Agile
Version e6.1

ORACLE

Oracle® Agile Engineering Data Management

Administration Manual for Agile e6.1.2

Part No. E20355-01

April 2011

Copyright and Trademarks

CONTENTS

Preface

The Oracle documentation set includes Adobe® Acrobat™ PDF files. The Oracle Technology Network (OTN) Web site (http://www.oracle.com/technology/documentation/agile.html) contains the latest versions of the Oracle Agile EDM PDF files. You can view or download these manuals from the Web site, or you can ask your Agile administrator if there is an Oracle Documentation folder available on your network from which you can access the documentation (PDF) files.

| **N o t e** | To read the PDF files, you must use the free Adobe Acrobat Reader™ version 7.0 or later. This program can be downloaded from the Adobe Web site (http://www.adobe.com). |

| **N o t e** | Before calling Agile Support about a problem with an Oracle Agile EDM manual, please have the full part number, which is located on the title page. |

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Readme

Any last-minute information about Oracle Agile EDM can be found in the Release Notes file on the Oracle Technology Network (OTN) Web site (http://www.oracle.com/technology/documentation/agile_eseries.html)

Agile Training Aids

Go to the Oracle University Web page (http://www.oracle.com/education/chooser/selectcountry_new.html) for more information on Agile Training offerings.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Introduction

The Administration Guide describes how to administrate the installed Agile e6.1.2 components.

About Agile e6.1.2 Administration

An Agile e6.1.2 application/environment defines a particular hardware and software configuration that allows Agile e6.1.2 to work with an Oracle database and Oracle WebLogic Server.

In this documentation we will use the word "application" instead of "environment".

To administer the Agile e6.1.2 application, you can create and manage Agile e6.1.2 applications. For each Agile e6.1.2 application that you create, information is stored on the Agile e6.1.2 server that specifies how to connect to the database and locate data and which oracle application server to use.

When you install an Agile e6.1.2 server, the installation program automatically creates a default application which is configured during the installation, and the application name is associated with the database.

It is also possible to create additional applications within the same installation, thus it is not necessary to maintain separate installations for different purposes. A single Agile e6.1.2 installation often includes separate applications for various purposes, such as testing (for testing purposes before creating the actual working applications), development, production, and education.

The applications created for an installation can be associated with the same or with different databases (e.g. a multi-environment system, in which development and test environments, and their respective databases, co-exist with a production environment and its database). In general it is required to have a separate production infrastructure to avoid the risk of loss of production by changing something in e.g. the development environment.

About Agile e6.1.2 Business Services Administration

When you create an Agile e6.1.2 application, the Business Services component will be installed/deployed on your oracle WebLogic server for each new application. The Business Services comprise of the Workflow Module, Permission Manager and Product Configurator.

Overview of the Workflow Module

To include Workflow processes in an Agile e6.1.2 application, optional configuration parameters can be defined for the processes within the overall Agile e6.1.2 application definition. For more information, refer to the chapter Administering Agile e6.1.2.

Note   For further information on using the Workflow module, refer to the *Agile e6.1.2 Online Help > Using Agile e6 > Product Data Management > Workflow*.

Modifying Configurations without the Administration Client

Any file or configuration modification made without the Administration Client will be lost by changing the application values with the Administration client again except using the component based J2EE installation.
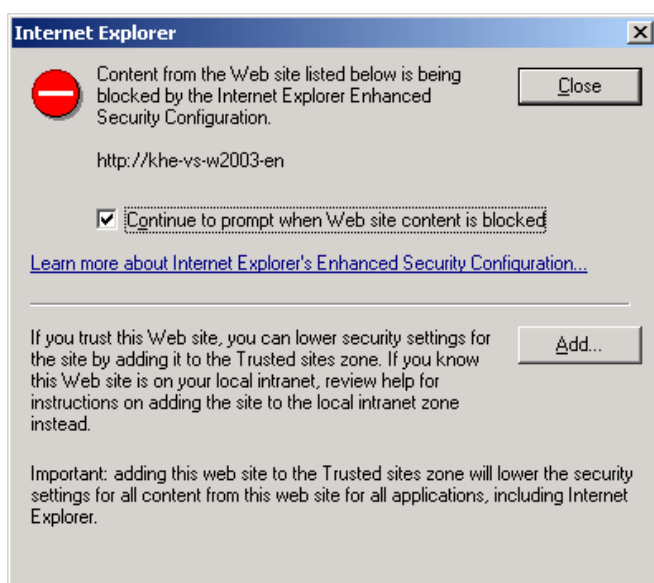
If you want to install/deploy the Agile e6.1.2 J2EE components with a separate user or on a separate server different from the one used for the "native" Agile e6.1.2 installation, it is necessary to perform also a separate e6 installation of the J2EE components with the separate user or on the separate server.

<div align="right">

Chapter 2

Administering Agile e6.1.2

</div>

The Agile e6.1.2 Administration client lets you create, configure, and delete Agile e6.1.2 applications via the web browser. Agile recommends that you use a supported browser (see Prerequisites Guide for Agile e6.1.2) to connect to the Administration client.

> **Note** If the following window opens, you might have to add the Administration client to the trusted web sites in your browser settings.

The Administration Client

The Administration client will be installed automatically with the Installer if you install a server component. It can be used to create new or modify existing applications. The Administration client is a web based application and is deployed in an Apache Tomcat servlet container. For detailed information on Apache Tomcat servlet container refer to http://tomcat.apache.org/ . Currently we are using Apache Tomcat v6.0.26.

You can find Apache Tomcat in:

- **Windows**: %ALLUSERSPROFILE%\agile\installer\6.1.2\admin
- **Unix**: $HOME/agile/installer/6.1.2/admin.

Starting the Administration Client

Windows: Start the service "Apache Tomcat AdminClient" and set the startup type to "Automatically".

Unix: Use the standard Apache Tomcat startup and stop scripts.

A 32bit Java 6 version has to be used. On most new Unix servers the default Java version is a 64bit Java.

To set the 32 Bit Java 6 for Tomcat:

1. Create the file "setenv.sh" in the $HOME/agile/installer/6.1/admin/apache-tomcat/bin directory of Tomcat with a line like:

    ```
    export JRE_HOME=/usr/local/java/jdk1.6/jre
    ```

2. Tomcat will automatically use this Java version which must point to a 32 bit Java.

3. Start Tomcat with "startup.sh". Stop Tomcat with "shutdown.sh".

    Tomcat will show the used Java version at startup with something like:

    ```
    Using JRE_HOME: /usr/local/java/jdk1.6/jre
    ```

Changing the Password

The Administration client has a default password which has to be changed after the Agile e6.1.2 installation.

□   The encrypted password for the Admin client can be found in:

*   In Windows:

    ```
    %ALLUSERSPROFILE%\agile\installer\6.1.2\admin\apache-
    tomcat\webapps\AdminClient\metadata\Adminserver_Props.txt
    ```
*   In Unix:

    ```
    $HOME/agile/installer/6.1.2/admin/apache-
    tomcat/webapps/AdminClient/metadata/Adminserver_Props.txt.
    ```

□   To create a new password, execute:

*   In Windows:

    ```
    %ep_root%/axalant/cmd/epkeytool.bat -encryptpwd -pass <password>
    -keyStore cwallet.sso -keyAlias
    "C=DE,ST=Baden,L=Karlsruhe,O=Oracle,OU=Agile PLM,CN=PLM"
    ```
*   In Unix:

    ```
    $ep_root/axalant/scripts/epkeytool.sh -encryptpwd -pass
    <password> -keyStore cwallet.sso -keyAlias
    "C=DE,ST=Baden,L=Karlsruhe,O=Oracle,OU=Agile PLM,CN=PLM"
    ```

The resulting output is your encrypted password. Change the property "password" in the "Adminserver_Props.txt" file with the newly generated password, and restart the Apache Tomcat process.

Changing the Apache Tomcat Configuration

Tomcat is configured with a HTTP connector by default.

Note   The Administration client uses port 8030 for the HTTP connector.
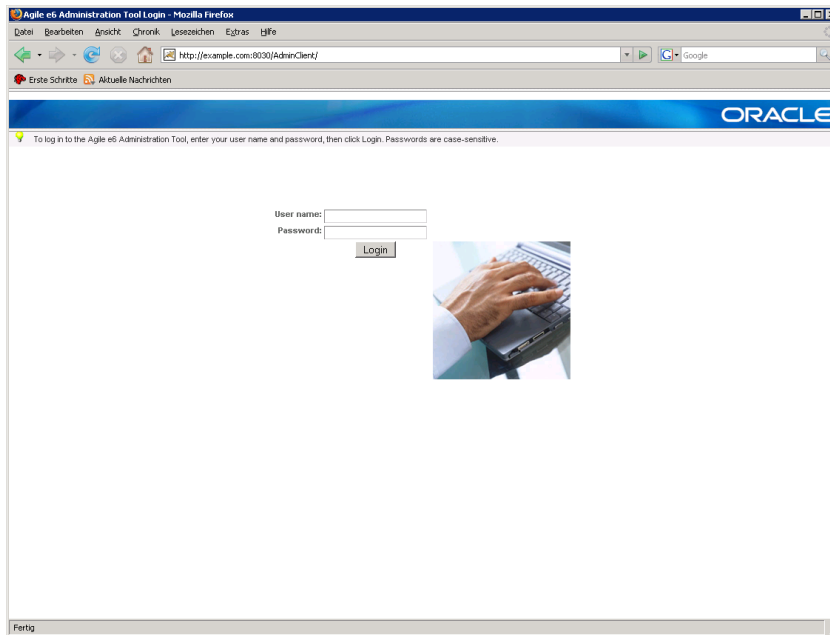
If you want to change this port you have to edit the Apache Tomcat "conf/server.xml" file and restart the Apache Tomcat process. If you want to enable HTTPS for the Administration client please refer to http://tomcat.apache.org/tomcat-6.0-doc/ssl-howto.html for more information.

