Oracle® Endeca Information Discovery Studio

Provisioning Service Administration Guide

Version 3.0.0 • March 2013

ORACLE

Copyright and disclaimer

Table of Contents

Preface

Oracle® Endeca Information Discovery Studio is an enterprise data discovery platform for advanced, yet intuitive, exploration and analysis of complex and varied data.

Information is loaded from disparate source systems and stored in a faceted data model that dynamically supports changing data. This integrated and enriched data is made available for search, discovery, and analysis via interactive and configurable applications. Oracle Information Discovery Studio includes a Provisioning Service that allows you to upload data directly from spreadsheet files.

Oracle Endeca Information Discovery Studio enables an iterative “model-as-you-go” approach that simultaneously frees IT from the burdens of traditional data modeling and supports the broad exploration and analysis needs of business users.

About this guide

This guide describes how to configure and manage the Provisioning Service.

The Provisioning Service processes Excel files uploaded by Studio users. The Provisioning Service converts the data in the file, and then loads the converted data into an Endeca Server data domain.

Who should use this guide

This guide is intended for system administrators managing the Provisioning Service.

Conventions used in this document

The following conventions are used in this document.

Typographic conventions

The following table describes the typographic conventions used in this document.

Typeface	Meaning
User Interface Elements	This formatting is used for graphical user interface elements such as pages, dialog boxes, buttons, and fields.
`Code Sample`	This formatting is used for sample code phrases within a paragraph.
Variable	This formatting is used for variable values. For variables within a code sample, the formatting is *`Variable`*.
`File Path`	This formatting is used for file names and paths.

Symbol conventions

The following table describes symbol conventions used in this document.

Symbol	Description	Example	Meaning
>	The right angle bracket, or greater-than sign, indicates menu item selections in a graphic user interface.	File > New > Project	From the File menu, choose New, then from the New submenu, choose Project.

Contacting Oracle Customer Support

Oracle Customer Support provides registered users with important information regarding Oracle software, implementation questions, product and solution help, as well as overall news and updates from Oracle.

You can contact Oracle Customer Support through Oracle's Support portal, My Oracle Support at *https://support.oracle.com*.

Chapter 1

Configuring the Provisioning Service

The Provisioning Service runs as a web application in a WebLogic Server container. A standard WebLogic Server configuration file (`eidProvisioningConfig/plan.xml`) is provided to configure the Provisioning Service.

About the Provisioning Service

Oracle Endeca Information Discovery Studio includes the Provisioning Service, which enables dynamic application creation from data uploaded from the desktop.

The Provisioning Service profiles and creates data before sending it to Endeca Server for ingest.

Note: In Version 3.0, the Provisioning Service only supports upload of Excel files.

The Provisioning Service runs as a web application in a WebLogic Server container.

Modifying the Provisioning Service configuration

This topic describes the overall procedure for modifying the Provisioning Service configuration.

To modify the Provisioning Service configuration:

1. Modify the configuration in the deployment plan (`eidProvisioningConfig/plan.xml`), which is located in the Provisioning Service domain home directory in the WebLogic Server installation.
2. Use the weblogic.deployer tool to deploy the modified deployment plan to the Provisioning Service.

   For details about this tool, see *http://docs.oracle.com/cd/E23943_01/web.1111/e13702/wldeployer.htm*.

Provisioning Service configuration properties

This topic describes the available configuration parameters for the Provisioning Service.

The following example illustrates the available Provisioning Service configuration properties:

```
<variable-definition>
    <variable>
      <name>endeca-server-host-name</name>
      <value>${endecaServerHost}</value>
    </variable>
    <variable>
      <name>endeca-server-ws-port</name>
      <value>${endecaServerPort}</value>
    </variable>
    <variable>
      <name>endeca-server-context-root</name>
      <value>${endecaServerContextRoot}</value>
    </variable>
    <variable>
      <name>endeca-server-data-domain-profile</name>
      <value>${dataDomainProfile}</value>
    </variable>
    <variable>
      <name>endeca-server-security-enabled</name>
      <value>${endecaServerSSL}</value>
    </variable>
    <variable>
      <name>upload-file-directory</name>
      <value/>
    </variable>
    <variable>
      <name>message-driven-bean-limit</name>
      <value>2</value>
    </variable>
 </variable-definition>
```

The following table describes the Provisioning Service configuration properties.

