorg` Administration tool involves:

- Specifying the Agent Name and password
- Specifying the data block size

To modify the settings of the NOTIFYORG administration tool

- 1 Go to `<home_dir>/config/notifyorg/nmycfg/notifyorg`.
- 2 Open `notifyorg.properties`.
- 3 Change the following settings:

SETTING	DESCRIPTION
AgentName	The name of the agent to connect to the BLM.
AgentPassword	The corresponding password
DATABLOCKSIZE	<p>Default number of items to handle before submitting the data.</p> <p>The NOTIFYORG administration tool processes the data in sets corresponding to the specified size until done.</p> <p>This value can be overridden when running the administration tool.</p>

- 1 Save your changes.

Example of notifyorg.properties

Agent Name and password	AgentName=agent AgentPassword=agent
Data Block Size	# DATABLOCKSIZE DataBlockSize=1000

notifyorg Syntax

notifyorg [<optional parameters>]

PARAMETERS	DESCRIPTION
help	Displays help for the tool
[<optional parameters>]	Optional parameters

COMMAND	DESCRIPTION	PARAMETERS
notifyorg	Processes the synchronization of organizations	[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.

Approval Sequencer Administration Tools

You use the following administration tools to administrate the Approval Sequencer:

- `approvalsequencer`
- `approvalsequenceradm`

These tools are located in `<home_dir>/bin`.

approvalsequencer Syntax

`approvalsequencer`

The configuration parameters are in the `agent.properties` configuration file. This file is located in `<home_dir>/config/approvalsequencer`.

approvalsequenceradm Syntax

```
approvalsequenceradm help | <host> <port> <command>
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<host>	Specifies the Approval Sequencer host
<port>	Specifies the Approval Sequencer administration port
<command>	Administration command

COMMAND	DESCRIPTION	PARAMETERS	RETURN VALUES
shutdown	Terminates the Approval Sequencer	none	none
kill	Forces shutdown	none	none

CustDim Loader Administration Tools

You use the following administration tools to administrate the CustDim Loader:

- `custdimloader`
- `custdimadm`

These tools are located in `<home_dir>/bin`.

Configuring the CustDim Loader Database Connection

You can define the CustDim loader database connection parameters. By default, you enter the CustDim connection parameters for the CID and CBU during installation. However, you may need to change the connection parameters after installation.

The CustDim loader uses the following configuration files:

- `instance_route.properties` configuration file to access the CID.
- `cbu_instance.properties` configuration file to access the CBU.

These files are located in

`<home_dir>/config/custdim/nmycfg/dal/instances`.

To configure the database connection

- 1 Go to `<home_dir>/config/CustDim/nmycfg/cid2cbu`.
- 2 Do one of the following:
 - To change the connection to the CID, open `instance_route.properties`.
 - To change the connection to the CBU, open `cbu_instance.properties`
- 3 Enter the following:
 - **DRIVER:** enter the name of the driver to use
 - **URL:** enter the location of the database
 - **USER:** enter the login
 - **PASSWORD:** enter the password
- 4 Save your changes.

custdim Syntax

```
custdimloader CustDim
```

PARAMETERS	DESCRIPTION
CustDim	Name of the connector to start

custdimadm Syntax

```
custdimadm help | <host> <port> <command> [<parameters>]
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<host>	Specifies the agent host
<port>	Specifies the agent administration port
<command>	Administration command

COMMAND	DESCRIPTION
start	Start the connector. If no data is found then the connector is automatically stopped.
shutdown	Shuts down the connector. This command waits for the completion of the current input file and all of its associated processing.
pause	Pause the execution of the connector. This command forces the connector to pause the extraction of new messages from all its inbound queues (invoice file for the splitter, message for the transformer and message for the CID loader and CBU loader)
resume	Resume the execution of the connector. This command forces the connector to resume the extraction of new messages from all its inbound queues.
kill	Kill the connector process. This command stops the connector immediately. It does not wait for the completion of all internal components and processing. This may lead to loss of data.

stat	<p>For each message type, displays the following information:</p> <ul style="list-style-type: none">- name of message type- min processing time- max processing time- total processing time- mean processing time- deviation processing time- nb number of messages- probability the percentage of successful messages processed- first timestamp of the first message processed- last timestamp of the last message processed- throughput number of messages processed per minute
------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

SmartLink (ISF) Message Cache Administration Tool

You use the following administration tools to administrate the SmartLink (ISF) Message Cache:

- `messagecacheAdminTool`

This tool is located in `<home_dir>/bin`.