Administering with the Administration Client

Logging In/Logging Out

1. Access the Administration client via http://<servername>:8030/AdminClient.

    The *login* page appears.

2. Log in with the following parameters:
    *   User: plm
    *   Password: e.g. "ChangeOnInstall" or your new generated password

    The Welcome page appears.
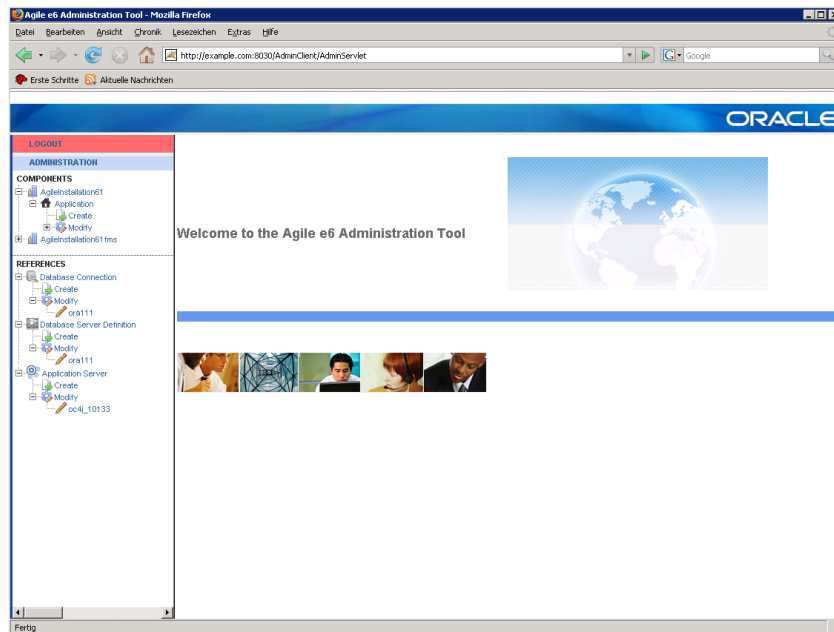
The navigation at the left hand side shows all available installations that can be configured (under 'Components').

Under 'References' the available database clients, databases, and application servers can be found.
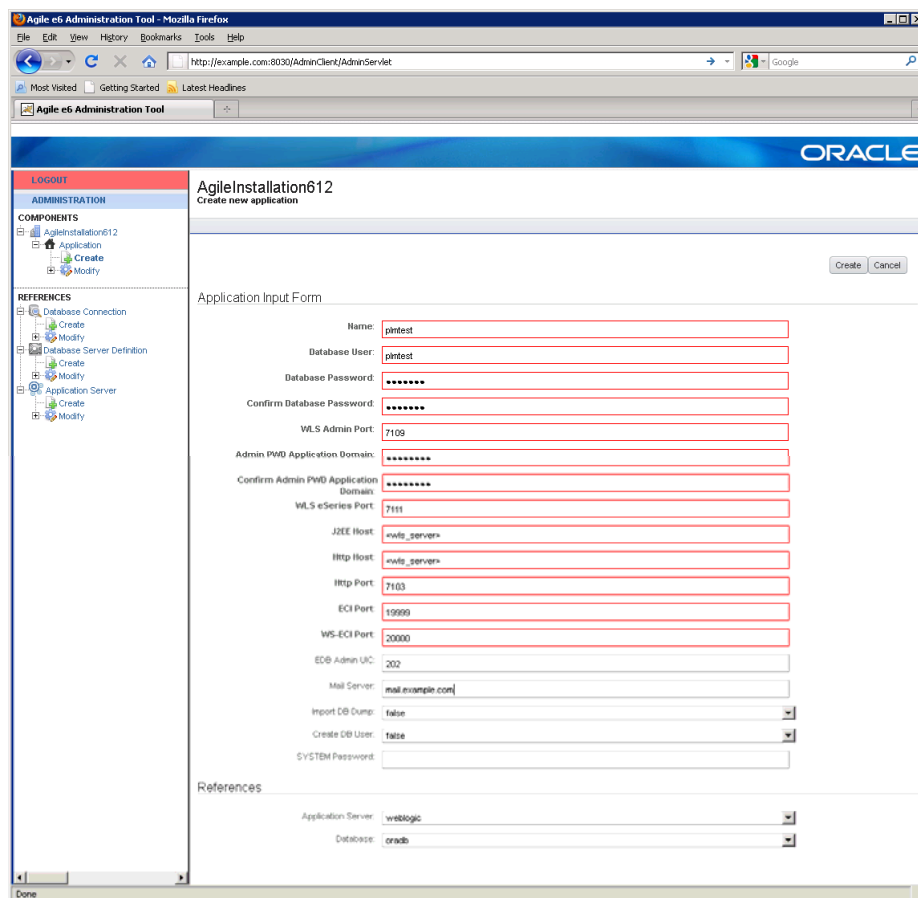
3. To log out click the "Logout" link in the upper left corner.

Note   The Administration client times out in 10 minutes. Then you have to log in again.

Creating/Modifying an Application

1. Under <installation name> -> Application click create.

The *Application Input Form* page appears.

2. Create a new application with the following parameters:

Application Input Form

Setting	Description
Name	The name of the default application you want to create (former axalantORIGIN)
Database User	The database user you want to use for the application.
Database Password	The password of the database user.
Verify Password	The password entered above has to be repeated for the verification.
WLS Admin Port	The listen port for the WebLogic administration server which will contain the WebLogic administration console
Admin PWD Application Domain	The password of the WebLogic user used for the application domain
Verify Password	The password entered above has to be repeated for the verification.
WLS eSeries Port	The listen port for the WebLogic eSeries server. The installer will create a separate managed server with the name "eSeries-01" in addition to the administration server. This server will contain the application specific deployments.

Setting	Description
J2EE Host	The host where WebLogic is running (the WebLogic server host name).
Http Host	The host where the Web client can be reached over http (the WebLogic server hostname).
Http Port	The port where the Web client can be reached over http (the HTTP port of WebLogic server).
ECI Port	The port where the business service will be configured to "wait" for requests.
WS-ECI Port	The port where the Webservices will be configured to "wait" for requests.
Admin UIC	The UIC of an Agile e6.1.2 user to whom administrative emails should be sent.
Mail Server	If you want to send emails with the business services, define the name of your SMTP mail server. Otherwise use "nn".
Import DB Dump	Select if you want the standard dump to be imported to the above defined database user.
Create DB User	Select if you want the database user to be created automatically.
SYSTEM Password	Password of the Oracle database user. Note You only have to apply a value here if you want to create a new database user and selected this in the field above.

References

In the References section you can define which database and application server should be used for this application. Normally, you don't have to change these values.

Setting	Description
Application Server	The name of the reference of the Oracle WebLogic Server where Webservices and BusinessService will be deployed.
Database	The name of the reference of the database you want to use for the application.

Note   Creating a new application can take up to 20 minutes, depending on if a database user will be created, the database dump for the application has to be imported, the domain creation and setup.

3. Click **Create**.

Updating and Deleting an Application

1. To update an application click Application -> Modify.

    The following page opens:
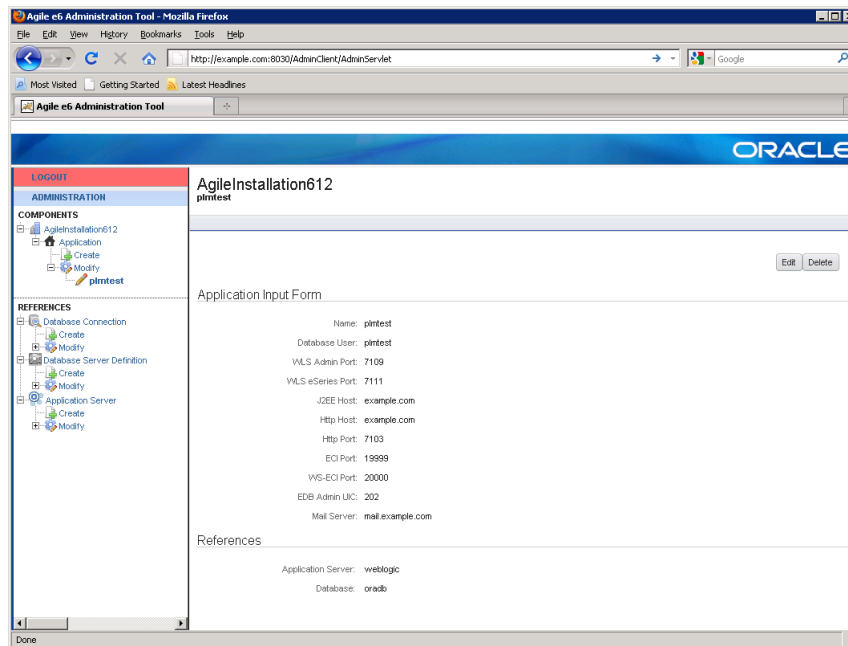
Here you can see the current values of your application. It is possible to delete the application with the "Delete" button in the upper left corner.

Note Deleting an application will not remove the domain for web services/business service from the Oracle WebLogic Server. This must be done manually. Also, the used database schema remains unchanged.

2.   Click "Edit" Button in the upper left corner to change the values for the application.

The following page opens:

To apply the changes, press the "Update" button in the upper right corner. The admin server will delete your current application and create a new one with the new values. Also the web services/business services will be redeployed with the new values.

Note Updating a production application that is in use and running is NOT recommended.

