Ericsson Catalog Manager and Ericsson Order Care

Realize Higher Consistency for Faster Time-to-Revenue

SUF Installation and Configuration Guide

1/1543-CSH 109 420/1C Uen Rev A

Contents

1 Introduction

This document provides an introduction to installing, configuring, and using the System Upgrade Framework (SUF).

1.1 Purpose and Scope

The purpose of this document is to provide information on how to install, configure, and use SUF. To perform the tasks in this document requires that you have system administration or application development experience.

1.2 Reader's Guideline

This section describes the version syntax covered in this document and any additional, required information.

Commands that you enter on the command line appear in courier font, such as the following:

```
svnadmin dump C:\SVN\myProject > C:\backupFolder\myProject.bak
```

Document names and sections within documentation are set in italics, such as the following:

For more information on making a copy of your project metadata, see the *Velocity Studio User Guide*, under *Velocity Studio User Interface* > *Common Actions Outside Service Designer*.

Note: To navigate the documentation, an arrow appears (>), which separates each hyperlink to be clicked.

2 SUF Overview

SUF is both a framework and a tool to simplify software upgrades. It is also a utility or a software management tool that provides a centralized control to install and manage software applications in a network. The key elements in SUF are as follows:

- A **node** is a logical group of hosts where an application is deployed. It helps to configure multiple hosts collectively. *Node-Type* is a class representing one type of nodes.
- A **package** includes the installables and playlists for an application. A package maps to a node type. You can upload *Packages* in the SUF repository and install it on a matching node-type.

- A **playlist** defines how to install a package. Playlists are part of the package and a package creator writes them. A playlist is a sequence of steps that the SUF server runs.

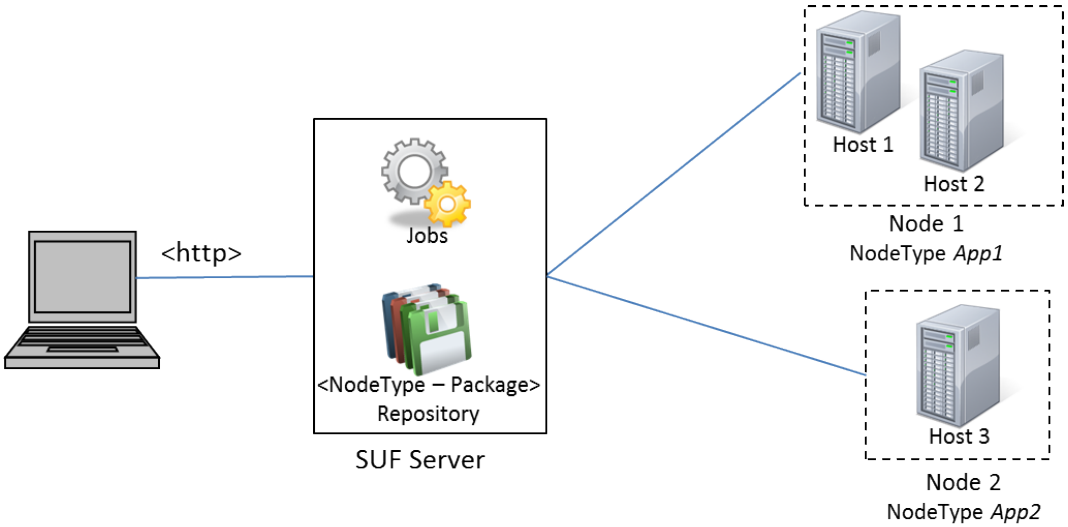

Figure 1 SUF elements

For more information, see the *SUF User Guide*.

2.1 SUF User Interface

The SUF graphical user interface (GUI) contains the following elements:

- Header
- Main menu

2.1.1 Header

The header appears at the top of each page and contains the following options:

.

- About
- Site Map
- User Guide
- Logout

About

This option contains the SUF version and copyright information. This dialog contains a link to the platform server information. After you click the link, the following information appears:

- Operating system/kernel details

- Virtual machine details
- Web server details
- SUF application details

Site Map

Click this option to view all menu options and available navigation paths in the user interface.

User Guide

Click this option to access the SUF GUI online user guide.

Logout

Click this option to exit the SUF application GUI. If there is no activity performed in the SUF application after 25 minutes, an alert message displays, indicating that the session will expire in five minutes. Clicking the **OK** button allows your session to remain active for another thirty minutes.

2.1.2 Main Menu

The SUF menu bar contains the following options:

- Administration
- Network
- Repository
- Job

Administration

This menu contains the following submenu options:

- **General** – Use this option to manage users and user classes.
- **Node Type** – Select this option to manage node types.
- **Audit Log** – Use this option to view audit logs.
- **Licensing** – Select this option to view license details.

Note: The Administration option is accessible only to a user who has his role set to *administrator*.

Network

This menu contains the **View** submenu option, which allows you to view, edit, download, and delete the network configuration file.

Repository

This menu contains the following submenu options:

- **Node Package** – Use this option to import, view, or delete a node package.
- **Library** – Select this option to import a library to the repository, or view or delete a library.

Job

This menu contains the following submenu options:

- **Current** – Use this option to create, view, download, abort, archive, and delete jobs.
- **Schedule** – Select this option to create, view, schedule, and delete job schedules.
- **Archive** – Use this option to view and delete archived jobs.

For more information, see the *SUF User Guide*.

2.2 Understand the Workflow

It is important to understand the workflow to operate SUF. The following figure shows the relationship between the different components in the application.

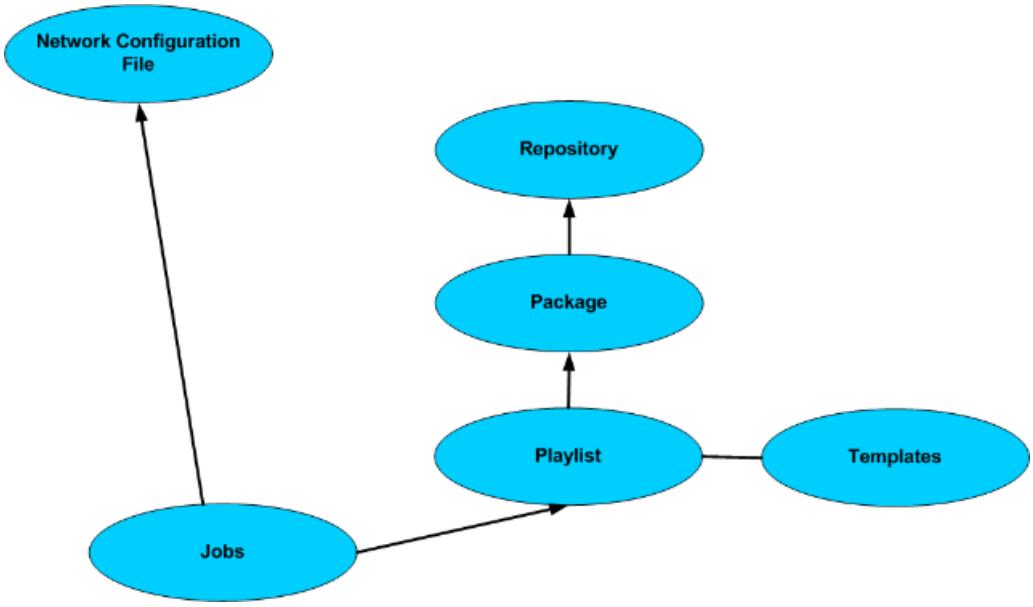

Figure 2 SUF workflow

A job requires a playlist, which is located in a package that is stored in the repository. The playlist may use a template file that holds some pre-populated values, which can be changed during job creation, if needed. The templates are also stored inside the package. Additionally, the job also must know on which host or hosts that it must be processed. Thus, it also has a connection to the applicable network list.

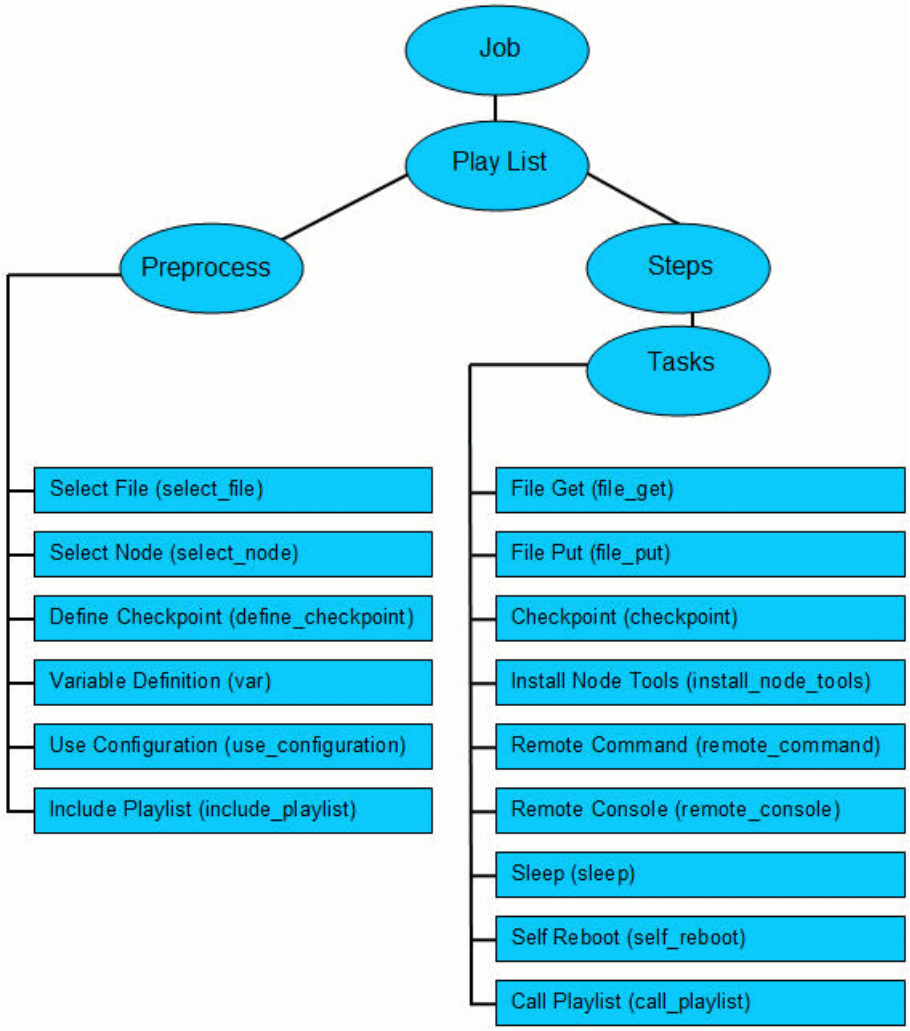

Figure 3 Workflow - Job relationship

Note: It is always important to verify that the required network lists are available.

