

Porting Guide for Consistency Checker

Ericsson Dynamic Activation

GUIDE

Copyright

© Ericsson AB 2017. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



Contents

1	Introduction	1
1.1	Purpose and Scope	1
1.2	Target Groups	1
1.3	Typographic Conventions	1
1.4	Prerequisites	2
2	Software Porting	3
2.1	AS specific deployment descriptors	3
2.2	System property	4
2.3	Java Message Service	4
2.4	Security	4
2.5	Deployment	5
2.6	Logging	5
3	Documentation Porting	7
	Glossary	9
	Reference List	11





1 Introduction

This section is an introduction to this document. It contains information about the prerequisites, purpose, scope, and target group for the document. This section also contains explanations of typographic conventions used in this document.

1.1 Purpose and Scope

This document is a generic porting guideline for deploying the Consistency Checker on other environment than the reference environment, which is Glassfish Server v3.1.2, see Reference [4].

Another purpose of this document is to instruct the user how to port the documentation required for further work with the software.

1.2 Target Groups

The target group for this document is a user responsible for porting the application to the target deployment environment.

1.3 Typographic Conventions

This section describes the typographic conventions commonly used, see Table 1.

Table 1 Typographic Conventions

Convention	Description	Example
Output Information	Text displayed by the system is shown in monospaced font.	System awaiting input
User Input	A command that must be entered in a Command Line Interface (CLI) exactly as written is shown in bold monospaced font.	cd \$HOME
Command Variables	Command variables included in a command, are enclosed by angle brackets <>. They are shown in bold, italic monospaced font.	<home_directory>



Convention	Description	Example
System Elements	Command and parameter names, program names, path names, URLs, and directory names are shown in monospaced font. Slash (/) is used throughout the Consistency Checker documentation. It might differ in practice depending on the target OS environment.	The files are located in <code>/etc/opt/ericsson</code>
Code Examples	Code examples are shown in monospaced font. The backslash (\) is used to show where long lines are split.	<pre>private Map<String, AttributeRule> attributeMap = \ new HashMap<String, AttributeRule>();</pre>

1.4 Prerequisites

The following are the prerequisites to make full use of this document:

- Adequate knowledge of a target operating system - Windows, Linux™ or UNIX™.
- Adequate knowledge of Java™ Enterprise Edition (JEE) and the target application server (AS). Standard JEE concepts will not be explained here.

To get an overview of the Consistency Checker and deployment scenarios, see *Function Specification Consistency Checker*, Reference [1].



2 Software Porting

The Consistency Checker application is a portable JEE 6 compliant enterprise application. This section describes the steps that has to be considered when deploying the Consistency Checker on another AS than Glassfish Server.

2.1 AS specific deployment descriptors

AS specific deployment descriptors can be required, depending on the target AS.

In the Consistency Checker application there are four applications with embedded modules that may need AS specific deployment descriptors:

- Consistency Checker Management Enterprise Application - the `APP_CC_Management.ear` archive file.
 - `bootstrap.war`
 - `controller.jar`
 - `controller-gui.war`
 - `verification.war`
- Consistency Checker Real-time Enterprise Application - the `APP_Realtime.ear` archive file.
 - `EJB_Realtime.jar`
 - `Realtime-bootstrap.war`
- Consistency Checker Offline Enterprise Application - the `APP_Offline.ear` archive file.
 - `EJB_Offline.jar`
 - `Offline-bootstrap.war`
- Consistency Checker Deviation Enterprise Application – the `APP_Deviation.ear` archive file.
 - `deviation.jar`

Depending on the target AS it might be required that the AS specific deployment descriptors are embedded in respective archive.



Note: If the AS supports the standard deployment API, based on Java Specification Request (JSR) 88, it could be used with the AS specific deployment descriptors. The standard deployment API enables any JEE application to be deployed by any deployment tool that uses the deployment APIs onto any JEE compatible environment.