Setting and Changing Initial User Passwords in a New Agile e6.1.2 Application

Note The enhanced security module is enabled in each newly created application (with a new dump) by default. You can only log in to this application with the user "manager"!

Perform the following steps:

1. Start an e6.1.2 client and log in to your application with the user "manager" with password "manager".

   For the first log in you are asked to set a new password for this user. This will be the password for the future logins.

2. Set a new password for the user.

All other users are deactivated and have to be activated by setting a new valid password for them.

Set the password for user "EDBCUSTO" and all other standard users

1. Open the User List: Manager > Permissions > User > Basic Data.

2. Set the password: Refresh > Select "EDBCUSTO" > Context Menu: Set Password.

Note At the first log in of user "EDBCUSTO" with the initial password the user will be asked to set a new password. This will be the password for the future logins.

Repeat these steps for all the newly created users and standard users.

<div align="right">

Chapter 3

Managing References

</div>

Adding a New Database for an Application

Requirements for a new database:

1. A new database connection.

2. A new database server - This will be connected over the database connection.

3. SQLNet – This has to be configured already and working.

To create a new database:

1. In the left pane, click "Create" under References > Database Connection.

   This following page appears:
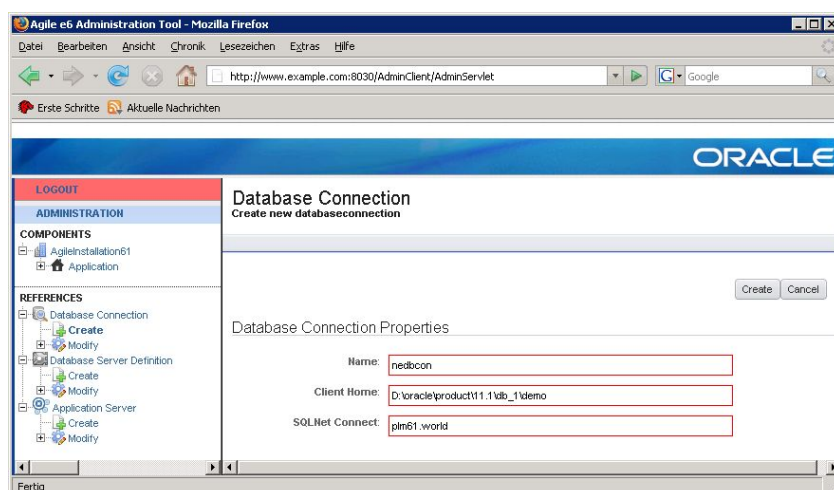
2. Create a new database connection with the following parameters:

Setting	Description
Name	The new name which identifies the connection
Client Home	The Oracle Client Home directory
SQLNet Connect	The SQLNet connect string as defined, e.g. tnsnames.ora

3. Click "Create" button in the upper right corner.

To define a new database server:

1. In the left pane, click "Create" under References > Database Connection.

   This following page appears:
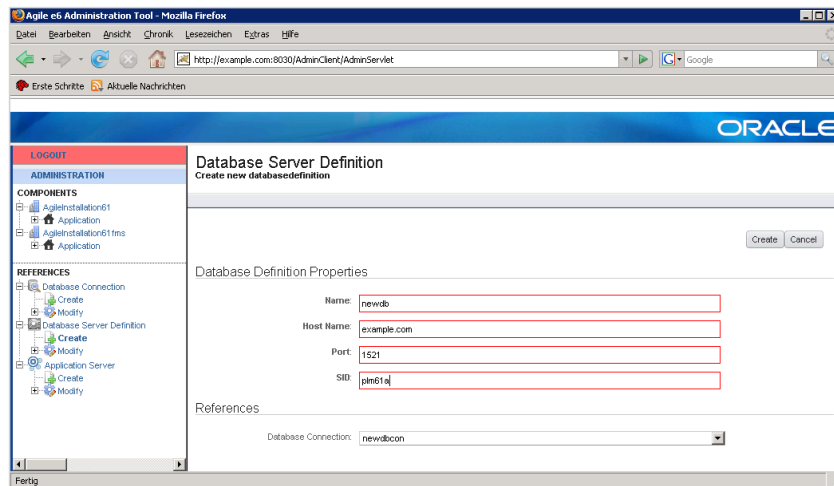
2. Create a new database server with the following parameters:

Setting	Description
Name	The new name which identifies the database definition
Host Name	The host name of your database server.
SID	The SID of the database you want to connect to.
Port	The port where the listener listens on the database server.

3. Click "Create" button in the upper right corner.

Adding a New Application Server for an Application

1. In the left pane, click "Create" under References > Application Server.

   This following page appears:
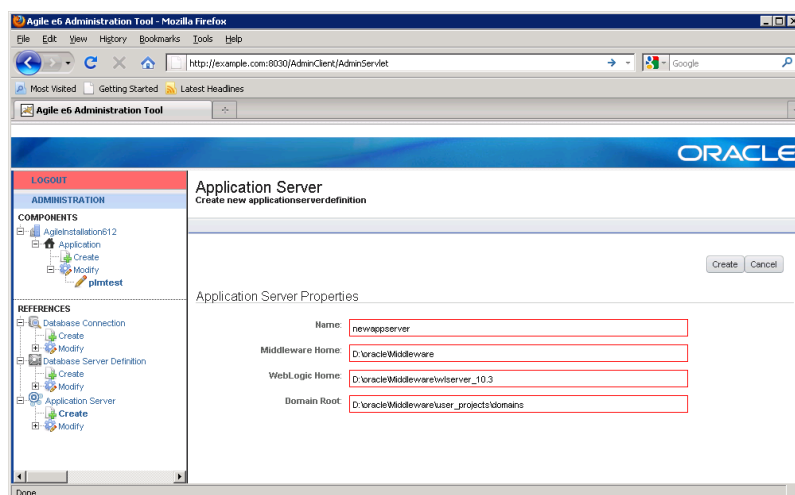
2. Define a new application server with the following parameters:

Setting	Description
Name	The new name which identifies the Oracle WebLogic server definition
Middleware Home	The Middleware home where you want to install WebLogic to. (e.g.: D:\oracle\Middleware)
WebLogic Home	The WebLogic home directory which normally is "inside" your Middleware home (e.g.: D:\oracle\Middleware\wlserver_10.3)
Domain Root	The Middleware home where you want to install WebLogic (e.g.: D:\oracle\Middleware\user_projects\domains)

3. Click "Create" button in the upper right corner.

Deleting a Reference

To delete a reference:

1. In the left pane, expand the "Modify" node of the corresponding reference (Reference database connection, database server definition or application server) and select the name of the reference.

2. Click "Delete" in the upper left corner.

   This will only delete the reference in the "installation_prop.xml" file if the reference is not in use. If it is in use, an error message is displayed.

   Note Nothing else will be deleted.

<div align="right">**Chapter 4**</div>

Advanced Administration Tasks

Agile e6.1.2 and Oracle WebLogic Server

Note   In this document, replace <installation_name> with your installation name (e.g. AgileInstallation61).

The Oracle WebLogic server is mandatory for Agile e6.1.2. After the Agile e6.1.2 installation, the Web Presentation Service, Java Client Web Start, PLMAPI (HTTP support), the Web Fileservice, Jvue Applet and DaemonAdminServlet deployment have to be available in one WebLogic domain with default name "eSeries_domain". Additionally, for each created application the Business service and Webservices deployment have to be available in one WebLogic domain with default name "eSeries_domain_<application_name>".

The "eSeries_domain" and "eSeries_domain_<application_name>" can be accessed over the following link:

```
http://example.com:7101/console
```

To verify the availability of the above deployments, log in to your Oracle WebLogic domains.

The following applications have to be available in the domain "eSeries_domain":

- □   DaemonAdminServlet

- □   HTTPSupport

- □   JavaClient

- □   JVue

- □   WebPresentationService

- □   WebFileservice

- □   VueLink

The following context roots of these applications should be working state:

Note   Use your Oracle WebLogic server name and http port.

- □   DaemonAdminServlet: http://example.com:7103/DaemonAdminServlet/

- □   HTTPSupport: http://example.com:7103/plm-api-axis/services

- □   JavaClient: http://example.com:7103/Jacc

- □   JVue: http://example.com:7103/JVue/jvue.jar

- □   VueLink: http:// example.com:7103/VueLink/Vuelink

- □   WebFileService: http:// example.com:7103/FileService

□   WebPresentationService: http:// example.com: 7103/AgilePlmWps

The following applications have to be available in the domain "eSeries_domain_<application_name>":

□   BusinessService

□   Web Services

The following diagram shows the WebLogic deployment architecture after an e6 installation.
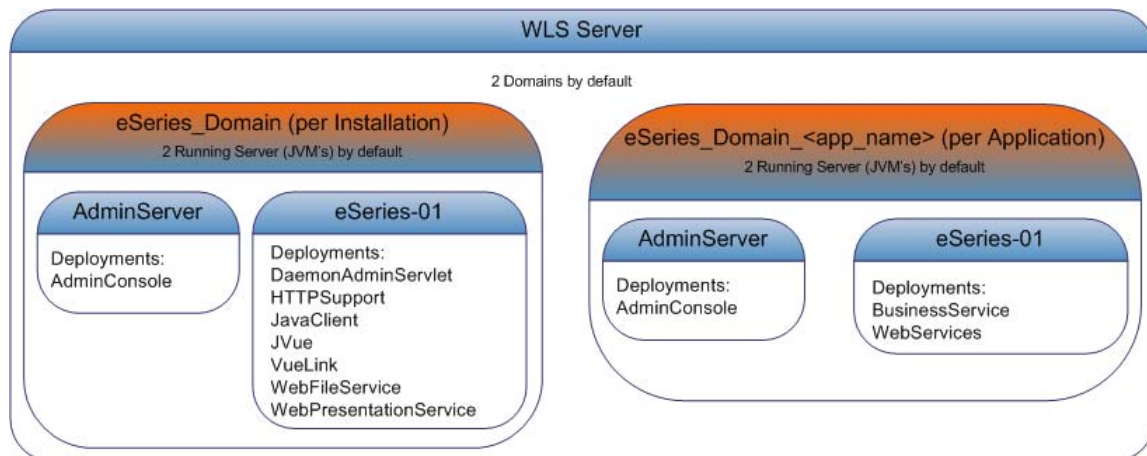
Domains Directory: <MIDDLEWARE_HOME>/user_projects/domains

Server Directories: <DOMAIN_HOME>/servers/AdminServer (eSeries-01)

Further information about the Oracle WebLogic server administration can be found under the Oracle WebLogic Server documentation in the Oracle Fusion Middleware Documentation Library:

http://download.oracle.com/docs/cd/E15523_01/index.htm

See: Product Area -> WebLogic Server

Customer Adaptations Deployment

With e6.1.2 customer adaptations made to files which are part of a deployment, have to be made outside the WebLogic server. These adapted files will be available even after applying a hotfix. Changing e6 files directly in the Oracle WebLogic server deployment is not supported.