2.3 AVM SUF Package

The Advanced Virtual Machine (AVM) SUF package includes the following files:

- AVM_Platform.rpm
- AVM_SUF.tar.gz
- Network.xml
- AVM_<*applicationName*>.rpm

2.3.1 AVM_Platform.rpm

This package contains the AVM product in RPM format. It includes DDL, scripts, Designer.jar, cwf.war, modules, and so on. Through FTP, upload the

OrderCare_AVM_<version>.rpm package from */var/lib/sufstorage/swtools* location on the SUF server.

2.3.2 AVM_SUF.tar.gz

This package includes SUF playlists. Playlists are the instructions for the SUF server on how to install or deploy the application across SUF clients. There is a playlist for each step in the installation process and a few top-level playlists. These top-level playlists invoke other playlists to perform specific tasks.

The names of the top-level playlists start with an underscore (_) and they appear at the top of the list in the SUF Web user interface (UI). The following table describes the top level and the corresponding playlists.

Top-Level Playlist	Description	Invoked Playlists
_NewInstallation.xml	This playlist is used for the new installation of both the platform and application.	• avm_rpm.xml • application_rpm.xml • license_distribution.xml • db_init_.xml • application_deploy.xml • catalog_import.xml • config_init.xml • db_upgrade.xml • application_sql.xml • jdbc_install.xml • jboss_start.xml • jndi_setup.xml • ngee_conf.xml • war_deploy.xml
_UpgradeInstallation.xml	This playlist is used to upgrade an installation of both platform and application.	• avm_rpm.xml • application_rpm.xml • application_deploy.xml • catalog_import.xml • db_upgrade.xml • application_sql.xml • jboss_start.xml • war_deploy.xml
_AVMUpgrade.xml	This playlist is used to upgrade the AVM platform only.	• avm_rpm.xml • db_upgrade.xml • jboss_start.xml • war_deploy.xml
_ApplicationUpgrade.xml	This playlist is used to upgrade the application only.	• application_rpm.xml • application_deploy.xml • catalog_import.xml • db_upgrade.xml • application_sql.xml • jboss_start.xml

2.3.3 Network.xml

The AVM SUF package includes a network.xml configuration file that contains the network setting to be imported and changed. This file is uploaded using SUF server's UI. The file provides a default network configuration. You must edit the configuration in SUF Web UI according to your network infrastructure.

The following describes the syntax of the default network configuration file, network.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
    <network name="CW_NETWORK">
      <node name="Jboss" type="CW">
          <host name="machine01">
             <access name="machine01_ssh"
                address="192.168.12.21"
                connection="SSH"
                username="userappserv"
                password="pwdapp.pwd"
             />
     <! -- Used to ftp the war to jboss. -->
             <access name="machine01_sftp"
               address="192.168.12.21"
               connection="SFTP"
               username="userappserv"
               password="app.pwd"/>
          </host>
      </node>
      <node name="Oracle" type="CW">
          <host name="oracle">
             <access name="oracle_ssh"
               address="192.168.15.112"
               connection="SSH"
               username="useroracle"
               password="ora.pwd"
             />
             <access name="oracle_sftp"
                address="192.168.15.112"
                connection="SFTP"
                username="useroracle"
                password="ora.pwd"
             />
          </host>
        </node>
      </network>
```

Note: The naming convention is important. For all JBoss hosts, the access name must be *jboss_ssh* and *jboss_sftp*. For the Oracle host, the access name must be *oracle_ssh* and *oracle_sftp*.

2.3.4 AVM_<*applicationName*>.rpm

This part of the package includes the application deployment.jar, config.xml, and application SQL files. Through FTP, upload the AVM_<*applicationName*>.rpm file from the */var/lib/sufstorage/swtools* location of your SUF server.

The application team needs to create an RPM file with the following structure and deliver it.

- <*applicationName*>
  - scriptData

- sql
  - main.sql
- <*applicationName*>.jar
- <*applicationName*_counters>.jar
- appScript.txt
- config.xml
- exportedCatalog.xml
- logback-template.xml
- smService.json
- smProfile.json

Notes:

- The naming convention is important. In the AVM_<*applicationName*>.rpm, the <*ApplicationName*> must be the same name as the deployment JAR file, which is metadata's internal name.
- The customer folder might not exist when the application is delivered. If the *config.xml* and *host_node_mapping.properties* files are customized, these files must be added to the SUF server to the */opt/OrderCare_SUF/Application/<ApplicationName>/customer* location through FTP.

For each application, you must have the following files, where **<***ApplicationName***>** represents the application name (for example, ECM):

- ***<ApplicationName>*/customer**
  This folder contains two files:
  - <*ApplicationName*>/customer/config.xml
  - <*ApplicationName*>/customer/host_node_mapping.properties

  These files are not included in the package delivery. The host_node_mapping.properties file is mandatory. Refer to the Configuration Files section for details.

  Uploading the Customer File

  These two files under <*ApplicationName*>*/customer* are from the customer. Therefore, the application team might not have these file when they create the application.rpm package, as it requires the exact configuration and host_node mapping from customer's network environment. Since these two files are not included in the application package, you need to send these files to the SUF server through FTP to this location:

  /opt/OrderCare_SUF/Applications/<ApplicationName>/customer

- ***<ApplicationName>*/sql/**
  This part of the package contains all the SQL files. These SQL files are required to be run before starting the runtime. These files contain some data population, trigger, index, or stored procedure creation etc. These SQL or DDL are not captured by database upgrade playlist. These SQL files are delivered by application team. However, these files can be

changed. The *main.sql* file is run first. If this file refers to other SQL files, use the following syntax to make the changes:

```
@@<Other_sql_file_1>.sql;
@@<Other_sql_file_2>.sql;
```

- **<*ApplicationName*>/config.xml**

  This file contains the application configuration. The configuration initialization playlist checks whether you have specified a configuration file. Otherwise, it uses the <*ApplicationName*>/config.xml file.

- **<*ApplicationName*>/<ApplicationName>.jar**

  The deployment JAR file contains the application metadata that is created through the buildDeployment command.

- **<*ApplicationName*>/appScript.txt**

  This optional file is for the application-specific script that the runScript.cmd command can run (refer to the *Velocity Studio User Guide* for details). The SUF playlist can automatically run it.

- **<*ApplicationName*>/scriptData**

  This optional folder is for application-specific files that must be placed in the node. The SUF playlist places this folder in the node.

- **<*ApplicationName*>/logback-template.xml**
  The AVM SUF package includes the default logback-template.xml file. You can use this file as a base if you want to add the file to the AVM_<*Application*>.rpm package and configure Management Server logback.

- **<*ApplicationName*>/smService.json**

  This file contains the JSON service definition and be included in the application .rpm package.

- **<*ApplicationName*>/smProfile.json**

  This file contains the JSON service profile definition and be included in the application .rpm package. A service must be in at least one service provider to allow for deployment use through a service assignment.

2.3.4.1 AVM License

The AVM installation requires licenses for both application servers and SUF server:

- The license file for the SUF server must be uploaded to the SUF server under /var/lib/sufstorage/repository/cw.

- The license file for each application server host is <*hostname*>_avm_license.lic (for example, jboss_01_avm_license.lic and jboss_02_avm_license.lic). You can specify the path to these license files during installation.

2.3.4.2 Configuration Files

Besides having the AVM SUF package, you need to have the following files, which are not a part of the delivered package:

- **Config.xml**
  This file represents the application configuration file. The file is optional, and if it is not available, the <*ApplicationName*>/config.xml file is used instead.

- **host_node_mapping.properties**
  Use this properties file when deploying the .war file to different JBoss servers. This mandatory file contains host-node pairs, such as the following:

  ```
  Jboss_01=PE
  Jboss_02=UI1
  Jboss_03=UI2
  ```

  The war_deploy playlist reads this properties file and adds a system parameter to specify the node on which this particular JBoss will be running. If this file does not exist, all JBoss servers run on the *Root* node, which is the application name.

  These two files must be copied to /var/lib/sufstorage/swtools on your SUF server.

2.3.4.3 Management Server Logback Configuration

You can add a logback-template.xml file to the AVM_<*Application*>.rpm package to configure the Management Server logback. This file is optional. If it exists, the AVM playlist changes the logback configuration accordingly.

The logging template file contains placeholders specific to the AVM logging system. These files are as follows:

- $CW_STDOUT_FILTERS$
- $CW_DB_FILTERS$
- $CW_DEFAULT_LOG_LEVELS$

If any of these placeholders are moved, changed, or removed, it may affect the AVM's ability to control log levels based on the AVM configuration. Take care when making changes to this file.

The AVM SUF package includes the default logback-template.xml file. You can use this file as a base if you want to add the file as mentioned previously.

2.4 Create an Application RPM

The AVM_<*applicationName*>.rpm is prepared by the application development team. This page provides the instructions on how to create the RPM file for different applications.

1 Create a .zip file, AVM_<ApplicationName>.zip, with the following structure:

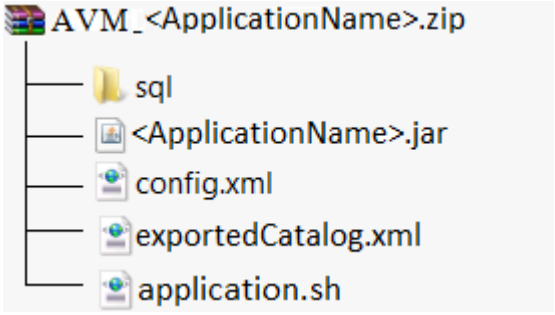

Figure 4 File structure in .zip file

2 Create a *.rpmmacros* file that defines the macros for RPM creation. Define these macros in the file:

```
%_topdir %(echo $HOME)/rpmbuild
%_tmppath %(echo $HOME)/rpmbuild/tmp
%_rpmfilename %%{NAME}.rpm
%__os_install_post \
    /usr/lib/rpm/redhat/brp-compress \
    %{!?__debug_package:/usr/lib/rpm/redhat/brp-strip %{__strip} \
    /usr/lib/rpm/redhat/brp-strip-static-archive %{__strip} \
    /usr/lib/rpm/redhat/brp-strip-comment-note %{__strip}
%{__objdump} \
%{nil}
```

3 Create a specification (spec) file named *OrderCareSUF_Application.spec* with the following content. You do not need to replace the %{ApplicationName} in this file. It is a variable and is set when creating the RPM file.