Table 1.1: Provisioning Service configuration properties

Property name	Description	Default value	Valid values
`endeca-server-host-name`	Specifies the name of the host for the Endeca Server where you want to provision data. The default value assumes that the Endeca Server is installed on the same machine as the Provisioning Service.	localhost	Valid host names or IP addresses

Property name	Description	Default value	Valid values
`endeca-server-ws-port`	Specifies the Web services port for the Endeca Server where you want to provision data. The default value is the default Endeca Server Web services port.	7002	Valid port numbers
`endeca-server-context-root`	Specifies the WebLogic application root context of the Endeca Server.	/endeca-server	Valid root context names in WebLogic
`endeca-server-data-domain-profile`	Specifies the name of the Endeca data domain profile that will be used to create data domains when data is uploaded through the Provisioning Service.	default	Valid data domain profile names. Oracle recommends that you define one or more data domain profiles that match your needs. See *Specifying the data domain profile on page 10* for details.
`endeca-server-security-enabled`	Specifies whether the Provisioning Service should use SSL to communicate with Endeca Server.	true	• true The Provisioning Service uses SSL to communicate with Endeca Server. • false The Provisioning Service does not use SSL to communicate with Endeca Server.
`upload-file-directory`	Specifies the directory on the Provisioning Service host machine where uploaded files will be stored. The directory is relative to the domain home directory.	Operating system temporary directory.	./ps_files

Property name	Description	Default value	Valid values
`message-driven-bean-limit`	Specifies the maximum number of message-driven beans allowed in the pool on the WebLogic Server host of the Provisioning Service. The number of message-driven beans defines the number of Excel files that the Provisioning Service can process at each stage of the processing workflow. The Provisioning Service workflow consists of three stages.	2	Positive integers

Deploying the modified configuration to the Provisioning Service

Use the WebLogic deployer to deploy the modified configuration to the Provisioning Service.

Before running the WebLogic deployer tool, set up your environment to run the tool. Add WebLogic Server classes to the CLASSPATH and the JDK binaries to the PATH.

To deploy your modified configuration to the Provisioning Service, run the command:

```
java weblogic.Deployer -deploy –adminurl http:/
/localhost:7001  -username weblogic deploy –source eidProvisioning.ear -plan eidProvisioningConfig
/plan.xml
```

where

- `http://localhost:7001` is the URL of the WebLogic Administration Server
- `weblogic` is the Weblogic Server admin user name
- `eidProvisioning.ear` is the Provisioning Service web application
- `eidProvisioningConfig/plan.xml` is the deployment plan.

For additional details about deploying a deployment plan to WebLogic Server, see *http://docs.oracle.com/cd/E23943_01/web.1111/e13702/wldeployer.htm* and *http://docs.oracle.com/cd/E23943_01/web.1111/e13702/deploy.htm#i1024294*.

Configuring the Endeca Server to provision

The default configuration of the Provisioning Service defines a connection to an Endeca Server running on the same machine and WebLogic server instance as the Provisioning Service, and listening on the default Endeca

Server port. Standard practice in production environments, however, is to run Endeca Servers on different machines from the Provisioning Service.

Each Provisioning Service provisions one Endeca Server with data uploaded by users.

To configure the Provisioning Service to connect to an Endeca Server other than the default, modify the following configuration properties:

- `endeca-server-host-name`

  The `endeca-server-host-name` property specifies the name or IP address of the host machine where you run the Endeca Server that you want to provision.

- `endeca-server-ws-port`

  The `endeca-server-ws-port` property specifies the Web services port on which the Endeca Server listens. The default Endeca Server Web services port is `7001`. The default secure port is `7002`. If you use a different port for your Endeca Server, change the value of this property to match the port you use.

- `endeca-server-context-root`

  The `endeca-server-context-root` property specifies the application root context of the Endeca Server on the WebLogic container. The default root context is `/endeca-server`. If you install Endeca Server to a different root context, change the value of this property to match the root context where Endeca Server is installed.

Specifying the data domain profile

When a user uploads a file, the Provisioning Service creates a new data domain to store and process the uploaded data.

Endeca Server uses data domain profiles to define the characteristics of the data domains it creates. Use the `endeca-server-data-domain-profile` configuration property to specify the data domain profile that the Provisioning Service will use to create new data domains when users upload files. For additional information about data domain profiles, see “Managing Data Domain Profiles” in the *Oracle Endeca Server Cluster Guide*.

The data domain properties defined by the data domain profile affect the performance of the applications that use that data domain, in particular query performance. The default Provisioning Service configuration uses the default data domain profile on the Endeca Server. The default data domain profile defines the properties of a non-clustered data domain on a single-node Endeca Server instance. Therefore, you will likely find that you need to define your own data domain profile for data domains created by the Provisioning Service.

Data domain profile recommendations

Oracle recommends the following configurations for a data domain profile to support the Provisioning Service:

- `allowQueriesOnLeader`

  See "Data Domain profile parameters" in the *Oracle Endeca Server Cluster Guide* for recommendations.

- `readOnly`

  Set to `NO`

- `numFollowers`

  If you want to use a single dgraph process, set to `0`. Otherwise, see "Data Domain profile parameters" in the *Oracle Endeca Server Cluster Guide* for recommendations.

- `allowOversubscribe`

  Set to `true`.

- `numComputeThreads`

  To determine the correct value, use the following formula: `(core count of the Endeca Server host/ anticipated number of user-created applications) * 2`

- `computeCacheSizeMB`

  See "Data Domain profile parameters" in the *Oracle Endeca Server Cluster Guide* for recommendations.