The deployment process in e6.1.2 will create a "staging" directory in <ep_root>. This process is active for all described batch deployment tasks. The "staging" directory contains 4 subdirectories.

□   **Merged:** This directory contains the merged result of the "product" and "custom" folder. In this directory structure you can check if your adapted files are available in the unzipped archive. From this directory structure the final archive will be created in the deploy directory. **No customer changes are required here.**

□   **Product:** This directory contains a subdirectory for every deployed component. Under these subdirectories, the unzipped archives are available. If you want to change any file in the archive, the file has to be copied to the "custom" folder of the same directory. The directory structure in the custom folder is the same as in the product folder, but as per

default structure without files.
No customer changes required here.

□ **Deploy**: This directory contains the final archive before it will be deployed.
No customer changes required here.

□ **Custom**: This directory contains the complete directory structure of the product folder, but as per the default structure without files. You will find only the folders without any files that you can modify in the directory structure. The files you want to modify can be taken from the product folder directory structure.
Customers apply changed files here.

Note   It is not possible to delete files which are available in the original archive. Adding more files to the "custom" directory is possible.

Agile e6.1.2 Business Service Administration

Business Services are always installed/deployed if a new application is created with the Administration client. Therefore, the availability of the application server is mandatory.

If you want to perform additional changes to the Workflow module, as described in the Manager Information of the Workflow module in the online help, you have to edit the configuration file for the Business Services.

To modify the configuration file you have to perform the steps described in the "Customer adaptations deployment" section of this document.

Example: Adapt values in "ABS_<application_name>.ini" file of the Business Service.

To modify the configuration file it can be copied from the "product" directory after the installation. The file is available in "<ep_root>\staging\product\BusinessService\ BusinessService.ear\BusinessService.war\WEB-INF\classes".

Save the modified file to the
"<ep_root>\staging\custom\BusinessService\<application_name>.ear\BusinessService.war\WEB-INF\classes" folder.

The file that you modified will be used with the next deployment of the Business Service.

The Business Services will connect to the defined database user/schema directly after they are started in the Oracle WebLogic Server.

Note   Once this connection is lost, say due to database reboot, or due to dropping of the database schema or due to loss of service, you must restart the Business Service inside the Oracle WebLogic Server.

Note   It is also possible to restart the complete Oracle WebLogic server, but all the included services will not be available for that time.

Adapting Business Service Notification Mail Templates

After the first Business Service deployment of a new application the notification mail templates are unpacked and available in

"*<ep_root>\staging\custom\BusinessService\<application_name>.ear\BusinessService.war\WEB-INF\lib\abs-notifier-templates\lay\notifier*".

All adaptations made to files in this directory (also all new files) will be available after the next deployment of the Business Service. All files and directories below the "abs-notifier-templates" directory will be automatically packed to "abs-notifier-templates-1.0-SNAPSHOT.jar" file, which will be used for the deployment.

About PLMAPI / HTTP(S) Support

PLMAPI allows the communication to the Agile e6.1.2 application server through firewalls (via http). It is supported only in the Java client. For further information on the architecture, refer to the 'Architecture Guide for Agile e6.1.2'.

The plmapi is always installed/deployed if a new Agile e6.1.2 server installation is created and can be used for all created applications of the installation. To be able to use the plmapi, it has to be configured in the Java client. To use the plmapi, no changes are required in the configuration of the Oracle Application Server.

For further information on how to set up the HTTP(S) support in the Java client, refer to the 'Preferences' section of in the "Getting Started" section of the online help, and the Architecture Guide.

Special Batch Installation Tasks

Prerequisites

Note   Agile e6.1.2 should be already installed!

For all batch tasks set the JAVA_HOME environment variable to Java6 before you execute setup.cmd/setup.csh in a cmd/csh shell.

Use the installation user of the e6.1.2 installation.

On Windows use an Administrator cmd shell and execute setup.cmd for all batch tasks in:

```
%ALLUSERSPROFILE%\agile\installer\6.1.2
```

In the Unix system, use a csh shell and execute setup.csh for all batch tasks in:

```
${HOME}/agile/installer/6.1.2
```

Properties files (e.g.: "batch.properties") must be created in:

▫   Windows: %ALLUSERSPROFILE%\agile\installer\6.1.2\properties

▫   Unix: ${HOME}/agile/installer/6.1.2/properties

The batch commands are described for windows operating system. On unix operating system use "setup.csh" instead of "setup.cmd" and different redirector settings for stderr and stdout.

Example: For Windows (Administrator cmd):
```
cd %ALLUSERSPROFILE%\agile\installer\6.1.2
set JAVA_HOME=<java6_home>
```

```
setup.cmd application.install -propertyfile
properties/batch.properties > batch_install.log 2>&1
```

Example: For Unix (csh):

```
cd ${HOME}/agile/installer/6.1.2

setenv JAVA_HOME <java6_home>

setup.csh application.install -propertyfile
properties/batch.properties >& batch_install.log
```

Note   After each batch installation task, check the log file.

Create an Application

1. Create the properties file (e.g.: batch.properties) for the application. You can use different batch properties files for each application (example):

```
plm.inst.name=<installation_name>

# application

plm.application.name=mytest

plm.application.j2eehost=<WebLogic_servername>

plm.application.httphost=<WebLogic_http_servername>

plm.application.httpport=7103

plm.application.dbuser=mytest

plm.application.dbpassword=mytest

plm.application.eciport=20000

plm.application.wseciport=20001

# apply the database password unencrypted

plm.application.dbpassword.wlscrypt=mytest

plm.application.adminserver_port=7105

plm.application.eseriesserver_port=7107

plm.application.mailrelay=<Mailserver>

plm.application.adminserver_password=welcome1

# Application server to deploy business service / webservices as
defined in installation_prop.xml

plm.application.applicationserverdefinition=weblogic

# Defines if reference dump will be imported

plm.application.dumpimport=true

# Defines if the schema will be created (system password also has to
be applied for creating a schema)

plm.application.creschema=true

# Defines if the content of an existing schema will be deleted
(Attention: All objects of the defined schema will be lost)?

plm.application.delschemaobj=false

# Database to import as defined in installation_prop.xml (normally no
need to change this)

plm.application.databasedefinition=oradb
```

```
plm.oracle.systempwd=<systempwd>
```

2. After defining the properties, execute (example):

```
setup.cmd application.install -propertyfile
properties/batch.properties > application_install.log 2>&1
```

This will create the application directory and the configuration files in the <ep_root>/init directory. Additionally, if enabled, the schema will be created and the reference dump will be imported.

To deploy the j2ee applications for an e6 application the additional steps need to be executed.

1. Create J2EE application domain (example):

```
setup.cmd j2eeappserver.install.domain_app -propertyfile
properties/batch.properties > install_domain_app.log 2>&1
```

2. Deploy J2EE applications (BusinessService, WebServices) to domain (example):

```
setup.cmd j2eeappserver.deploy.app -propertyfile
properties/batch.properties > deploy_app.log 2>&1
```

(Re)Deploying Business Services for an Application

Redeploying a Business Service causes a recreation of the Business Service 'ear' file and the deployment with the configured values as defined the installation_prop.xml file.

1. Create Properties file (example):

```
plm.inst.name=<installation_name>

plm.application.name=mytest

plm.application.dbpassword.wlscrypt=mytest

plm.application.adminserver_password=welcome1
```

2. Redeploy BusinessService (example):

```
setup.cmd j2eeappserver.deploy.businessservice -propertyfile
properties/batch.properties > deploy_businessservice.log 2>&1
```

(Re)Deploying WebServices for an Application

1. Create the Properties file (example):

```
plm.inst.name=<installation_name>

plm.application.name=mytest

plm.application.adminserver_password=welcome1
```

2. Redeploy the BusinessService (example):

```
setup.cmd j2eeappserver.deploy.webservices -propertyfile
properties/batch.properties > deploy_webservices.log 2>&1
```

(Re)Deploying JavaClient

1. Create the Properties file (example):

```
plm.inst.name=<installation_name>

plm.j2eeappserver.adminserver_password=welcome1
```

2. Redeploy (example):

```
setup.cmd j2eeappserver.deploy.javaclient -propertyfile
```

```
properties/batch.properties > deploy_javaclient.log 2>&1
```

(Re)Deploying WebPresentationService

1. Create the Properties file (example):

    ```
    plm.inst.name=<installation_name>
    plm.j2eeappserver.adminserver_password=welcome1
    ```

2. Redeploy (example):

    ```
    setup.cmd j2eeappserver.deploy.webpresentationservice -propertyfile
    properties/batch.properties > deploy_wps.log 2>&1
    ```

(Re)Deploying WebDevelopmentToolkit

WebDevelopmentToolkit is not deployed after default installation. If you want to use it you have to first deploy it, as described in this section. This can also be used then to redeploy the WebDevelopmentToolkit.