Configuring the SmartLink (ISF) Message Cache Administration Tool

After installation, you can change the settings of the SmartLink (ISF) Message Cache administration tool.

You use the `persistence_tools.properties` configuration file to set the properties of the SmartLink (ISF) Message Cache administration tool. This file is located in `<home_dir>/lib/admin/persistence`.

Configuring the SmartLink (ISF) Message Cache Administration tool involves:

- Specifying the JDBC driver
- Specifying the database URL
- Specifying the encoding
- Specifying the data block size

To modify the settings of the SmartLink (ISF) Message Cache Tool

- 1 Go to `<home_dir>/lib/admin/persistence`.
- 2 Open `persistence_tools.properties`.
- 3 Change the following settings:

SETTING	DESCRIPTION
DRIVER	Class name of the ODBC driver to use
URL	URL of the SmartLink (ISF) Message Cache database
ENCODING	Encoding to use
DATABLOCKSIZE	<p>Default number of items to handle before submitting the data.</p> <p>The SmartLink (ISF) Message Cache administration tool processes the data in sets corresponding to the specified size until done.</p> <p>This value can be overridden when running the administration tool.</p>

Example of persistence_tools.properties for Oracle

JDBC driver	<pre># driver: This is the class name of the jdbc driver driver=oracle.jdbc.driver.OracleDriverdriver=oracle.jdbc.driver.OracleDriver</pre>
URL of the database	<pre># url:This is url for the database (typically jdbc:oracle:oci8:@OracleCidName) url=jdbc:oracle:oci8:@ISF</pre>
<p>Internal settings DO NOT MODIFY</p> <p>If you need to modify these settings, you must reinstall and enter the new values during installation.</p>	<pre># SQL Base Path SqlBasePath=<home_dir>/data/oracle/ # LOG Path LogPath=<home_dir>/var/logs/admin/ # OUTPUT PATH OutPath=<home_dir>/var/output/admin/ # VERSION STRING Version=X.X.X.XXX</pre>
Encoding	<pre># ENCODING Encoding=ISO-8859-1</pre>
Data Block Size	<pre># DATABLOCKSIZE DataBlockSize=10000</pre>

messagecacheAdminTool Syntax

```
messagecacheAdminTool help | <command> <parameters> [<optional parameters>] <help>
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<command>	Administration command
<parameters>	Specifies the parameters for the command
[<optional parameters>]	Optional parameters
<command> help	Displays help for the command

COMMAND	DESCRIPTION	PARAMETERS
create_messagecache_structure	Creates a Message Cache database structure	<db_instance> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <login> login for the CID admin account <password> password for the CID admin account <cachename> name of the message cache [--custom:sqlfile] name of custom sql file to run [--quiet] Do not display information

drop_messagecache_structure	Deletes the Message Cache structure in the specified database instance	<p><db_instance></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><login> login for the CID admin account</p> <p><password> password for the CID admin account</p> <p><cachename> name of the message cache</p> <p>[--custom:sqlfile] name of custom sql file to run</p> <p>[--quiet] Do not display information</p>
purge_messages	Purges the messages stored in the Message Cache	<p><db_instance></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><login> login for the CID admin account</p> <p><password> password for the CID admin account</p> <p><cachename> name of the message cache</p> <p>[--days:days] The age of the messages to keep in days. For example, entering 7 purges all of the requests that were not created during the last week.</p> <p>[--custom:sqlfile] name of custom sql file to run</p> <p>[--quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>

Report Manager Administration Tool

You use the `savedreportAdminTool` administration tool to purge the files created by the WFS FileManager.

This tool is located in `<home_dir>/bin`.

Configuring the Report Manager Administration Tool

You can define different properties of the Report Manager.

The Report Manager uses the `options.properties` configuration file to set its properties. This file is located in `<home_dir>/config/reportmanager/nmycfg/wfs`.

You can configure the following:

- The length of time a file can be locked
- Location of the directories

To set the lifespan of file locks

- 1 Go to `<home_dir>/config/reportmanager/nmycfg/wfs`.
- 2 Open `options.properties`.
- 3 Enter the following:
 - For `filemgr.lockfiles.admin.timetolive`, enter the lifespan of the lock file in minutes. When the lock file is older than the specified length of time, the administration tool may purge it.
 - For `filemgr.lockfiles.ignoreifolderthan`, enter the age of the lock file in seconds. When the lock file is older than the specified amount of time, the Report Manager ignores it.
- 4 Save your changes.

To specify the root directories

- 1 Go to `<home_dir>/config/reportmanager/nmycfg/wfs`.
- 2 Open `options.properties`.
- 3 For permanent report files, enter the full path. Use the syntax:

```
filemgr.rootdir.<repositoryCode>=full_path
```

where <repositoryCode> is the code to use in report manager APIs

- 4 For temporary report files, enter the full path and specify the directory type. Use the syntax:

```
filemgr.rootdir.<repositoryCode>=full_path
```

```
filemgr.rootdir.<repositoryCode>.type=cache
```

- 5 Save your changes.

Example of options.properties

The number of minutes a lock file is allowed to exist	<pre>filemgr.lockfiles.admin.timetolive=1440</pre>
The number of seconds before the Report Manager ignores the lock file	<pre>filemgr.lockfiles.ignoreifolderthan=120</pre>
Declaration of cache1	<pre>filemgr.rootdir.cache1=C:/mydir/var/wfs/cache filemgr.rootdir.cache1.type=cache</pre>
Declaration of cache2	<pre>filemgr.rootdir.cache2=C:/mydir/cache filemgr.rootdir.cache2.type=cache</pre>
Declaration of permanent file directories	<pre>filemgr.rootdir.permanent1=C:/mydir/var/wfs/permanent filemgr.rootdir.permanent2=C:/mydir/var/wfs/permanent</pre>

savedreportAdminTool Syntax

```
savedreportAdminTool help | <command> <repositoryCode>
[<keyCode>]
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<command>	Administration command
<repositoryCode>	The file directory specified in options.properties
<keyCode>	The Key code of the directory

COMMAND	DESCRIPTION	PARAMETERS
purgeCorrupted	Removes the reports not correctly written	repositoryCode
purgeAllReports	Removes all the reports for a given key	repositoryCode keyCode
purgeCache	Removes all the reports in a cache repository	repositoryCode keyCode

Job Service Administration Tools

You use the following administration tools to administrate jobs:

- `jobAdminTool`
- `jobprocessorstart`
- `jobprocessoradm`

These tools are located in `<home_dir>/bin`.

Configuring the Job Service Administration Tool

After installation, you can change the settings of the Job Service administration tool.

This tool shares the `job_tools.properties` configuration file with the CID Administration tool. You use this file to set the properties of the Job Service administration tool. This file is located in `<home_dir>/lib/admin/jobs`.

Configuring the CID Administration tool involves:

- Specifying the JDBC driver
- Specifying the database URL
- Specifying the encoding
- Specifying the data block size

When configuring this tool, keep in mind that you can override the data block size when running the tool.

Example of `job_tools.properties`

Example of `job_tool.properties` for Oracle

JDBC driver	<pre># driver: This is the class name of the jdbc driver driver=oracle.jdbc.driver.OracleDriverdriver=oracle.jdbc.driver.OracleDriver</pre>
URL of the database	<pre># url:This is url for the database (typically jdbc:oracle:oci8:@OracleCidName) url=jdbc:oracle:oci8:@CID</pre>
Internal settings DO NOT MODIFY If you need to modify these settings, you must reinstall and enter the new values during installation.	<pre># SQL Base Path SqlBasePath=<home_dir>/data/oracle/ # LOG Path LogPath=<home_dir>/var/logs/admin/ # OUTPUT PATH OutPath=<home_dir>/var/output/admin/ # VERSION STRING Version=X.X.X.XXX</pre>
Encoding	<pre># ENCODING Encoding=ISO-8859-1</pre>
Data Block Size	<pre># DATABLOCKSIZE DataBlockSize=10000</pre>