```
Name: OrderCare_%{ApplicationName}
Version: 6
Release: 0
Summary: Ordercare SUF Package for %{ApplicationName}
License: Restricted
BuildArchitectures: noarch
BuildRoot: %{_builddir}/%{name}-root
Prefix: /OrderCare_%{ApplicationName}
%description
%prep
%build
%install
pwd
rm -rf $RPM_BUILD_ROOT
mkdir -p $RPM_BUILD_ROOT/OrderCare_%{ApplicationName}
cd $RPM_BUILD_ROOT/OrderCare_%{ApplicationName}
```

```
unzip $RPM_BUILD_ROOT/../OrderCare_%{ApplicationName}.zip
%clean
%files
%defattr(644,root,root)
/*
```

4 Log in to a Linux system as root user and create the following folder structure:

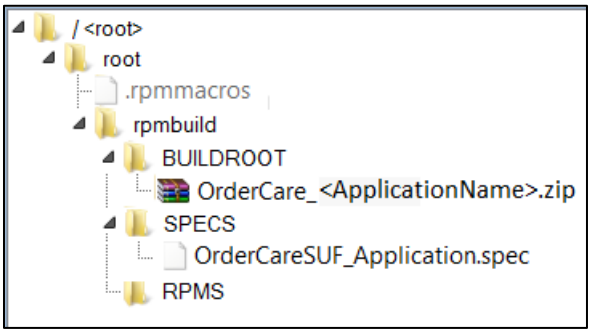

Figure 5 Create folder structure in Linux

5 Run the following command:

```
rpmbuild --bb rpm/SPECS/OrderCareSUF_Application.spec  --define
'ApplicationName'
```

6 If the command runs successfully, you can find the RPM file in */root/rpmbuild/RPMS/OrderCare_.rpm*.

2.5 Playlist Details

Playlists are the instruction for SUF server on how to install or deploy the application. There is a playlist for each step in the installation process. To install or upgrade AVM, you need to create a job and run a top-level playlist.

The four top-level playlists are as follows:

- _NewInstallation.xml
- _UpgradeInstallation.xml
- _AVMUpgrade.xml
- _MetaDataUpgrade.xml

The top-level playlists invoke other task-specific playlists. The following table provides a description of these playlists:

Top-Level Playlist	**Description**
application_deploy.xml	This playlist deploy the application metadata in the database.
application_rpm.xml	This playlist installs the AVM_<*applicationName*>.rpm package.

Top-Level Playlist	Description
application_sql.xml	This playlist runs the application specific SQL files.
avm_rpm.xml	This playlist installs the AVM_Platform.rpm package.
catalog_import.xml	This playlist imports the catalog configuration for the Catalog application.
config_init.xml	This playlist initializes the application configuration.
db_init.xml	This playlist initializes the platform database.
db_upgrade.xml	This playlist upgrades the platform database for application.
jboss_start.xml	This playlist starts or restarts the application server based on its current state. The playlist also enables the default port required by the server. If you need to enable other ports to run your application, you can change the playlist.
jdbc_install.xml	This playlist installs the JDBC driver in application servers.
jndi_setup.xml	This playlist sets up the JNDI configuration in application servers.
license_distribution.xml	This playlist distributes the license to the application servers.
ngee_conf.xml	This playlist copies the Management Server configuration on the application servers.
verify_prereq.xml	This playlist verifies the pre-requisites on application servers.
war_deploy.xml	This playlist deploys the WAR file on application servers.

When running a top-level playlist, each invoked playlist is processed as a step. If a playlist performs an application server-specific task, it iterates all application server hosts under the JBoss node. On the other hand, a database-specific task is processed only once. If a task-specific playlist fails, you can fix its reason for failing and rerun the top-level playlist from the step that it failed. You do not need not to rerun the previously successful steps.

The following sections describe details of each task-specific playlist. Refer to the failover step for troubleshooting if a step fails while it is running.

2.5.1 application_deploy.xml

This playlist deploys the <*ApplicationName*>.jar in the platform. It is processed during either a new installation or an application upgrade. The parameters for this task are as follows:

- Database Username
- Database Password
- Host
- Port
- SID
- Application Version

Failover: This step may fail because of database connection issues. You can correct the problem and can rerun this task. If the task fails due to application failure, you must contact the application team.

2.5.2 application_rpm.xml

This playlist installs the AVM application package on SUF server at the /opt/OrderCare_SUF/Applications/<*ApplicationName*>/ location. The parameter for this playlist is AVM_<*applicationName*>.rpm.

Failover: If this task fails, check the installation file, as it may be corrupted. Proceed to fix the issue and then rerun the same step.

2.5.3 application_sql.xml

This playlist runs the main.sql file available in the application package.

Failover: If this step fails for trivial reasons, such as a file being unreadable or a database connection issue, fix these issues and then rerun this task. If the task fails due to application failure, you must contact the application team.

2.5.4 avm_rpm.xml

This playlist installs the AVM AVM package on SUF server at/opt/OrderCare_SUF/AVM/ location. The parameter for this playlist is AVM_Platform.rpm.

Failover: If the task fails, check if a newer version of AVM is already installed. This step overwrites an older or identical version, but does not overwrite a newer version of AVM. You have to manually uninstall the newer version of AVM. After fixing the issue, you can rerun this step.

2.5.5 catalog_import.xml

If the catalog configuration is available, this playlist imports it to the Catalog application. The parameter for this playlist is the Catalog import file /opt/OrderCare_SUF/<ApplicationName>/exportedCatalog.xml.

Failover: If this step fails due to network access issues, you can rerun the task.

2.5.6 config_init.xml

This playlist initializes the application's configuration. This file looks for the following the three configuration files:

1 First, it looks for the customer configuration file located in /opt/OrderCare_SUF/Application/<ApplicationName>/customer/config.xml.
2 If this file is not available, the playlist looks for /opt/OrderCare_SUF/Application/customer/config.xml.
3 If both files are not available, it uses the default configuration.

Failover: If this step fails for trivial reasons, such as a file being unreadable or a database connection issue, fix these issues and then rerun this task. If the task fails due to application failure, you must contact the application team.

2.5.7 db_init.xml

This playlist creates and initializes the platform database tables. This playlist is used only during a new installation. This playlist has the following parameters:

- Database sys password
- Database new Username
- Database Password

Failover: The possible reasons for task failure includes a locked sys account, an incorrect sys password, or the specified username already exists and is currently in use. After rectifying the issue, you can rerun this task, which cleans the database before re-initializing it.

2.5.8 db_upgrade.xml

This playlist upgrades the platform database to run the application. It is the platform's internal operations. The playlist has the following parameters:

- Database Username
- Database Password
- Host
- Port
- SID
- Application Version

- Deployment JAR

Failover: If this step fails for trivial reasons, such as a file being unreadable or a database connection issue, fix these issues and then rerun this task. If the task fails due to application failure, you must contact the application team.

2.5.9 boss_start.xml

This playlist copies the JDBC drivers on application servers.

Failover: This step may fail due to issues, such as network access, port not available, and so on. After rectifying these issues, run this playlist again.

2.5.10 jdbc_install.xml

This playlist copies the JDBC drivers on application servers.

Failover: If this step fails due to network access issues, rectify the issue, and then run the task again. This action overwrites the previously copied files.

2.5.11 jndi_setup.xml

This playlist sets up the JNDI configuration on application servers.

Failover: This step may fail due to issues, such as network access, port not available, and so on. After rectifying these issues, run this playlist again.