1. Create the Properties file (example):

    ```
    plm.inst.name=<installation_name>
    plm.j2eeappserver.adminserver_password=welcome1
    j2eeappserver.deploy.webdevelopmenttoolkit=true
    ```

2. Redeploy (example):

    ```
    setup.cmd j2eeappserver.deploy.webdevelopmenttoolkit -propertyfile
    properties/batch.properties > deploy_wdt.log 2>&1
    ```

(Re)Deploying HTTPSupport (plmapi)

1. Create the Properties file (example):

    ```
    plm.inst.name=<installation_name>
    plm.j2eeappserver.adminserver_password=welcome1
    ```

2. Redeploy (example):

    ```
    setup.cmd j2eeappserver.deploy. httpsupport -propertyfile
    properties/batch.properties > deploy_httpsupport.log 2>&1
    ```

(Re)Deploying WebFileService

1. Create the Properties file (example):

    ```
    plm.inst.name=<installation_name>
    plm.j2eeappserver.adminserver_password=welcome1
    ```

2. Redeploy (example):

    ```
    setup.cmd j2eeappserver.deploy.webfileservice -propertyfile
    properties/batch.properties > deploy_wfs.log 2>&1
    ```

(Re)Deploying JVue (AutoVue Applet)

1. Create the Properties file (example):

    ```
    plm.inst.name=<installation_name>
    plm.j2eeappserver.adminserver_password=welcome1
    ```

2. Redeploy (example):

```
setup.cmd j2eeappserver.deploy.jvue -propertyfile
properties/batch.properties > deploy_jvue.log 2>&1
```

(Re)Deploying Vuelink (AutoVue DMS Servlet)

1. Create the Properties file (example):

```
plm.inst.name=<installation_name>
```

```
plm.j2eeappserver.adminserver_password=welcome1
```

2. Redeploy (example):

```
setup.cmd j2eeappserver.deploy.vuelink -propertyfile
properties/batch.properties > deploy_vuelink.log 2>&1
```

(Re)Deploying DaemonAdminServlet

1. Create the Properties file (example):

```
plm.inst.name=<installation_name>
```

```
plm.j2eeappserver.adminserver_password=welcome1
```

2. Redeploy (example):

```
setup.cmd j2eeappserver.deploy.daemonadminservlet -propertyfile
properties/batch.properties > deploy_admsvlt.log 2>&1
```

Chapter 5

RAC Support

This section describes the needed modifications to an Agile e6.1.2 installation including deployed Business Services to support a RAC database. It is recommended to do the RAC database setup and e6.1.2 adaptations with the help of an oracle consultant.

Prerequisites

1. RAC database is running. Oracle consulting has reviewed and approved the RAC database installation.

2. Requisite changes in tnsnames.ora, sqlnet.ora on the oracle client side where the e6 server is running are carried out.

3. Oracle WebLogic Server is installed and running.

4. Install e6.1.2 as described in the installation manual for single database, only changing the following:

   - In the "Reference Configuration" mask you have to provide the following entries:
     - As "SQLNet Connect" add the net alias for your RAC database in tnsnames.ora
     - As SID select the SID of one of the instances of your RAC database.
     - As "Host Name" add the IP Address/hostname where this instance is running.

   The following picture shows an example configuration.

5. Continue the installation as it is described in the installation manual for Agile e6.1.2.

Business Service Limitation

▫ The Business Service is using jdbc driver to connect to the database. In order to benefit from TAF (Transparent Application Failover) feature of RAC database, jdbc driver must be jdbc oci driver. As Oracle Weblogic 10.3.3 supports only jdbc thin drivers, Business Service can't be configured to use TAF and will work as it is using single database instance. It will connect only to the instance provided in the "Reference Configuration" mask. If this instance is down, the application will continue to work normally using any other available instance of the RAC database, but Business Service will not work.

<div align="right">

Chapter 6

Java Client

</div>

Predefined Java Client Connection Settings

When starting the Agile PLM Client the client following connection settings can be passed

```
-a <Application Name>
-h <Host Name>
-d <Daemon number>
-p <ECI port>
-u <PLM Username>
```

Java Client Native Installation Windows example:

1. Open properties for the shortcut "Agile e6.1 Java Client.lnk" in "C:\Documents and Settings\All Users\Start Menu\Programs\"

2. At the end of the "Target" Attribute add connection setting you want to use e.g.:
   -a plmref -h example.com

3. It is also possible to edit the "jacc.cmd" file in "C:\Documents and Settings\All Users\Application Data\Oracle\Agile\EDM" and to add the setting there. Search the line which ends with

   …com.agile.jacc.e6.Jacc %*"

   and replace this with e.g.:

   "…com.agile.jacc.e6.Jacc %* -a < Application Name > -h <Host name>…"

Example: Java Client Web Start:

1. If not yet available, copy "<ep_root>\staging\product\JavaClient\jacc.defaults" to "<ep_root>\staging\custom\JavaClient\jacc.defaults"

2. Adapt the properties you need in "<ep_root>\staging\custom\JavaClient\jacc.defaults":

   jacc.node, jacc.port, jacc.app

Java Client DFM side definition

To configure the Java Client to start with a specific DFM side you have several options which are read in the order given here. The first occurrence will set the variable EP_DDM_SITE:

1. Edit the "jacc.cmd" file in "C:\Documents and Settings\All Users\Application Data\Oracle\Agile\EDM" and to add the setting there. Search the line where "javaw" is executed and add e.g.: -DEP_DDM_SITE="ka". The line looks like:

   ""%JAVA_HOME%\bin\javaw.exe" %VM_OPTS% -Djacc.home="$APPDATA\e61" -DEP_DDM_SITE="mu" …."

2. If not yet available copy "<ep_root>\staging\product\JavaClient\jacc.jnlp" to "<ep_root>\staging\custom\JavaClient\jacc.jnlp"

3. Open "<ep_root>\staging\custom\JavaClient\jacc.defaults \jacc.jnlp" add the line

```
<property name="EP_DDM_SITE" value="sitejnlp"/>
```

4. Add the entry "EP_DDM_SITE=ka" to $APPDATA\Agile\e61\jacc.ini.

   For example "C:\Documents and Settings\<username>\Application Data\Agile\e61\jacc.ini".

   The jacc.ini file is available after the first execution of the JavaClient. Do not edit the file while the client is running or your setting will be overwritten.

5. If not yet available copy "<ep_root>\staging\product\JavaClient\jacc.defaults" to "<ep_root>\staging\custom\JavaClient\jacc.defaults"

6. Open "<ep_root>\staging\custom\JavaClient\jacc.defaults" and add a line: EP_DDM_SITE=ka

7. Set the environment variable "EP_DDM_SITE=ka"

   Note   Step 1 works only for Java Client Native installations,

   Note   Step 2 works only for Java Client Web start installations.

8. By applying multiple jnlp files on the oracle application server with different site entries it is possible to provide a web start environment for different EP_DDM_SITE's with one single deployed Java Client.

9. To verify the current Java Client EP_DDM_SITE setting, create a LGV Procedure with the following line:

```
put (client_env ("EP_DDM_SITE"))
```

10. Execute the LGV procedure. If the setup is correct you will see the defined site as output in your client message window.

Display Images with the Webstart Java Client

In case you want to display your own images in the Webstart Java Client, make sure the images or icon names are all written in lower case as otherwise they won't be displayed correctly. The Webstart Java Client gets the images from the Webserver in which the files are case sensitive.

Secure Environment (HTTP(s) Support)

Prerequisites

Before starting to setup a secure environment make sure your standard installation as described in the installation documentation works without issues.

In the secure environment we will just make some modifications to a standard environment. To configure the secure environment you need a certificate from a trusted certificate authority.

In this example scenario a self signed certificate is used. The example scenario uses windows operating system. Self signed certificates require adding security exceptions to browsers and java virtual machines to accept these certificates. Certificates from a trusted certificate authority do not require these security exceptions.

Note   Setup of a production secure environments needs expertise in network security setup. The following setup is only a simple example which shows the e6 requirements. This must be extended for production use by customer experts with their networking security infrastructure requirements.

Secure External Communication

In Agile e6.1.2 it is possible to setup external access to the Agile e6.1.2 environment with the Java client or a Web Browser using the Web Client. It is possible to setup the complete external communication over internet using the HTTPS protocol. Internal Intranet communication will be still over HTTP and RPC calls.

This section describes how to setup a secure environment for this use case.

Secure External communication means the communication in the Internet, From a Java/Web client over the internet to a server in the DMZ which acts as the "End Point" for the Java/Web client. In our case the "End Point" is apache httpd, configured as reverse proxy in the DMZ.

Certificate needs to be installed for apache in the DMZ in this scenario.