- `sessionIdType`

  Set to `header`. (Note: This is the default value.)

- `sessionIdKey`

  Set to `sessionIdKey`. (Note: This is the default value.)

Configuring storage of uploaded files

Each file uploaded is stored on the file system of the Provisioning Service.

The default location of uploaded files is the temporary directory defined for the operating system. The system temporary directory is usually not a very secure directory, and scripts may modify the contents of this directory, so the recommended practice is to store file uploads in a different directory.

Use the `upload-file-directory` configuration property to specify the path to the directory where you want to store uploaded files. Paths are defined relative to the root path of the WebLogic domain to which the Provisioning Service application is deployed. The specified directory must be writeable for the user that runs the WebLogic server.

For example, if you specify a value of `ps-file` for this property, you must create the `ps-file` directory and configure its permissions to allow write permissions to the WebLogic Server user.

Managing memory usage

You may notice reduced performance when loading large files. In that case, you may want to modify the memory usage of the Provisioning Service.

Two configuration settings control memory usage in the Provisioning Service:

- The maximum number of Message-Driven Beans (MDBs) allowed in the pool.

  The maximum number of Message-Driven Beans for the Provisioning Service is controlled by the `message-driven-bean-limit` parameter in `plan.xml`. The default value of this variable is 2.

- The Java heap space. Java heap space is controlled by the `USER_MEM_ARGS=` parameter in the file `$DOMAIN_HOME/bin/setDomainEnv.sh` in Linux, or `$DOMAIN_HOME/bin/setDomainEnv.cmd` in Windows.

  In the line

  ```
  USER_MEM_ARGS="-Xms256m -Xmx6144m -XX:PermSize=64M -XX:MaxPermSize=256M"
  ```

  change `-Xmx6144m` to the amount of heap space you want to use. The default heap space is 6GB.

These two configuration settings are related. If you specify a larger pool of MDBs, you should also specify more heap space for the Provisioning Service. Conversely, if you reduce the amount of heap space, you should reduce the number of MDBs in the pool.

The starting estimate is 3 GB of heap space per unit in the pool. Thus, the default pool size of 2 calls for 6GB of heap space. As you use the Provisioning Service, you can fine tune these estimates to match your own needs and experience.

Chapter 2

Configuring logging for the Provisioning Service

The Provisioning Service uses standard WebLogic logging functionality.

About logging in the Provisioning Service

By default, the Provisioning Service logs all errors and warnings to the domain log broadcaster, to the server log, or to standard out, depending on the configuration of the WebLogic Server.

For details about configuring logging in WebLogic Server, see *Configuring Log Files and Filtering Log Messages for Oracle WebLogic Server*.

Changing the logging level for the Provisioning Service

When resolving issues, technical support may request that you modify the logging level of the Provisioning Service.

The Provisioning Service installation package you download from Oracle includes a `logging.properties` file you can use when configuring the logging level of your Provisioning Service. This file is stored in the `eidProvisioningConfig` directory under your domain.

To change the logging level of the Provisioning Service:

1. Modify the logging configuration of the WebLogic server to filter the correct level of logging messages (FINE or FINEST).

   For details about the WebLogic server logging configuration, see *Configuring Log Files and Filtering Log Messages for Oracle WebLogic Server*.

2. Restart the domain using `eidProvisioningConfig/logging properties`. Add the following line to setDomainEnv.sh or setDomainEnv.cmd:

   ```
   set JAVA_OPTIONS=$JAVA_OPTIONS -Djava.util.logging.config.file=$DOMAIN_HOME
   /eidProvisioningConfig/logging.properties
   ```

   where `$DOMAIN_HOME` is the path to domain home directory of the Provisioning Service.

Logging messages from the Provisioning Service are listed in the log file under the `ProvisioningLogger` class, as illustrated in the following example:

```
<Dec 21, 2012 12:45:40 PM EST> <Warning> <com.oracle.endeca.pdi.logging.ProvisioningLogger>
<BEA-000000> <OurMessage>
```

You can search and filter the logs to find this class so you can easily find logging messages from the Provisioning Service.

Example Provisioning Service logging.properties file

The code in this topic illustrates an example `logging.properties` file for the Provisioning Service.

```
# Specify the handlers to create in the root logger
handlers = weblogic.logging.ServerLoggingHandler
# Register handlers for the com.oracle.endeca.pdi and its child loggers
com.oracle.endeca.pdi.logging.ProvisioningLogger.handlers = weblogic.logging.ServerLoggingHandler
# Do not send the com.oracle.endeca.pdi log messages to the root handler
com.oracle.endeca.pdi.logging.ProvisioningLogger.useParentHandlers = false
# Set specific LogLevel
com.oracle.endeca.pdi.logging.ProvisioningLogger.level = FINE
```

The logging properties file defines the logging level following the JDK specification. Use the following logging levels:

- FINE (sets Debug logging level in WebLogic server)
- FINEST (sets Trace logging level in WebLogic server)
- ALL (sets All logging level in WebLogic server)

Index