2.2 System property

The target AS should be configured so that the Consistency Checker application has access to the system property:

```
com.ericsson.consistency.CHECKER_HOME.
```

The value of this property shall be the absolute path to the `<CheckerHomeDir>`.

2.3 Java Message Service

The Consistency Checker application utilizes the Java Message Service (JMS). This requires the target AS to be configured to provide required JMS resources.

Table 2 JMS target configuration

Type	JNDI
<code>javax.jms.QueueConnectionFactory</code>	<code>jms/consistency/QueueConnectionFactory</code>
<code>javax.jms.Queue</code>	<code>jms/consistency/order_execution</code>
<code>javax.jms.Queue</code>	<code>jms/consistency/event</code>

To balance the rate at which messages are consumed in the queues, the JMS service must be setup with flow control to throttle the incoming messages.

By applying this JMS service behavior, and because the JMS queues use the same connection factory, all queues will have the same behavior, namely flow control.

2.4 Security

The Consistency Checker controller depends on a security realm called `DefaultCCRealm`. The target AS must be configured to provide a realm with this name.

Users/accounts that are stored in the realm, which shall have access to the Consistency Checker management Graphical User Interface (GUI), must be assigned to at least one role.

The default role for CC is `ccuser`. It is possible to add new roles.

The Consistency Checker management GUI uses form based access control. This might require that the login HTML form is updated



according to the target AS. The following might need to be updated in the `APP_CC_Management.ear/controller-gui.war/login.jsp` file:

- The value of the action attribute in the form HTML tag should be the URL to the target AS form based security handler. Current value of the action attribute is `j_security_check`.
- The form based security handler in the target AS might require specific names on the login form input fields. Current names are `j_username` and `j_password`.

2.5 Deployment

Deploy the `APP_CC_management.ear`, `APP_Offline.ear`, `APP_Deviation.ear` and the `APP_Realtime.ear` (if the Online feature shall be used) files according to the target AS deployment instructions.

2.6 Logging

Consistency Checker uses by default standard logging via the Commons-logging API. If other logging frameworks than the standard logging are wanted, see Reference [6].

Note: If Log4J library is available in the application server classloader, it will take precedence over standard logging. For more information, see Reference [5].





3 Documentation Porting

In order to facilitate upgrade, expansion and maintenance of the deployed application, it is recommended to produce target Application Server specific documentation and script. The product line produces a set of such information for the Glassfish Server, which can be used as reference.

- Installation Instruction

This document describes the installation procedure. For example, for more information about installation on Glassfish Server, see *Installation Instruction for Consistency Checker on Glassfish Server Open Source Edition*, Reference [2].

- Installation and Uninstallation scripts

The scripts automate installation and uninstallation procedures to avoid human mistake during installation and uninstallation. For details on how to find the example scripts for Glassfish Server, see *System Administrators Guide for Consistency Checker*, Reference [3].

- System Administrators Guide

This document describes the information necessary for a system administrator to maintain the system. For example, for more information about system administration for Glassfish Server, see *System Administrators Guide for Consistency Checker*, Reference [3].





Glossary

AS

Application Server

GUI

Graphical User Interface

JEE

Java Enterprise Edition

JMS

Java Message Service

JSR

Java Specification Request





Reference List

Ericsson Documents

- [1] *Function Specification Consistency Checker*, 21/155 17-CSH 109 628 Uen
- [2] *Installation Instruction for Consistency Checker on Glassfish Server Open Source Edition*, 5/1531-CSH 109 628 Uen
- [3] *System Administrators Guide for Consistency Checker* , 5/1543-CSH 109 628 Uen

3PP documents

- [4] *Glassfish Server 3.1.2 Installation Guide*, <https://glassfish.java.net/docs/3.1.2/installation-guide.pdf>
- [5] *Apache log4j*, <http://logging.apache.org/log4j/>
- [6] *Commons logging*, <http://commons.apache.org/proper/commons-logging/>