Following picture illustrates an example communication.
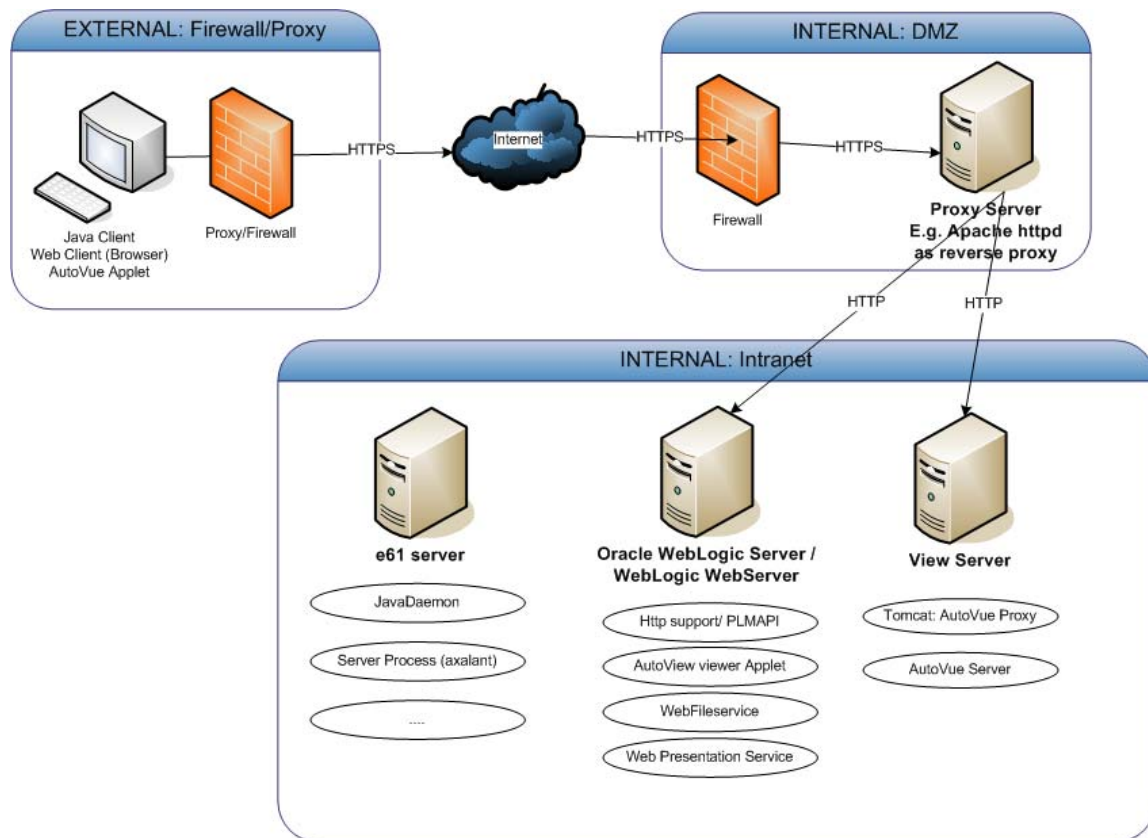
As you can see in the above picture the Java/Web client connects to the Proxy Server over HTTPS.

The proxy server is configured to pass the incoming requests from the internet to the end points on the Oracle WebLogic Server, and if you use AutoVue to the View server in the intranet.

The end points in the intranet are

▫   PLMAPI for Java Client HTTP support

▫   Java Client Web Start for Java Client download

▫   Web Presentation Service for the Web Client

▫   Web Fileservice for file transfer over http

▫   AutoVue proxy for AutoVue Applet

▫   VueLink Servlet for Applet download

For this scenario you have to adapt the Java client to use https protocol and configure the proxy server to pass the incoming requests and accept https connections.

Note   For setting up the Firewalls or other DMZ infrastructure, refer to the documentation provided by respective suppliers.

Setup Apache HTTPD as SSL Reverse Proxy

As mentioned, this document describes only the minimal apache configuration needed, which is not sufficient for securing production environments.

This scenario used Apache HTTP Server (httpd) 2.2.x including OpenSSL (used version is: httpd-2.2.16-win32-x86-openssl-0.9.8o.msi).

Next steps require the installed apache, which need not be fully configured.

In this example we're using a self signed certificate for apache. The certificate identifies the apache server to the client.

This first configuration is not a full SSL reverse proxy, because intranet communication from proxy to internal WebLogic server still goes over http. Full SSL reverse proxy is configured in the "Secure Internal Communication" section in this document.

Create Self Signed Certificate (Open cmd shell):

Creating self signed certificates is only recommended for testing. For production environments certificate from a trusted certificate authority is recommended.

Example:

```
set PATH=%PATH%;C:\Program Files (x86)\Apache Software Foundation\Apache2.2\bin
set OPENSSL_CONF=C:\Program Files (x86)\Apache Software Foundation\Apache2.2\conf\openssl.cnf
cd C:\Program Files (x86)\Apache Software Foundation\Apache2.2\conf
openssl req -new -x509 -nodes -out server.crt -keyout server.key
```

This will create the certificate that will be used automatically by apache httpd.

Changes in <apache_home>\conf\httpd.conf

Activate Modules:

```
LoadModule substitute_module modules/mod_substitute.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule ssl_module modules/mod_ssl.so
```

This will also activate the needed proxy modules.

Add a proxy section:

```
<IfModule proxy_module>
  ProxyRequests Off
  <Proxy *>
    Order deny,allow
    Allow from all
  </Proxy>
  # JNLP
  <Proxy "*.jnlp">
        SetOutputFilter SUBSTITUTE
        Substitute s#http://<wl_server>:7103#https://<proxy_server>#i
  </Proxy>
  # Proxys
  ProxyPass /autovueproxy http://<view_server>:8080/VueServlet/servlet/VueServlet
  ProxyPassReverse /autovueproxy http://<view_server>:8080/VueServlet/servlet/VueServlet
  ProxyPass /autovueapplet http://<wl_server>:7103/VueLink
  ProxyPassReverse /autovueapplet http://<wl_server>:7103/VueLink
  ProxyPass /plmapi http://<wl_server>:7103/plm-api-axis/services
  ProxyPassReverse /plmapi http://<wl_server>:7103/plm-api-axis/services
  ProxyPass /Jacc http://<wl_server>:7103/Jacc
  ProxyPassReverse /Jacc http://<wl_server>:7103/Jacc
  ProxyPass /fms http://<wl_server>:7103/FileService
  ProxyPassReverse /fms http://<wl_server>:7103/FileService
  ProxyPass /AgilePlmWps http://<wl_server>:7103/AgilePlmWps
  ProxyPassReverse /AgilePlmWps http://<wl_server>:7103/AgilePlmWps
</IfModule>
```

Activate SSL:

```
Include conf/extra/httpd-ssl.conf
```

Restart apache httpd.

Windows Server 2008 or Windows 7 special adaptation in conf/extra/httpd-ssl.conf:

The adaptation of the "SSLSessionCache" is only necessary on Windows Server 2008 or Windows 7, if httpd is installed in directory like: C:\Program Files (x86)\Apache Software Foundation\Apache2.2

The directory "C:\app\apache\sslcache\" must exist.

```
SSLSessionCache             "shmcb:C:\app\apache\sslcache\ssl_scache(512000)"
```

Restart apache httpd.

Check your configuration

If the reverse proxy is accessible from internal:

https://<proxy_server>

If you're using a self signed certificate you have to add a security exception to your browser for this

proxy server. The default apache web page which normally shows "It works!" must appear.

Following web links must be accessible without error:

▫ https://<proxy_server>/fms

▫ https://<proxy_server>/plmapi

▫ https://<proxy_server>/Jacc

▫ https://<proxy_server>/AgilePlmWps

Additionally, if you use AutoVue:

▫ https://<proxy_server>/autovueproxy

▫ https://<proxy_server>/autovueapplet/jvue.jar

Setting up the Java Client/Java Virtual Machine

To enable https support in the Java Client you have to activate/change the HTTP/S support setting.

1. Download Java Client Webstart.

2. Open: https://<proxy_server>/Jacc and select the download link.

Note	If you're using self signed certificate a message occurs that the certificate cannot be validated. Accept it if you trust the certificate.

3. After the download of the application, next warning appears – "digital signature of the application cannot be verified". If you trust the publisher, select "Always trust content of this publisher" and run the application.

   The Java Client starts.

4. Open the Java Client Preferences mask.
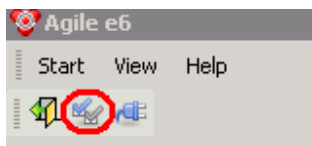
5. Enable the HTTP/S support and change/add the Server URL and Service name to:

   ```
   Server URL: https://<fully qualified proxy_server name>
   Service name: plmapi
   ```

   Following picture shows an example configuration:
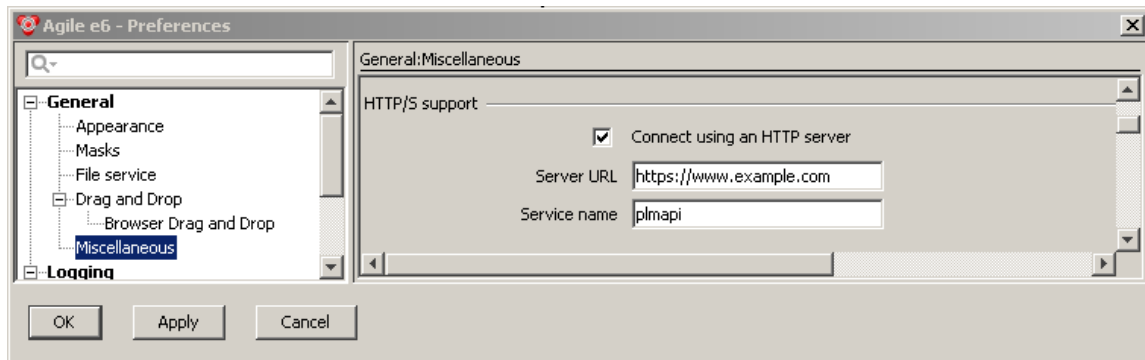
6. If you're using self signed certificate, import the certificate into the Java Virtual Machine keystore.