2.5.12 license_distribution.xml

This playlist copies the license files from the specified folder to different application servers. The files must be named as <*hostname*>_avm_license.lic, so that the playlist can identify which license belongs to which machine. The parameter for this playlist is the path of the license folder on SUF server.

Failover: If this step fails (for example, a server does not have a license file), you can copy the file manually at a later time on the missing server, and then continue with the next step. It overwrites the previously copied files if you fix the issue during the failure and then rerun this step.

2.5.13 ngee_conf.xml

This playlist copies the Management Server configuration on application servers.

Failover: If this step fails due to network access issues, you can rerun the task, which overwrites the previously copied files.

2.5.14 verify_prereq.xml

This playlist is run before upgrading the installation. It checks the Java™ version and JBoss configuration on all application server hosts. All application servers must have Java version 1.6 or higher, and JBoss EAP must be installed in the default configuration. There are no other parameters for this playlist.

Failover: If the task fails, check the Java and JBoss configurations on the application servers, and then rerun this playlist.

2.5.15 war_deploy.xml

This playlist deploys the AVM WAR file to application servers. The WAR file is deployed in /var/lib/jbossas/standalone/deployments.

Failover: This step may fail due to issues, such as network access. After rectifying these issues, run this playlist again.

2.6 SUF Prerequisites

The following are some assumptions and recommendations before you start working with AVM SUF:

- The AVM SUF only works on Management Servers that are SUF server and clients.
- In SUF clients, JBoss EAP 6.1 must be installed with Management Server SUF. The following table describes important JBoss folders used in the playlists:

Folder	Description
JBoss modules	/usr/share/jbossas/modules/system/layers/base
JBoss bin	/usr/share/jbossas/bin/
JBoss deployments	/var/lib/jbossas/standalone/deployments

- Oracle must be installed only on one SUF client.
- The AVM SUF package has been developed and verified with SUF version 3.5.
- Java (JDK 1.6) is in the environment path for both the SUF server and clients.
- You must have a root account to access the SUF server and clients.
- In SUF JBoss clients, the JBoss Command Line Interface (CLI) port (9999) is accessible for the SUF server.
- A Sys account is required to access the Oracle database on one SUF client.
- AVM licenses are generated for both the SUF server and clients.

- The SUF server must have the libxslt package installed on it. This package is required for processing XML files.

2.7 SUF Server Configuration

To install AVM on a SUF server, follow these steps:

1 Log in to the SUF server as an administrator.
2 Create a **Node Type** called **CW**.
3 Create the User Class, **cw_class**, with full network, repository, and job permissions (that is, **Admin**, **Editor**, and **Viewer** network permissions) for the CW node type.
4 Create a User called **cw** with **cw_class** as its user class.
5 Import the network.xml file from the AVM package, and then proceed to edit the network connections.
6 Create a job to import the AVM_SUF.tar.gz package that contains all playlists. Ensure to use the CW node type and cw_class user class.
7 For a first-time installation, create a job to verify the prerequisites. Use the verify_prereq.xml playlist from the package.

Note: This document is a quick checklist of all installation steps of the AVM package for SUF. For first-time users, it is recommended that you read each chapter of this document in detail, to attain a full understanding of the installation process.

2.7.1 Log in to the SUF Server

The SUF Graphical User Interface (GUI) allows you to manage users and user classes, create node types, view audit logs, import network files, import repositories, create and view jobs, and perform other similar tasks.

To log in to the SUF GUI, follow these steps:

1 In your Web browser, enter the Web address without any spaces in between (for example, `https://<IP address of SUF Server>/suf/`) and port, if different from the default.
2 Enter `nmroot` in both the **User** and **Password** fields.

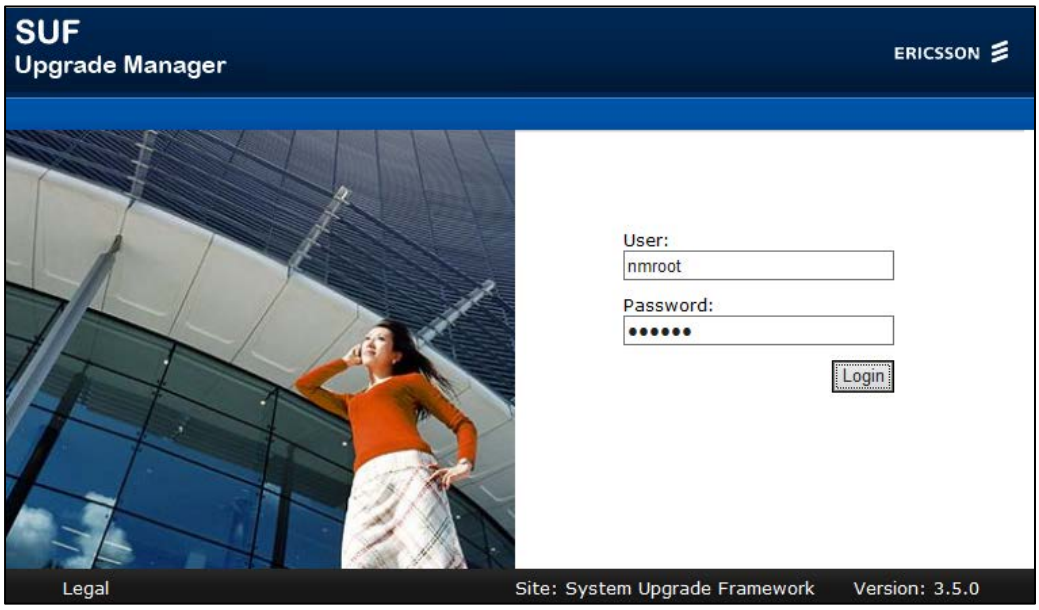

Figure 6 Log in to SUF server

3 Click the **Login** button.

Note: The default username and the password for SUF GUI is `nmtroot`. Multiple SUF GUI sessions are possible for other users at any given instance. However, only one GUI session is allowed for the nmtroot user. If any active session for the nmtroot user already exists, you can select the **Override Session** option, and then click the **Login** button. This action cancels the previous session and starts a new one.

For more information, see the *SUF User Guide* > Getting Started in the CPI Store.

2.7.2 Create a Node Type

To create a node type, follow these steps:

1 Log in to the SUF server as an administrator (that is, use nmroot account details).
2 From the main menu, click **Admimistration** > **Node Type** from the menu bar to launch the Node Type Management – Add page.

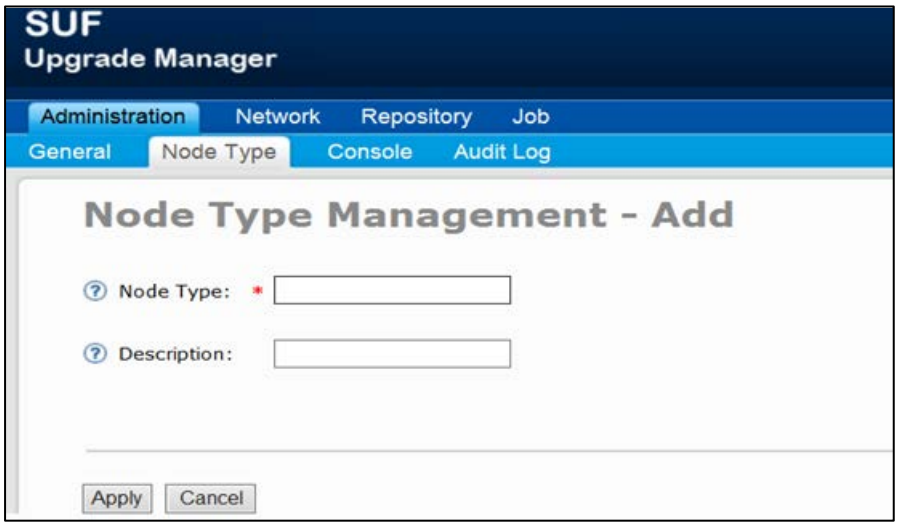

Figure 7 Add a node type

3 For the **Node Type** field, enter `CW`.

Note: This field is case-sensitive.

4 In the **Description** field, enter a brief description for the node type.

5 Click the **Apply** button to save your changes.

2.7.3 Create a User Class

To create a user class, do the following:

1 From the main menu, click the **Administration** > **General** > **User Classes**.

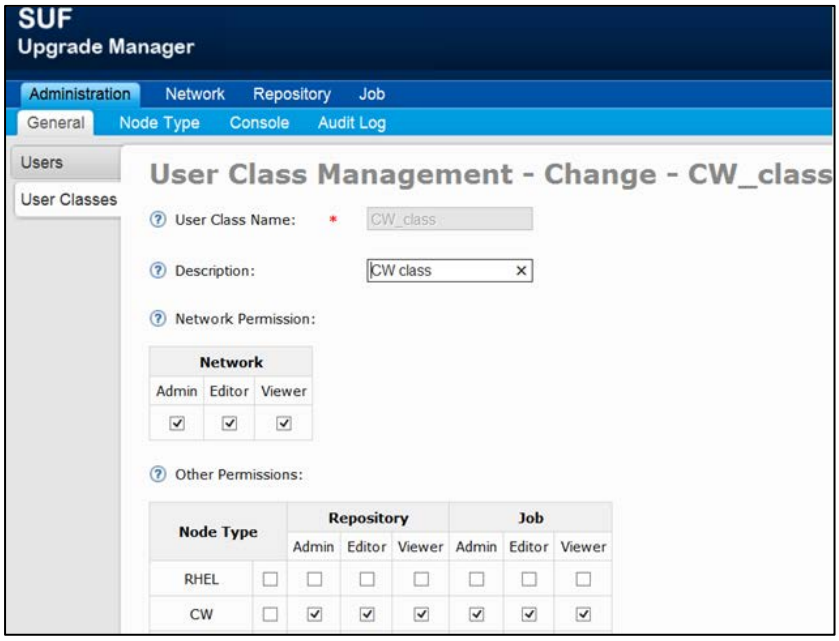

Figure 8 Create a user class

2 For the **User Class Name** field, enter `cw_class`.
3 In the Description field, enter a brief description for the user class.
4 Select the **Network**, **Repository**, and **Job** permissions by selecting the following:

- Admin
- Editor
- Viewer

Note: Step 2 must be done to be able to select these permissions.

5 Click the **Apply** button.

For more information, see the *SUF User Guide* in the CPI Store.

2.7.4 Create a User

To create a user for SUF, follow these steps:

1 From the menu bar, click **Administration** > **General** > **Users** menu, and create a new user, `cw`, with `cw_class` as its user class.
2 Enter `cw` for the following fields:

- User Name
- User ID
- Password

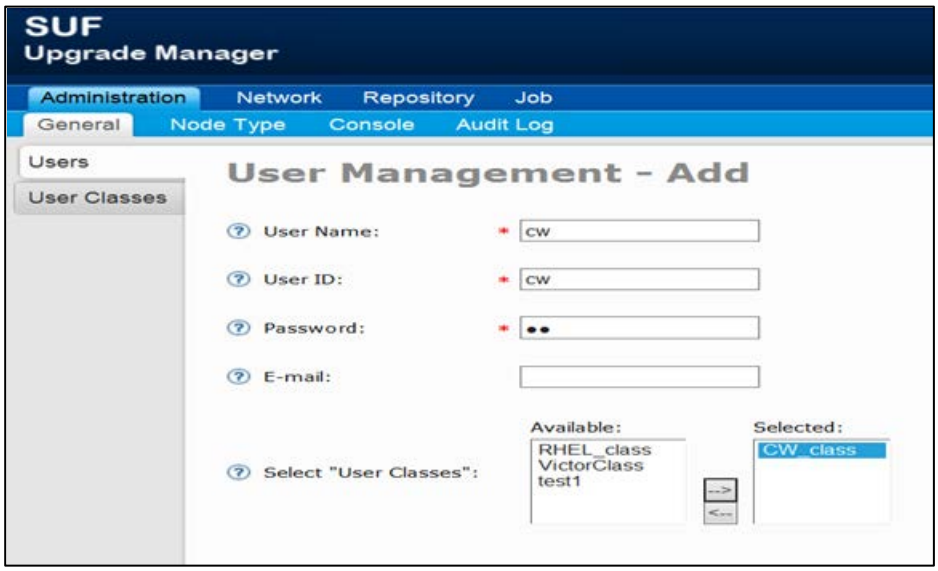

Figure 9 User Management - Add a user

3 Enter the user's **E-mail** address in the provided field.
4 For the **Select User Classes** field, select **cw_class** from the **Available** list, and then click the **Add** (right arrow) button to add the user class to the **Selected** list.
5 Click the **Apply** button to save your changes.

6 Log out from the SUF server, and then log in as the **cw** user.

For more information, see the *SUF User Guide* in the CPI Store.

2.7.5 Import Network File

To configure the SUF server, you must import the network.xml file by following these steps:

1 Log in to the SUF server as a **cw** user.
2 From the menu bar, click **Network**, and then click the **Import** button.