To modify the settings of the Job Service administration tool

- 1 Go to `<home_dir>/lib/admin/jobs`.
- 2 Open `job_tools.properties`.
- 3 Change the following settings:

SETTING	DESCRIPTION
DRIVER	Class name of the ODBC driver to use
URL	URL of the database
ENCODING	Encoding to use
DATABLOCKSIZE	<p>Default number of items to handle before submitting the data.</p> <p>The tool processes the data in sets corresponding to the specified size until done.</p> <p>This value can be overridden when running the Job Service administration tool.</p>

jobAdminTool Syntax

```
jobAdminTool help | <command> <parameters> [<optional
parameters>] <help>
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<command>	Administration command
<parameters>	Specifies the parameters for the command
[<optional parameters>]	Optional parameters
<command> help	Displays help for the command

COMMAND	DESCRIPTION	PARAMETERS
fail_in_progress_jobs	Changes the status of IN PROGRESS jobs to FAILED	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><job-type-code> code of the job types to purge</p> <p><NBhours> the minimum number of hours the job has had the IN PROGRESS status. 0 to change all IN PROGRESS jobs.</p> <p>[<service-code>] the service code of the jobs to change</p> <p>[<error-reason-code>] the reason code of the error</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>

purge_jobs	Purges the jobs which have a final state (DONE, CANCELLED, FAILED) for the specified length of time	<p><CID></p> <ul style="list-style-type: none">• Oracle: <instance alias>• DB2: <database alias>• SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><job-type-code> code of the job types to purge</p> <p><days> the minimum number of days the job has been DONE, CANCELLED, or FAILED</p> <p>[<job-status>...] the job status of the jobs to purge if other than the default final statuses of DONE, CANCELLED, and FAILED</p> <p>[<service-code>] the service code of the jobs to purge</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>
------------	-----------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

reset_in_progress_jobs	Changes the status of IN PROGRESS jobs to TO BE PROCESSED	<p><CID></p> <ul style="list-style-type: none"> • Oracle: <instance alias> • DB2: <database alias> • SQL Server: <database host> [:<port>] If no port is specified, the tool uses the default SQL server port <p><CID_ADMIN login> login for the CID admin account</p> <p><CID_ADMIN password> password for the CID admin account</p> <p><job-type-code> code of the job types to purge</p> <p><NBhours> the minimum number of hours the job has had the IN PROGRESS status (0 to change all IN PROGRESS jobs)</p> <p>[<service-code>] the service code of the jobs to change</p> <p>[-quiet] Do not display information</p> <p>[<datablocksize:####>] specifies the number of items to process before committing. Enter 0 to override default settings.</p>
------------------------	-----------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

jobprocessorstart Syntax

jobprocessorstart <job_type_name>

PARAMETERS	DESCRIPTION
<job_type_name>	<p>Name of the job type to process.</p> <p>The job type is found in <home_dir>/config/jobprocessors</p>

jobprocessoradm Syntax