   Connecting with the java client using plmapi over https requires java to trust the end point of the communication. The end point in our example is the apache httpd reverse proxy server.

7. Perform the following steps:

   1. Transfer the file "server.crt" created on the proxy server in "<apache_home>/conf" to your java client machine.

   2. Determine JAVA_HOME of the Java Virtual Machine where your Java Client Web Start is running (e.g. in windows with process explorer).

   3. Copy the Public Certificate (server.crt, see above) to %JAVA_HOME%\lib\security.

   4. Open an Administrative cmd shell and change to directory %JAVA_HOME%\lib\security.

   5. Execute:
      %JAVA_HOME%\bin\keytool -import -keystore cacerts -storepass changeit -trustcacerts -alias mytrustcert -file server.crt

8. Close Java Client, open "https://<proxy_server>/Jacc" and select the download link again.

Java Client with Forward Proxy

Proxy in this case does not mean the reverse proxy. It refers to the forward proxy for the outgoing connections from the client to the internet.

By default the Java client uses the proxy configuration of your java environment. This is configured on the client side in the Java Control Panel (e.g. open MS Windows control panel > Java) on the General Tab open the "Network Settings…".

Note   Depending on your proxy configuration, one of the proxy settings has to be selected.
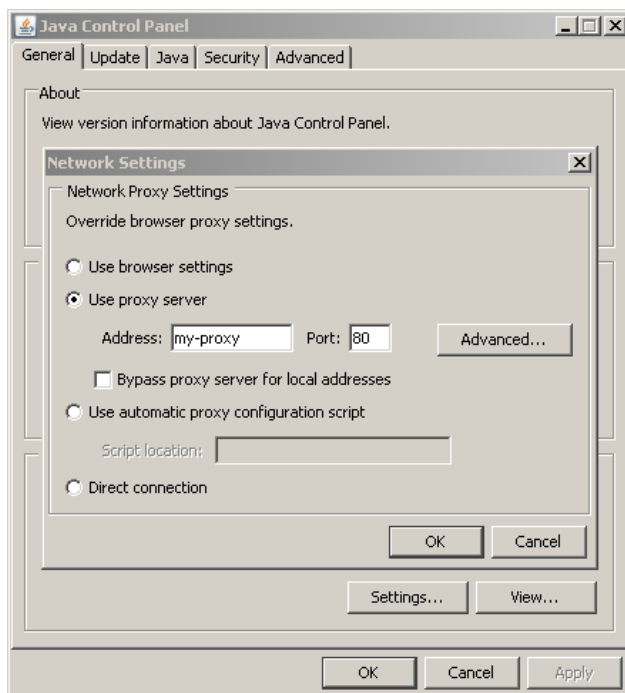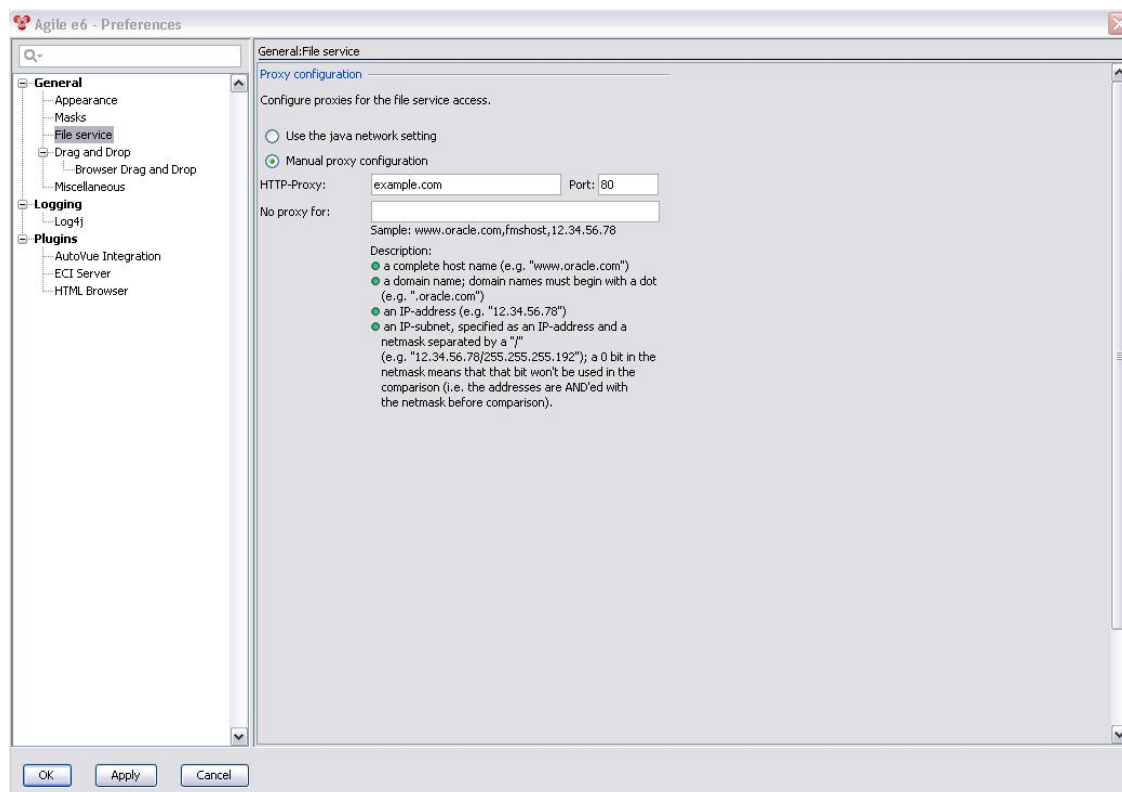
Accessing Web File Service with Forward Proxy

The proxy for the Web File Service can be configured in the Java client preferences. By default the java network settings are used.

1. Start Java Client Web Start

2. Open the Java Client Preferences mask.
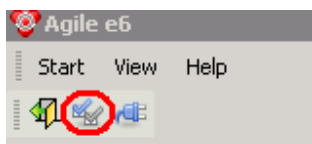
3. Adapt the File service proxy configuration if necessary.

Setting up the Web Fileservice

To enable https support for the Web Fileservice the Web address in the vault configuration has to be changed.

Limitation: As shown in the following screenshot, you can only define one "Web Address" per Vault. This means the "Web Address" is same for internal and external Java/Web Client. If you want to use this vault for internal and external java/web clients the <proxy_server> must be reached under his name from external and also from internal.

Start the Java client with a manager user and select Manager > File Management > Vaults.

```
Protocol: https
Host: <proxy server>
Port: 443
Path: /fms
```

Following picture shows the configuration:

Setting up AutoVue

General setup of AutoVue is described in the "*Installation and Administration Manual for AutoVue Integration on Windows for Agile e6.1.2*".

Perform these steps only if you are using AutoVue.

1. Setup the AutoVue tunnelling servlet that it can be reached over https. For example, tomcat is configured with https.

2. Start a Java Client with a manager user and select "System > AutoVue > Configuration".

3. To enable secure communication the following values must be changed:

   - EDB-PVM-AV-PROXY:

     Value (for example): https://<proxy_server>/autovueproxy

     Description: The URL where you can reach the AutoVue tunneling servlet over https.

   - EDB-PVM-AV-USE-PROXY:

     Value: true

     Description: To use HTTPS communication, set this to true.

   - EDB-PVM-AV-DMS:

     Value (for example): http://<weblogic_server>:<weblogic port>/VueLink/Vuelink

     Description: The Oracle Agile DMS Servlet address, where the AutoVue Server can reach the DMS Servlet.

     This must NOT be configured with https. Use http.

   - EDB-PVM-APPLET-BASE-URL

     Value (for example): https://<proxy_server>/autovueapplet

     Description: The base https URL for the AutoVue viewer applet download.

Setting up the Web Client

No configuration changes have to be done if you want to use the Web client with https.

Just use the HTTPS protocol and port in your browser.

The Web Fileservice adaptations in the dump also have to be done (see above) to use file checkin/out in the Web Client with https. Proxy configuration will be used from the browser.

Changing the Lightweight Report URL

The "Report_Service_URL" attribute in all application configuration files in <ep_root>/init/<application>.xml must be changed.

Search the line:

<PLMPresentationServices Report_Service_URL="…"/>

and adapt "Report_Service_URL" to:

https://<proxy_server>/AgilePlmWps/reporter/report

If you change this line it is valid for the complete application. An internal/external Client will also use this URL.

Secure Internal Communication

Secure internal communication is only possible by also enabling PLMAPI for the java client like for the external secure communication; otherwise the communication from the java client to the axalant server process is not secure. DTV Client uses also unsecure communication to the axalant server process. Secure connection from the DTV Client to the axalant server process is not possible.

Using Web Client with https and Web Fileservice with https protocol is also secure.

To use secure internal communication together with secure external communication, the proxy configuration has to be modified to a full ssl reverse proxy.

The communication between the servers in the "secured" LAN is unencrypted. You need to ensure that the network traffic in this area is not accessible to the unauthorized persons.