3 From the Network – Import page, click the **Browse** button, and select network.xml file. This file is located in the AVM SUF package.

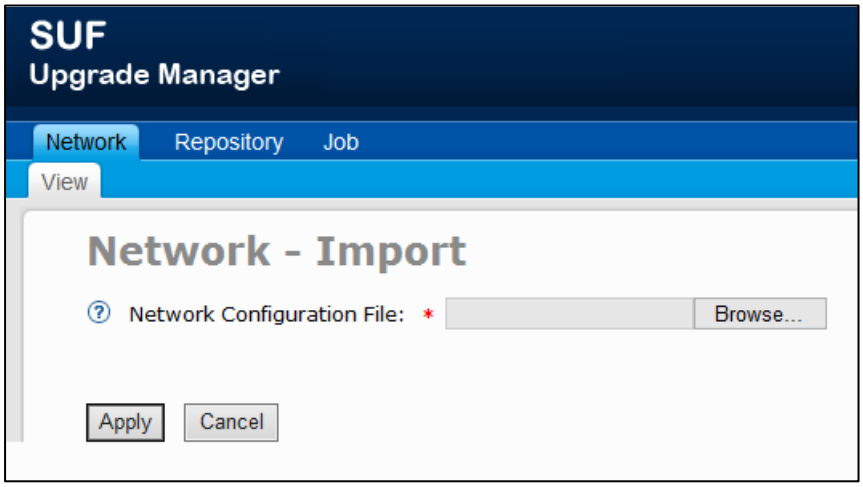

Figure 10 Import a network configuration file

Notes:

- This file creates a network named *CW_NETWORK*.
- The network has **JBoss** and **Oracle** as its two nodes.
- The **JBoss** node lists all JBoss servers in different SUF clients.
- The JBoss node can have multiple hosts. Each host has two accesses named **jboss_ssh** and **jboss_sftp**.
- The **Oracle** node can have only one Oracle instance in a SUF client, with two accesses named **oracle_ssh** and **oracle_sftp**.

4 Click the **Apply** button.
5 After importing the network file, you can edit and verify all network nodes.

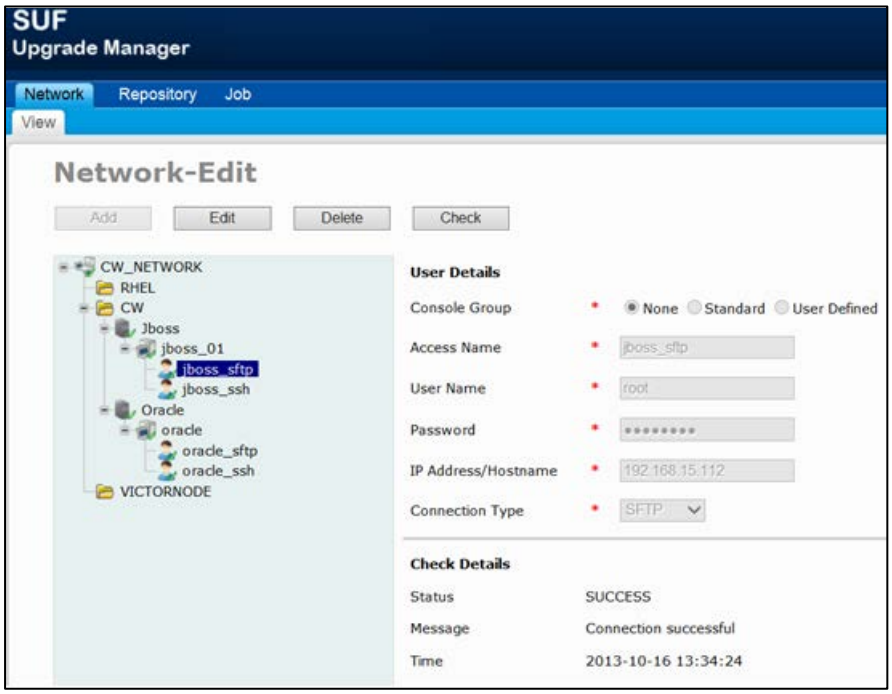

Figure 11 Edit network configuration

Note: You can either change or add new JBoss hosts. However, you can only change the existing Oracle host.

For more information, see the *SUF User Guide* in the CPI Store.

2.7.6 Import AVM_SUF.tar.gz

To import the .tar.gz file from the package to the SUF server, follow these steps:

1 Log in to the SUF server as a **cw** user.
2 From the menu bar, click **Repository** > **Node Package**, and then click the **Import** button.
3 From the Node Package - Import page, do the following:
   a In the **Package Name** field, enter `OrderCare_SUF`.
   b In the **Node Type** field, enter `cw`.
   c Click the **Browse** button and locate the AVM_SUF.tar.gz package file.
   d Click the **Apply** button.

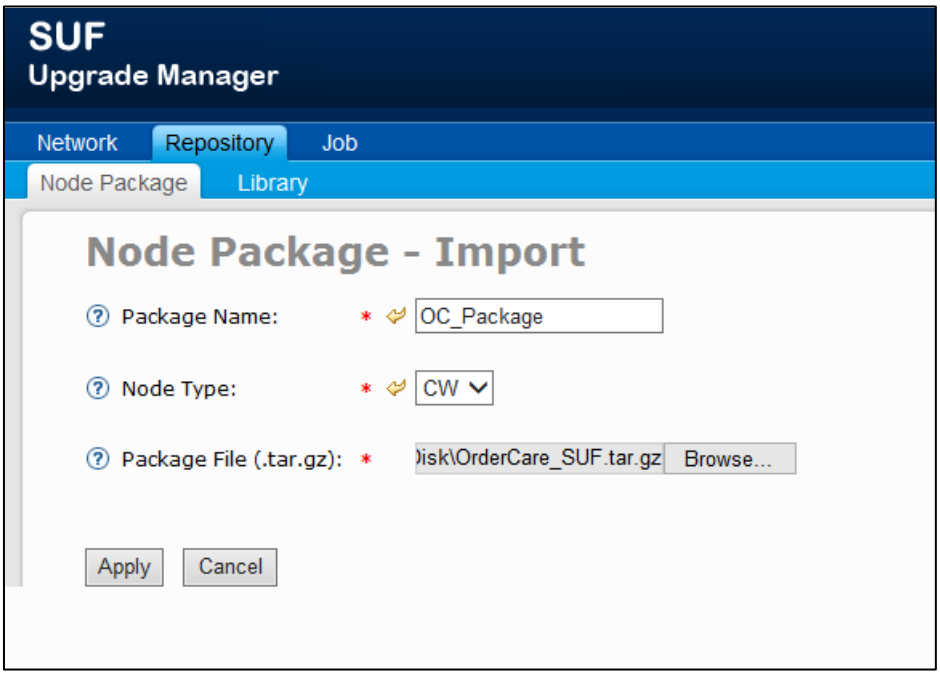

Figure 12 Import node package

For more information, see the *SUF User Guide* in the CPI Store.

2.7.7 Verify Prerequisites

For first-time deployment, you can create a job to verify the prerequisites to run any other playlists. This job checks the Java Development Kit (JDK) version on both SUF server and clients, and the JBoss installations on SUF clients.

To verify the prerequisites, do the following:

1 From the main menu, click **Job** > **Current**.
2 Click the **Create New** button to launch the Job - Add - Step 1 / 8 page.

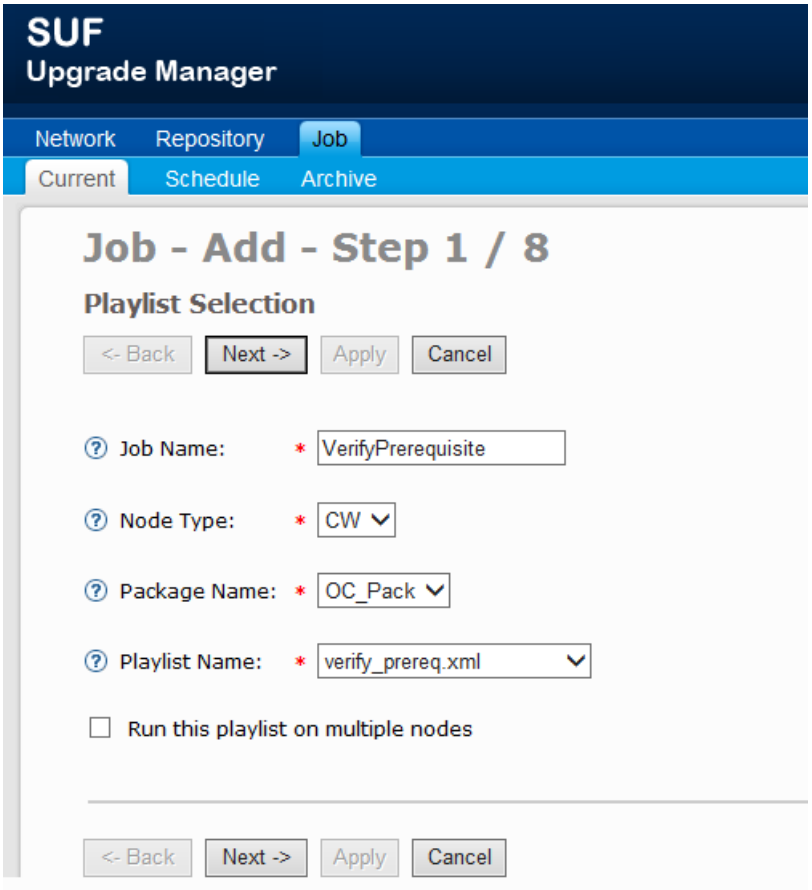

Figure 13 Select a playlist

3 To select a playlist, do the following:
   a Enter the **Job Name** in the field that appears.
   b For the **Node Type** field, click the dropdown menu and select **cw** from the list.
   c For the **Package Name** field, click the dropdown menu and select **OC_Pack** from the list.
   d For the **Playlist Name** field, select the **verify_prereq.xml** file, which contains a list of playlist for the Order Care package.

   Note: If you want to run the playlist on multiple nodes, select the **Run this playlist on multiple nodes** checkbox. Selecting this option enables an option to add multiple node sets in this step.

4 Click the **Next** button for the following steps:

   a File Selection
   b Node Selection
   c Variable Selection
   d Configuration Setup
   e Step and Task Selection
   f Checkpoint Definition

   Note: You can add a checkpoint in the Step and Task Selection step.

5 The Creation Summary step displays the job's summary details.

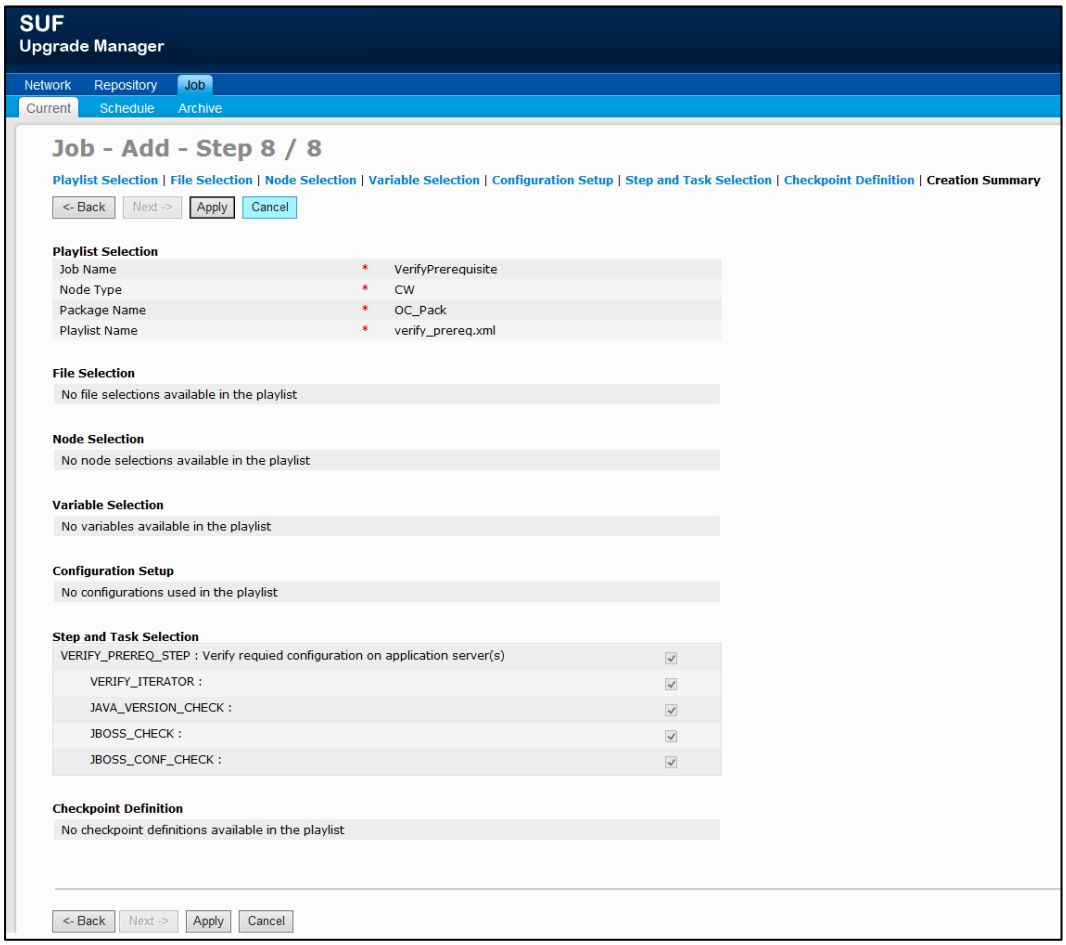

Figure 14 Creation Summary step

6 Click the **Apply** button to create the new job, and then click the **Done** button to return to the Job list page.

For more information, see the *SUF User Guide* > *Creating a New Job* in the CPI Store.

Note: If the job does not run successfully, check if there are any fundamental environment problems, such as an incorrect JDK version, and so on.

2.8 Installation and Upgrade Types

With the AVM SUF package, you can perform the following installations and upgrades:

- Perform a new installation
- Upgrade a full installation
- Upgrade an AVM installation
- Upgrade an application installation

2.8.1 Perform a New Installation

For a new installation, you need to perform the following:

1 Install the AVM_platform.rpm and application.rpm files
2 Distribute order care licenses to JBoss servers
3 Create a database user
4 Initialize the database
5 Deploy the metadata
6 Initialize the configuration
7 Upgrade database
8 Configure JBoss (JDBC JNDI)
9 Start JBoss
10 Deploy the WAR file

Through FTP, you can upload OrderCare_AVM_<version>.rpm and AVM_<applicationName>.rpm to SUF's /var/lib/sufstorage/swtools location.

For a new installation of Order Care, do the following:

1 From the main menu, click **Job** > **Current**.
2 Click the **Create New** button. The Job - Add - Step 1 / 8 page appears.

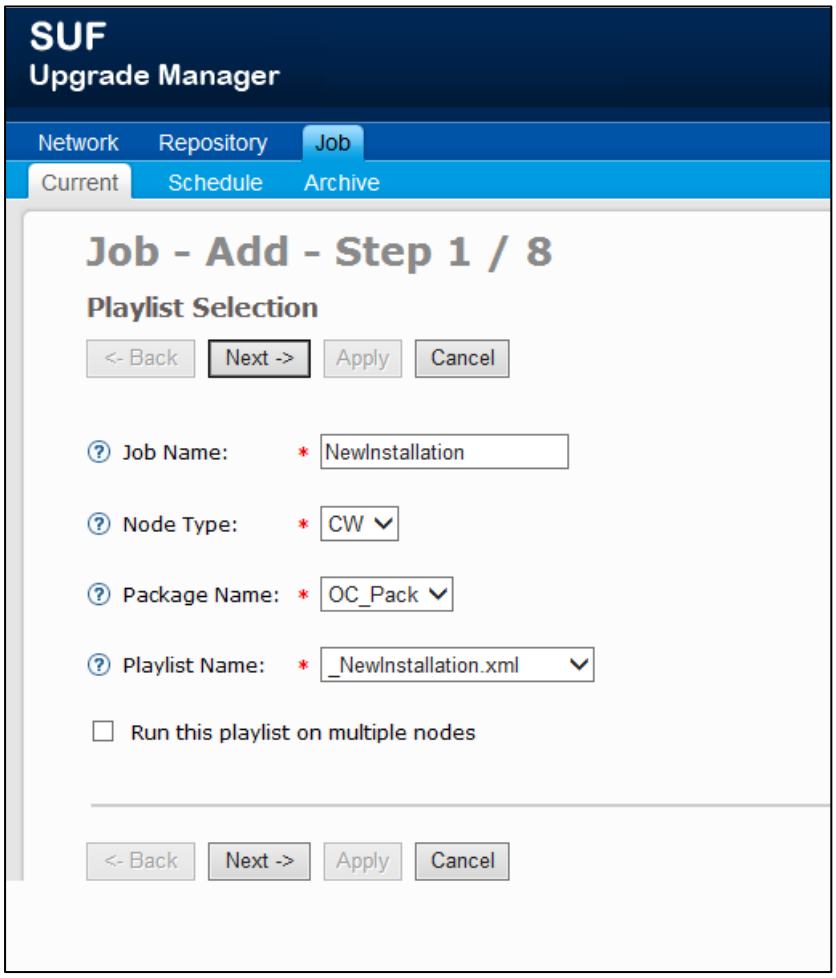

Figure 15 Step 1 – Playlist selection

a Enter the name of the job (for example, `NewInstallation`) in the **Job Name** field.
b Click the drop-down menu and select the node type `CW` for the **Node Type** field.
c Click the drop-down menu and select the package **OC-Package** for the **Package Name** field.
d Click the drop-down menu and select **_NewInstallation.xml** for the **Playlist Name** field.

Note: Based on the selected playlist, SUF provides the options to select an AVM package, Application package, or both.

3 Click the **Next** button to launch the Job - Add - Step 2 / 8 (File Selection) page.

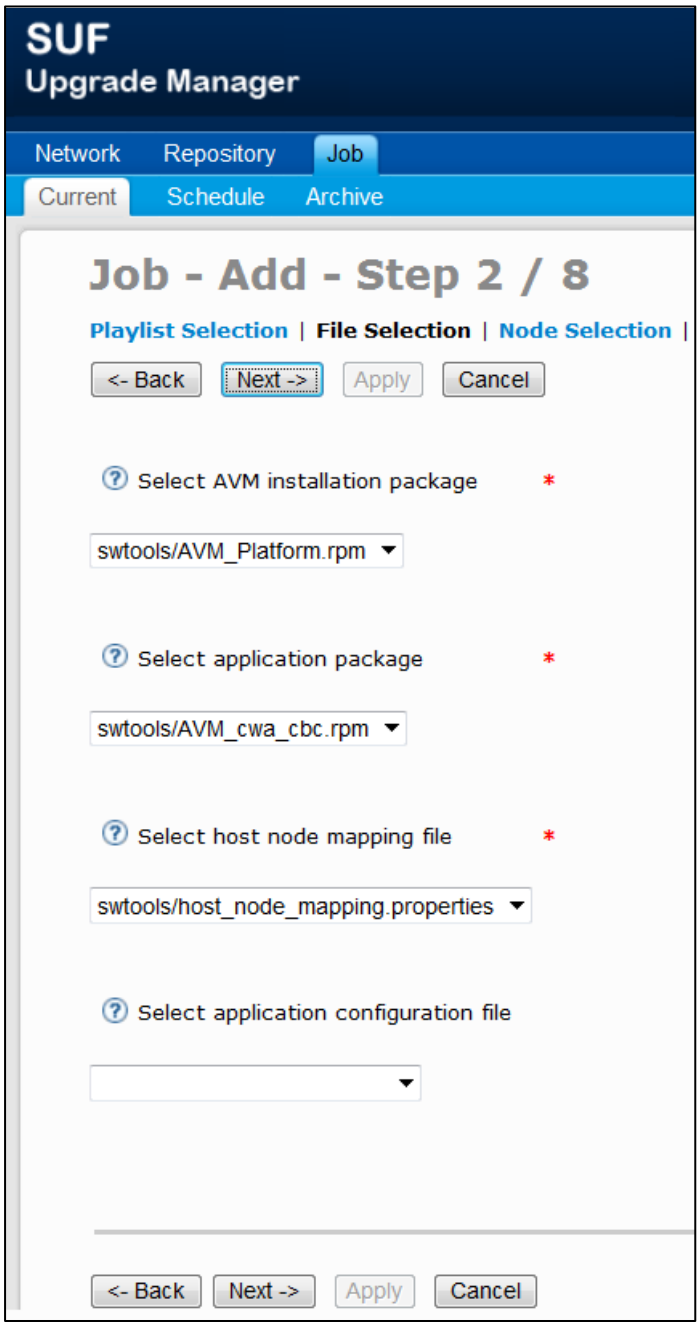

Figure 16 Step 2 – File selection

a Click the **Select AVM installation package** field's drop-down menu and select the **AVM RPM** package. This RPM is uploaded to the SUF server under /var/lib/sufstorage/swtools.

b Click the **Select application package** field's drop-down menu and select an application RPM package. This RPM is also uploaded to the SUF server under /var/lib/sufstorage/swtools.

4 Click the **Next** button to launch the Job - Add - Step 3 / 8 (Node Selection) page. There are no node selections available for this step.

5 Click the **Next** button to launch the Job - Add - Step 4 / 8 (Variable Selection) page.

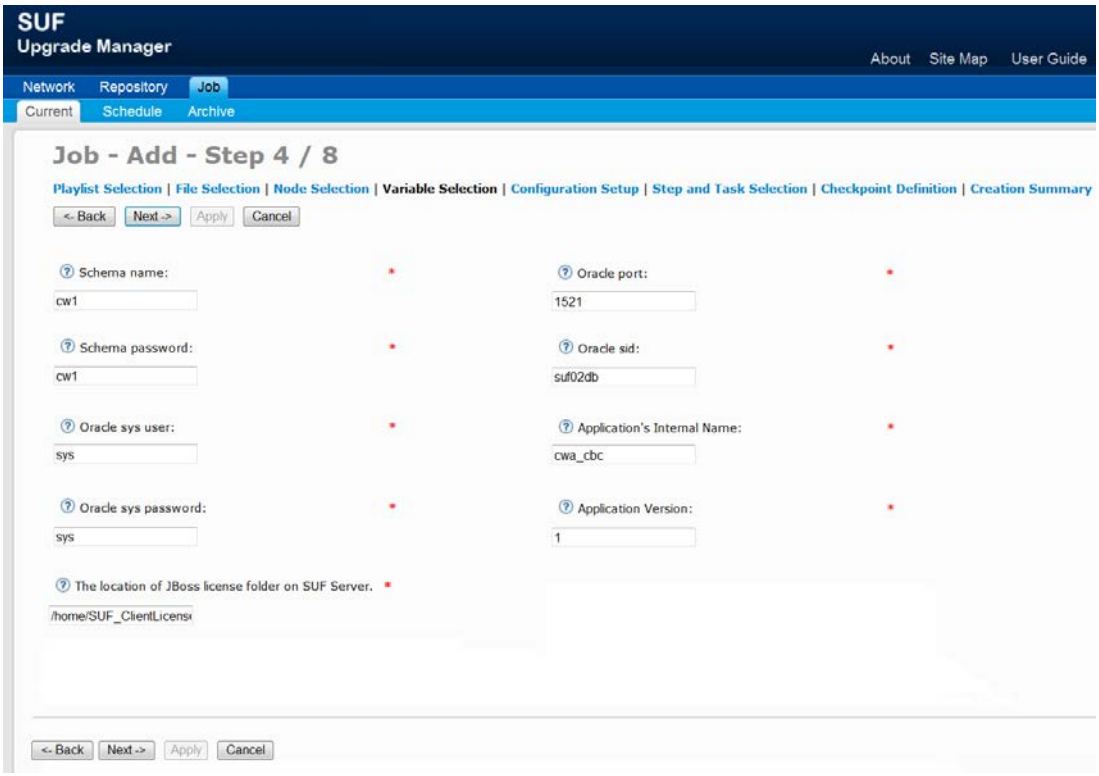

Figure 17 Step 4 - Variable selection

a Enter information for the **Database username** and **Database password** fields. This information is used to create a new database user. A user with the same name is dropped. The same credendials are used to upgrade the system.

b Enter the information for the **Oracle sys user** and **Oracle sys password** fields. The default value for these fields is sys. The Oracle sys user is used to create the new database user.

c Enter the value for the **Oracle Port** field. The default value is 1521.

d Enter the service ID for Oracle instance in the **Oracle sid** field.

e Enter the information for the **Application Version** field.

f Specify the location of JBoss license folder in **The location of JBoss license folder on SUF Server** field. This folder contains all licenses files to be distributed to the JBoss servers. The naming convention is important.

6 Click the **Next** button to launch the Job - Add - Step 5 / 8 (Configuration Selection) page. There are no node selections available for this step.

7 Click the **Next** button to launch the Job - Add - Step 6 / 8 (Step and Task Selection) page.

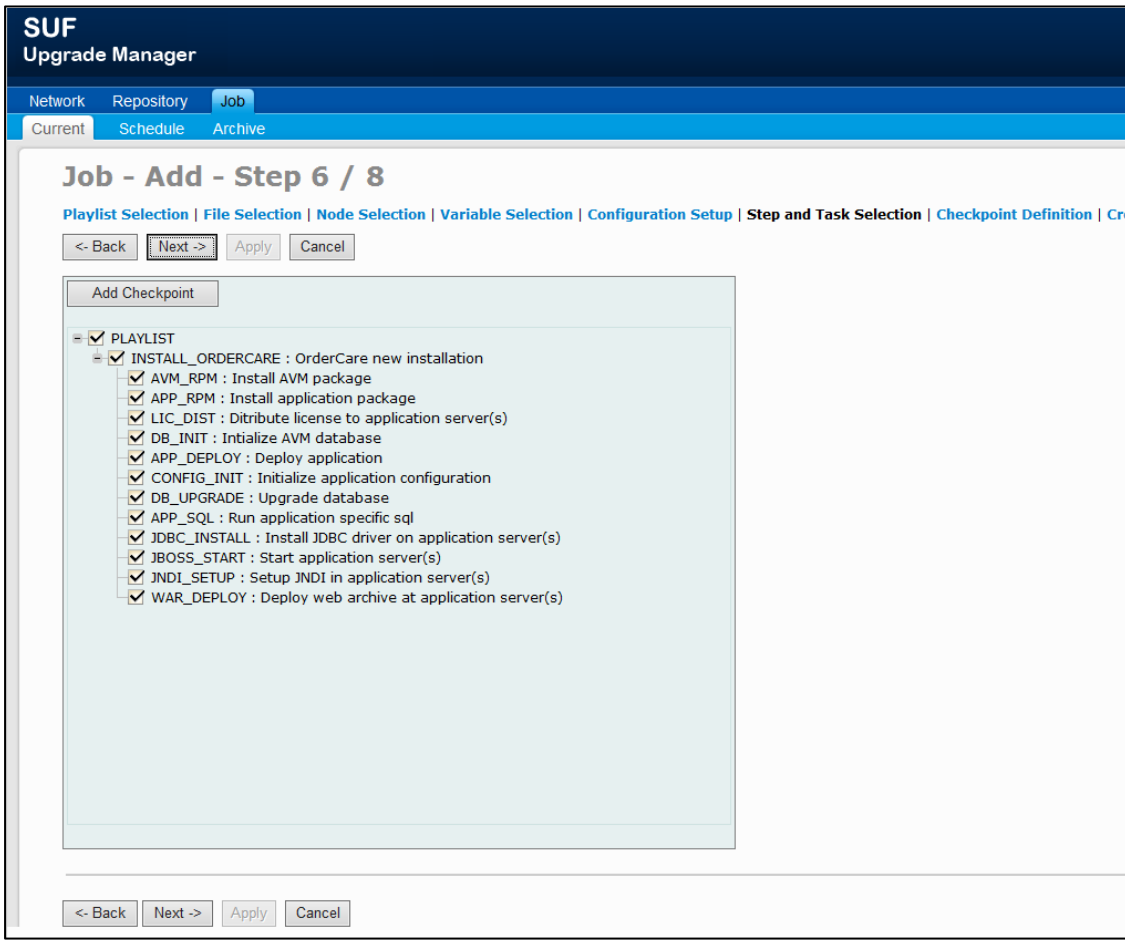

Figure 18 Step 6 - Step and task selection