```
jobprocessoradm help | <host> <port> <command>
```

PARAMETERS	DESCRIPTION
help	Displays help for the tool
<host>	Specifies the host
<port>	Specifies the administration port
<command>	Administration command

COMMAND	DESCRIPTION	PARAMETERS	RETURN VALUES
kill	Forces shutdown	none	none
shutdown	Terminates the job service	none	none
pause	Pauses the execution of the job service	none	none
resume	Resumes the execution of the job service Use this command to restart the job service after a stop command.	none	none

Index

A

- ActionManager
 - and persistent ActionManagers • 47
- Administering
 - architecture • 23
 - basic tasks • 26
 - location of configuration files • 24
 - managing the Approval Sequencer • 29
 - managing the OSS Connector • 28
 - managing the Synchronizer Connector • 27
 - overview • 22, 34
- Administration tools
 - about • 122
 - agentadm • 27, 36, 148
 - agentstart • 27, 147
 - approvequencer • 29, 155
 - approvequenceradm • 29, 156
 - cbuAdminTool • 143
 - cidAdminTool • 125
 - jobAdminTool • 171
 - jobprocessoradm • 174
 - jobprocessorstart • 173
 - messagecacheAdminTool • 161
 - ossadmin • 28, 152
 - ossstart • 28, 151
 - savedreportAdminTool • 75, 167
- agent.properties Configuration File
 - location • 29
 - using • 29
- agentadm Administration Tool
 - about • 27, 35
 - location • 35
 - syntax • 148
 - using • 27, 36
- agentstart Administration Tool
 - about • 27
 - location • 27
 - syntax • 147
 - using • 27
- Approval Sequencer
 - administration tool syntax • 155, 156
 - configuring • 29
 - managing • 29
 - starting • 29
 - stopping • 29
- approvequencer Administration Tool
 - about • 29
 - location • 29
 - syntax • 155

- using • 29
- approvalequenceraadm Administration Tool
 - about • 29, 155
 - location • 29
 - using • 29, 156

- Auditing
 - about • 112
 - and User Events • 112
 - reports • 114

B

- BLM (Business Logic Manager)
 - and application reference data • 53
- BLM Objects
 - refreshing • 53
 - types • 53

C

- CBU Reports
 - and the WFS • 75
 - managing • 75
- cbu_tools.properties Configuration File
 - about • 143
 - examples • 144
 - location • 143
 - using • 143
- cbuAdminTool Administration Tool
 - about • 143
 - configuring • 143
 - create_cbu_users command • 145
- CID (Customer Interaction Datastore)
 - administration tool • 123, 125
- cid_tools.properties Configuration File
 - about • 123
 - example • 124
 - location • 123
 - using • 123
- CID2CBU Loader
 - configuring • 133, 134
 - optimizing • 137
 - setting database connection • 136
 - starting • 30
 - stopping • 30
- cid2cbuloader Administration Tool
 - commands • 145
 - configuring • 143
- cid2cbuloader.properties Configuration File
 - about • 134
 - location • 134
 - using • 134, 137

- cid2cbuloaderadm Administration Tool
 - using • 30
- cidAdminTool Administration Tool
 - about • 123
 - admin_notifications command • 60, 61, 62, 64, 67
 - admin_requests command • 39, 40, 41, 125
 - configuring • 123, 124
 - create_cid_users command • 125
 - create_demo_cid_test command • 125
 - drop_demo_cid_structure command • 125
 - generate_logon_stats command • 114, 125
 - install_cid command • 125
 - purge_notifications command • 63
 - purge_persistent_shopping_action_managers command • 48
 - purge_requests command • 43, 125
 - purge_trouble_tickets command • 49, 125
 - syntax • 125

- Configuration Files
 - and WAR files • 24
 - job_tools.properties Configuration File • 169
 - location of • 24

- Configuring
 - Logger • 87
- CustDim Loader
 - about • 69
 - handling errors • 73
 - log entries • 71, 72, 73
 - monitoring • 71, 72, 73
 - pausing • 69
 - running • 69
 - starting • 31
 - stopping • 31

D

- Debugging
 - logs • 78

E

- Event Codes
 - about • 107

I

instance_route.properties Configuration File

- about • 136
- configuring • 136
- location • 136

instance_route_cbu.properties

- Configuration File
- about • 136
- location • 136
- using • 136

J

Job Processor

- administration tools • 168, 173, 174
- cancelling • 45
- purging • 44
- resubmitting • 45
- starting • 32
- stopping • 32**

job_tools.properties Configuration File