Each top-level playlist invoke some task-specific playlists. This page displays the list of all the invoked playlist for the selected top-level playlist. By default, all steps are selected. It is recommended to run all steps. However, if you have run any step previously, you may skip those steps. For example, uncheck the **Skip LIC_DIST** step if the license files are already transferred to JBoss servers. You can uncheck the **Skip JDBC_INSTALL** step that copies the Oracle JDBC driver to JBoss server, if it is already done by a previous installation.

You can add a checkpoint on any option displayed on this page. To add the checkpoints, do the following:

a Select one playlist (for example, DB_Upgrade: Upgrade database), and then click the **Add Checkpoint** button.
b The User Defined Checkpoint popup appears. Enter the check point ID and a description for the check point in the **Checkpoint ID** and **Checkpoint Description** fields, respectively.
c In the popup window, click the **Save** button.

8 Click the **Next** button from the Step and Task Selection page to launch the Job - Add - Step 7 / 8 (Checkpoint Definition) page.

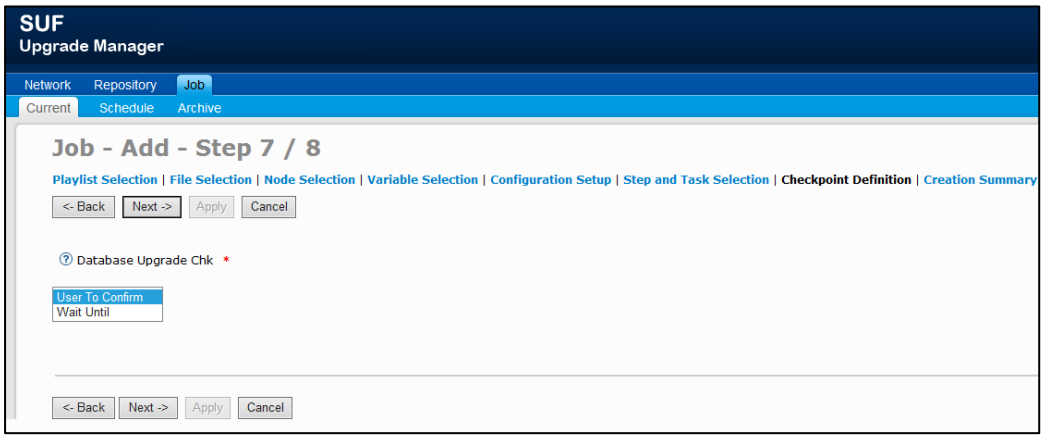

Figure 19 Step 7 - Checkpoint definition

This page specifies the actions to be performed for each checkpoint defined in the Job - Add - Step 6 / 8 (Step and Task Selection). This page contains **User To Confirm** and **Wait Until** options.

For more information, see the *SUF User Guide > Creating a New Job* in the CPI Store.

9 Click the **Next** button to launch the Job - Add - Step 8 / 8 (Creation Summary) page, which displays the summary of all values selected from steps 1 through 7.

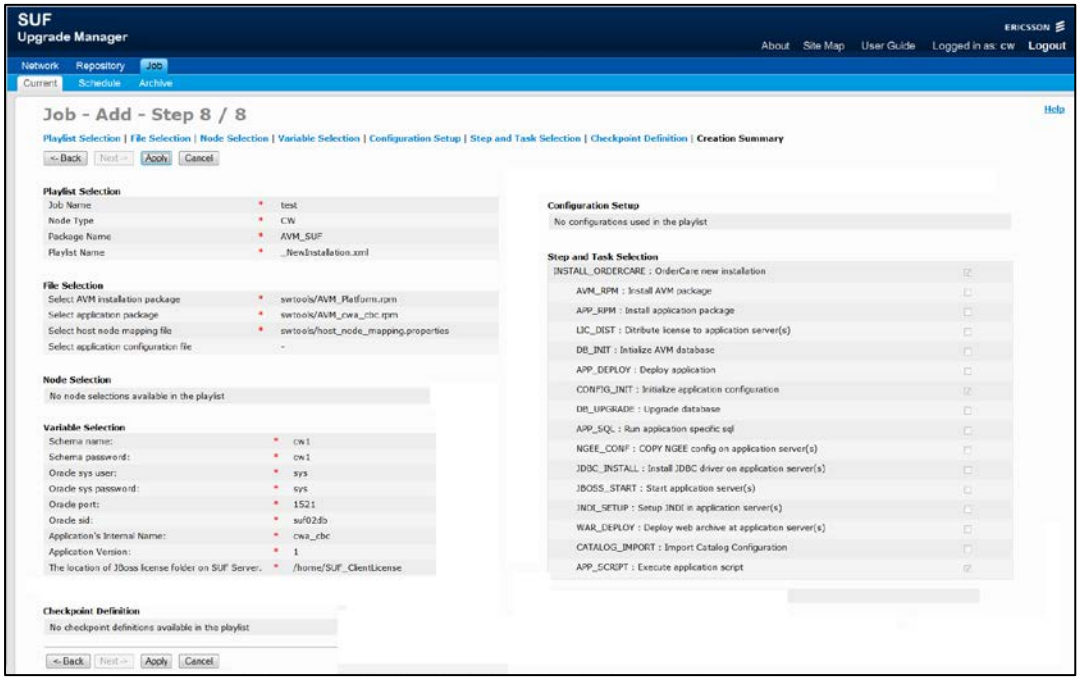

Figure 20 Step 8 - Creation Summary

You can go back to the previous steps to make corrections. Click the **Apply** to create the new job. Otherwise, click the **Done** button to return to Job list page.

For more information, see the *SUF User Guide* in the CPI Store.

2.8.2 Upgrade a Full Installation

The _UpgradeInstallation.xml top playlist is used for a system upgrade, including AVM and application upgrades. This job includes the following:

- Install AVM_platform.rpm and application.rpm
- Deploy the application
- Upgrade database
- Run the application SQL
- Stop or start JBoss
- Deploy the WAR file

This playlist is required for the deployment upgrade when there is a new release of the AVM and application.

To upgrade an installation of Order Care, do the following:

1 From the main menu, click **Job** > **Current**.
2 Click the **Create New** button. The Job - Add - Step 1 / 8 page appears.
   a Enter the name of the job (for example, `UpgradeInstallation`) in the **Job Name** field.
   b Click the drop-down menu and select the node type **CW** for the **Node Type** field.
   c Click the drop-down menu and select the package **OC-Package** for the **Package Name** field.
   d Click the drop-down menu and select _**UpgradeInstallation.xml** for **Playlist Name** field.

   Note: Based on the selected playlist, SUF provides the options to select an AVM package, Application package, or both. You must copy the RPM files in the /var/lib/sufstorage/swtools directory of SUF.
3 Click the **Next** button to launch the Job - Add - Step 2 / 8 (File Selection) page.

   a Click the **Select AVM installation package** field's drop-down menu and select the **AVM RPM** package. This RPM is uploaded to the SUF server under /var/lib/sufstorage/swtools.
   b Click the **Select application package** field's drop-down menu and select the **Application RPM** package. This RPM is also uploaded to the SUF server under /var/lib/sufstorage/swtools.
4 Click the **Next** button to launch the Job - Add - Step 3 / 8 (Node Selection) page. There are no node selections available for this step.
5 Click the **Next** button to launch the Job - Add - Step 4 / 8 (Variable Selection) page.
   a Enter information for the **Database username** and **Database password** fields. The credentials must be the same that you use to create the database user.
   b Enter the value for the **Oracle Port** field. The default value is 1521.
   c Enter the service ID for Oracle instance in the **Oracle sid** field.

   d Enter the information for the **Application Version** field.
6 Click the **Next** button to launch the Job - Add - Step 5 / 8 (Configuration Selection) page. There are no node selections available for this step.
7 Click the **Next** button to launch the Job - Add - Step 6 / 8 (Step and Task Selection) page. This page displays the task-specific playlists invoked by Upgrade Installation playlist.

   You can add a checkpoint on any option displayed on this page. To add the checkpoints, do the following:
   a Select one playlist (for example, **DB_Upgrade: Upgrade database**) and then click the **Add Checkpoint** button.
   b The User Defined Checkpoint popup appears. Enter the checkpoint ID and a description for the checkpoint in the **Checkpoint ID** and **Checkpoint Description** fields, respectively.
   c In the popup window, click the **Save** button.
8 Click the **Next** button from the Step and Task Selection page to launch the Job - Add - Step 7 / 8 (Checkpoint Definition) page. There are no node selections available for this step if you have not added the checkpoint in the Step 6 / 8 (Step and Task Selection).
9 Click the **Next** button to launch the Job - Add - Step 8 / 8 (Creation Summary) page, which displays the summary of all values selected from step 1 through 7. You can go back to the previous steps to make corrections. Click the **Apply** button to create the new job. Otherwise, click the **Done** button to return to Job list page.

For more information, see the *SUF User Guide* in the CPI Store.

2.8.3 Upgrade an AVM Installation

The _AVMUpgrade.xml playlist is used to do an AVM upgrade. This job includes the following:

- Install AVM_platform.rpm
- Upgrade database
- Stop or start JBoss
- Deploy the War file

This playlist is required for the deployment upgrade when there is a new release of AVM, but the application remains the same.

To do the AVM upgrade, follow these steps:

1 From the main menu, click **Job** > **Current**.
2 Click the **Create New** button. The Job - Add - Step 1 / 8 page appears.
   a Enter the name of the job (for example, `UpgradeInstallation`) in the **Job Name** field.
   b Click the drop-down menu and select **CW** for **Node Type** field.
   c Click the drop-down menu and select **OC-Package** for the **Package Name** field.
   d Click the drop-down menu and select **_AVMUpgrade.xml** for the **Playlist Name** field.

Note: Based on the selected playlist, SUF provides the options to select an AVM package or application package, or both. You must copy the RPM files in the /var/lib/sufstorage/swtools directory of SUF.

3 Click the **Next** button to launch the Job - Add - Step 2 / 8 (File Selection) page. Proceed to click the **Select AVM installation package** field's drop-down menu and select the **AVM RPM** package. This RPM is uploaded to the SUF server under /var/lib/sufstorage/swtools.
4 Click the **Next** button to launch the Job - Add - Step 3 / 8 (Node Selection) page. There are no node selections available for this step.
5 Click the **Next** button to launch the Job - Add - Step 4 / 8 (Variable Selection) page.
   a Enter information for the **Database username** and **Database password** fields. The credentials must be the same that you use to create the database user.
   b Enter the value for the **Oracle Port** field. The default value is 1521.
   c Enter the service ID for the Oracle instance in the **Oracle sid** field.
   d Enter the information for the **Application Name** field.
   e Enter the information for the **Application Version** field.
6 Click the **Next** button to launch the Job - Add - Step 5 / 8 (Configuration Selection) page. There are no node selections available for this step.
7 Click the **Next** button to launch the Job - Add - Step 6 / 8 (Step and Task Selection) page. This page displays the task-specific playlists invoked by top level playlist Upgrade Installation.

   You can add a checkpoint on any option displayed on this page. To add the checkpoints, do the following:

   a Select one playlist (for example, **DB_Upgrade: Upgrade database**) and then click the **Add Checkpoint** button.
   b The User Defined Checkpoint popup appear. Enter the checkpoint ID and a description for the checkpoint in the **Checkpoint ID** and **Checkpoint Description** fields, respectively.
   c In the popup window, click the **Save** button.
8 Click the **Next** button from the Step and Task Selection page to launch the Job - Add - Step 7 / 8 (Checkpoint Definition) page. There are no node selections available for this step if you have not added the checkpoint in the Step 6 / 8 (Step and Task Selection).
9 Click the **Next** button to launch the Job - Add - Step 8 / 8 (Creation Summary) page, which displays the summary of all values selected from steps 1 through 7. You can go back to the previous steps to make corrections. Click the **Apply** button to create the new job. Otherwise, click the **Done** button to return to Job list page.

For more information, see the *SUF User Guide* in the CPI Store.

2.8.4 Upgrade an Application Installation

The _ApplicationUpgrade.xml file is used for the application upgrade. This job includes the following:

- Installation application.rpm

- Deploy application
- Upgrade database
- Run the application SQL
- Stop or start JBoss. It is required for the deployment upgrade when there is a new release of application but the AVM remains same.

To upgrade application installation, do the following:

1 From the main menu, click **Job** > **Current**.
2 Click the **Create New** button. The Job - Add - Step 1 / 8 page appears.
   a Enter the name of the job (for example, UpgradeInstallation) in the **Job Name** field.
   b Click the drop-down menu and select **CW** for the **Node Type** field.
   c Click the drop-down menu and select **OC-Package** for the **Package Name** field.
   d Click the drop-down menu and select _**ApplicationUpgrade.xml** for the **Playlist Name** field.

   Note: Based on the selected playlist, SUF provides the options to select an AVM package, application package, or both. You must copy the RPM files in the /var/lib/sufstorage/swtools directory of SUF.
3 Click the **Next** button to launch the Job - Add - Step 2 / 8 (File Selection) page. Proceed to click the **Select application package** field's drop-down menu and select an application RPM package to upgrade. This RPM is also uploaded to the SUF server under /var/lib/sufstorage/swtools.
4 Click the **Next** button to launch the Job - Add - Step 3 / 8 (Node Selection) page. There are no node selections available for this step.
5 Click the **Next** button to launch the Job - Add - Step 4 / 8 (Variable Selection) page.
   a Enter information for the **Database username** and **Database password** fields. The credentials must be the same that you use to create the database user.
   b Enter the value for the **Oracle Port** field. The default value is 1521.
   c Enter the service ID for the Oracle instance in the **Oracle sid** field.
   d Enter the information for the **Application Version** field.
6 Click the **Next** button to launch the Job - Add - Step 5 / 8 (Configuration Selection) page. There are no node selections available for this step.
7 Click the **Next** button to launch the Job - Add - Step 6 / 8 (Step and Task Selection) page. This page displays the task-specific playlists invoked by the Application Upgrade playlist.

   You can add a checkpoint on any option displayed on this page. To add the checkpoints, do the following:
   a Select one playlist (for example, **DB_Upgrade: Upgrade database**) and then click the **Add Checkpoint** button.
   b The User Defined Checkpoint popup appears. Enter the checkpoint ID and a description for the checkpoint in the **Checkpoint ID** and **Checkpoint Description** fields, respectively.
   c In the popup window, click the Save button.
8 Click the **Next** button from the Step and Task Selection page to launch the Job - Add - Step 7 / 8 (Checkpoint Definition) page. There are no node selections available for this step if you have not added the checkpoint in the Step 6 / 8 (Step and Task Selection).

9 Click the **Next** button to launch the Job - Add - Step 8 / 8 (Creation Summary) page, which displays the summary of all values selected from steps 1 through 7. You can go back to the previous steps to make corrections. Click the **Apply** button to create the new job. Otherwise, click the **Done** button to return to Job list page.

For more information, see the *SUF User Guide* in the CPI Store.

3 Acronyms

SUF – System Upgrade Framework

4 Reference List

The following is a list of documentation for reference:

- *Velocity Studio User Guide*
- *Diagnostic and Support Guide*

5 Trademarks

Ericsson, the Ericsson logo and the Globemark are trademarks of Ericsson.

Ericsson is a recognized leader in delivering communications capabilities that enhance the human experience, ignite and power global commerce, and secure and protect the world's most critical information. Serving both service provider and enterprise Customers, Ericsson delivers innovative technology solutions encompassing end-to-end broadband, Voice over IP, multimedia services and applications, and wireless broadband designed to help people solve the world's greatest challenges. Ericsson does business in more than 150 countries. For more information, visit Ericsson on the Web at www.Ericsson.com.

6 Disclaimer

This document may contain statements about a number of possible benefits that Ericsson believes may be achieved by working with Ericsson. These might include such things as improved productivity, benefits to end users or cost savings. Obviously, these can only be estimates. Gains might be qualitative and hard to assess or dependent on factors beyond Ericsson's control. Any proposed savings are speculative and may not reflect actual value saved. Statements about future market or industry developments are also speculative.

Statements regarding performance, functionality, or capacity are based on standard operating assumptions, do not constitute warranties as to fitness for a particular purpose, and are subject to change without notice.

This document contains Ericsson's proprietary, confidential information and may not be transmitted, reproduced, disclosed, or used otherwise in whole or in part without the express written authorization of Ericsson.