- about • 169
- location • 169
- using • 170

jobAdminTool Administration Tool

- about • 168
- commands • 171
- configuring • 169, 170
- location • 168

L

log4j.properties Configuration File

- about • 87
- configuring drivers • 93, 94, 96
- configuring events • 91
- configuring output • 99, 102
- default driver settings • 95, 97, 102
- example • 103
- location • 87

LOG4J.PROPERTIES

CONFIGURATION FILE

CONFIGURING OUTPUT • 101

Logger

- configuring • 87, 89
- configuring events to log • 81, 91
- configuring output • 93, 99, 102, 108
- modules • 83
- rolling output driver • 94, 95
- severity levels • 82
- standard output driver • 93

- time stamp log file output driver • 96, 97, 98

LOGGER

CONFIGURING OUTPUT • 101

logger.properties Configuration File

- example • 103
- location • 87
- using • 89

Logs

- about • 78
- components • 80, 81, 82, 83
- contents of • 80
- event types • 80, 81
- examples • 103, 104
- severity levels • 82
- using • 84, 85, 86
- using with supervision platform • 105, 106, 107

M

messagecache_tools.properties

Configuration File

- about • 161
- location • 161
- using • 161

messagecacheAdminTool Administration Tool

- about • 161
- configuring • 161
- create_messagecache_structure command • 163
- drop_messagecache_structure command • 163
- purge_messages command • 163
- syntax • 163
- using • 51, 52

Monitoring

- about • 78
- using a supervision platform • 105
- using the logger • 78, 84

N

Notifications

- about • 59, 66
- and notifyorg tool • 67, 68
- cancelling • 62
- displaying • 60, 61
- in the CID • 59
- purging • 63
- resubmitting • 64

- statuses • 60
- synchronization of • 67, 68
- notifyorg Administration Tool
 - about • 66, 153
 - configuring • 153, 154
 - location • 153
 - syntax • 154
 - using • 67, 68
- notifyorg.properties Configuration File
 - about • 153
 - examples • 154
 - location • 153
 - using • 153

O

- OSS Connectors
 - running • 28
 - stopping • 28
- oss2sync Queue
 - about • 37
 - purging • 37
- ossadmin Administration Tool
 - about • 28
 - location • 28
 - using • 28, 152
- ossstart Administration Tool
 - about • 28, 151
 - location • 28
 - using • 28, 151

P

- Persistent Action Managers
 - about • 47
 - and shopping cart templates • 47
 - and Shopping Carts • 47
 - categories • 48
 - in the CID • 47
 - purging • 48
- policy.properties Configuration File
 - about • 53
 - location • 53
 - using • 53
- Purging
 - persistent action managers • 48
 - requests • 43
 - Synchronizer Queue • 37
 - trouble tickets • 49

R

- Reference Data

- about • 53
- and the BLM cache • 53
- limits of reloading feature • 54
- reloading • 54, 55
- reportmanageradm Administration Tool
 - about • 165
 - configuring • 165
 - location • 165
 - syntax • 167
 - using • 75, 76

Requests

- about • 38
- canceling • 40, 41
- displaying contents of • 39, 40
- in the CID • 38
- purging • 43
- resubmitting • 42
- status codes • 39

S

- Sessions
 - managing • 120
 - serialization • 120
- Shared Directories
 - /share directory • 119
 - /var directory • 119
 - about • 119
- Shopping Cart Templates
 - about • 47
 - and persistent action managers • 47
- Shopping Carts
 - and Persistent Action Managers • 47
- SmartLink (ISF) Administration Tools
 - agentadm administration tool • 148
 - agentstart administration tool • 147
- SmartLink (ISF) Message Cache
 - about • 50
 - creating • 51
 - dropping • 51
 - managing • 52
- Starting
 - Approval Sequencer • 29
 - OSS Connector • 28
 - Synchronizer Connector • 27
- Status
 - connector runtime statuses • 36
 - request statuses • 39
- Stopping
 - Approval Sequencer • 29
 - OSS Connector • 28

Synchronizer Connector • 27

Synchronizer Connectors

checking current mode • 36

checking status • 36

configuring • 27

managing • 27, 35

modes • 36

parameters • 36

purging default queue • 37

starting • 27

statuses • 35, 36

stopping • 27, 148

T

Trouble Tickets

about • 49

managing • 49

purging • 49

U

User Events

about • 112

activating • 113

disactivating • 113

generating reports • 114

W

WAR File

about • 24

and configuration files • 24