Following picture illustrates an example communication.
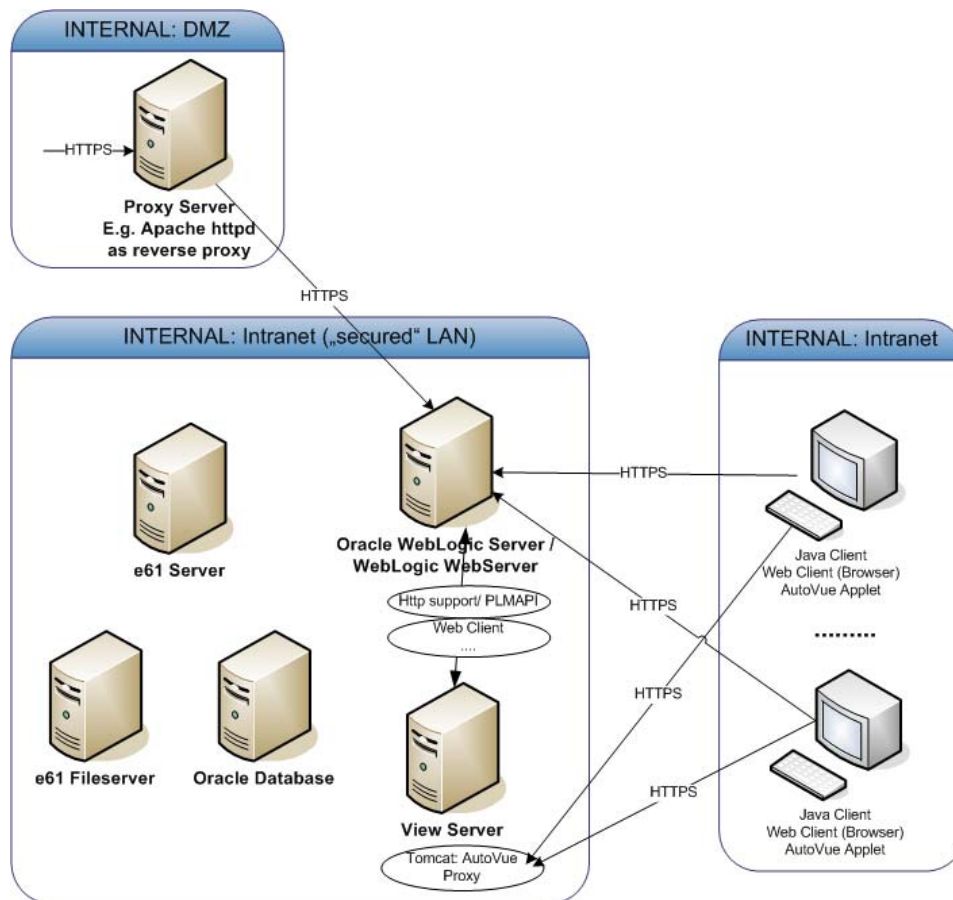
Modifying Apache HTTTPD to Full SLL Reverse Proxy

Following adaptations must be made to the current proxy section in the httpd.conf file:

```
<IfModule proxy_module>
  ProxyRequests Off
  SSLProxyEngine On
  <Proxy *>
    Order deny,allow
    Allow from all
  </Proxy>
  # JNLP
  <Proxy "*.jnlp">
      SetOutputFilter SUBSTITUTE
      Substitute s#https://<wl_server>:7104#https://<proxy_server>#i
  </Proxy>
  # Proxys
  ProxyPass /autovueproxy https://<view_server>:8443/VueServlet/servlet/VueServlet
  ProxyPassReverse /autovueproxy https://<view_server>:8443/VueServlet/servlet/VueServlet
  ProxyPass /autovueapplet https://<wl_server>:7104/VueLink
  ProxyPassReverse /autovueapplet https://<wl_server>:7104/VueLink
  ProxyPass /plmapi https://<wl_server>:7104/plm-api-axis/services
  ProxyPassReverse /plmapi https://<wl_server>:7104/plm-api-axis/services
  ProxyPass /Jacc https://<wl_server>:7104/Jacc
  ProxyPassReverse /Jacc https://<wl_server>:7104/Jacc
  ProxyPass /fms https://<wl_server>:7104/FileService
  ProxyPassReverse /fms https://<wl_server>:7104/FileService
  ProxyPass /AgilePlmWps https://<wl_server>:7104/AgilePlmWps
  ProxyPassReverse /AgilePlmWps https://<wl_server>:7104/AgilePlmWps
</IfModule>
```

This proxy is still used only for external communication. Internal communications use the direct connect to WebLogic https port and tomcat https port.

Restart apache httpd.

Setting up HTTPS on the Oracle WebLogic Servers

The demonstration digital certificates, private keys, and trusted CA certificates used in this description should NOT be used in a production environment.

They are provided by default during WebLogic installation/domain setup.

For a production environment follow the steps in the WebLogic documentation to use non Demo digital certificates, private keys, and trusted CA certificates instead.

To activate SSL:

1. Start the Administration Console (e.g.: http://<>:7101/console)

2. If you have not already done so, in the Change Centre of the Administration Console, click Lock & Edit.

3. In the left pane of the Console, expand "Environment" and select "Servers".

4. Click the name of the server for which you want to activate SSL Port.

5. At tab "Configuration" > "General", configure "SSL Listen Port Enabled" and "SSL Listen Port".
   If you want to disable the non-SSL Port, uncheck the "Listen Port Enabled" checkbox.
   If you disable non-SSL Port of the "AdminServer", e6 batch deployment will not work.

6. Select a SSL port which is not in use. Using a port number 1 above the "Listen Port" should be

fine in most cases.

7. Save and release your configuration.

8. Activate SSL ports for all servers (e.g: AdminServer, eSeries-01) in all Domains (e.g. eSeriesDomain, eSeriesDomain_plmref, …).

Setting up the Java Client/Java Virtual Machine

For Internal usage of HTTP/S support, a different setup as for external usage is required, because from internal java client connection to the WebLogic server uses normally no proxy.

To enable https support in the Java Client you have to activate/change the HTTP/S support setting.

1. Download Java Client Web Start.

   Open: https://<weblogic_server>:7104/Jacc and select the download link.

   If you're using WebLogic demo certificate, a message appears - "the certificate cannot be validated". Accept it if you trust the certificate. This adds exception to your browser. Additionally the Java Web Start again warns that the certificate cannot be validated. Accept if you trust the certificate, and select "Always trust content of this publisher".

   After you download the application, next warning appears - "the digital signature of the application cannot be verified". If you trust the publisher, select "Always trust content of this publisher" and run the application.

   The Java Client starts.

2. Open the Java Client Preferences mask.
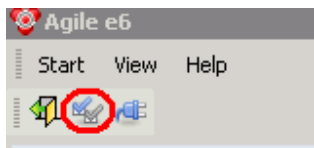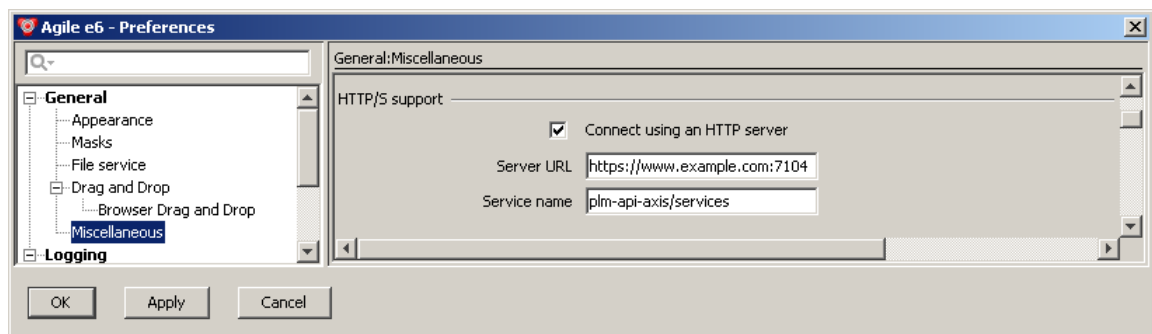
3. Enable the HTTP/S support and change/add the Server URL and Service name to:

   ```
   Server URL: https://<fully qualified weblogic_server
   name>:<https_port>
   Service name: plm-api-axis/services
   ```

   Following picture shows an example configuration:

   If you're using WebLogic demo certificate, import the certificate into Java Virtual Machine keystore.
   Connecting with the java client using plmapi over https requires java to trust the end point of

the communication. The end point in this example is the WebLogic server.

4. Perform the following steps:

   1. Convert the WebLogic demo certificate to "pem" format:

      1. Login to your WebLogic server.

      2. Open a cmd shell.

      3. Execute:
```
cd /D <wlhome>/server/bin
setWLSenv.cmd
cd <wlhome>/server/lib
java utils.der2pem CertGenCA.der
```

   2. Transfer the file "<wlhome>/server/lib/CertGenCA.pem" to your java client machine.

   3. Determine JAVA_HOME of the Java Virtual Machine where your Java Client Web Start is running (e.g. in windows with process explorer).

   4. Copy the Public Certificate (CertGenCA.pem, see above) to %JAVA_HOME%\lib\security.

   5. Open an Administrative cmd shell and change to directory %JAVA_HOME%\lib\security.

   6. Execute:

      %JAVA_HOME%\bin\keytool -import -keystore cacerts -storepass changeit -trustcacerts -alias mytrustcert -file CertGenCA.pem

5. Close the Java Client, open "https://<weblogic_server>:7104/Jacc" and select the download link again.

Setting up the Web Fileservice

For Internal usage of HTTP/S supports a different setup as for external usage is required, because from internal java client connection to the WebLogic server normally does not use a proxy.

To enable https support for the Web Fileservice the Web address in the vault configuration has to be changed.

Limitation see also external configuration.

Start the Java client with a manager user and select Manager > File Management > Vaults.
```
Protocol: https
Host: <weblogic_server>
Port: 7104
Path: /FileService
```

Setting up AutoVue

For Internal usage of HTTP/S supports a different setup as for external usage is required, because from internal the connection to the AutoView Proxy uses normally no proxy from dmz.

Start a Java Client with a manager user and select "System > AutoVue > Configuration".

To enable secure internal communication the following values are different to the secure external communication:

   ▫ EDB-PVM-AV-PROXY:

Value e.g.: `https://<view_server>:8443/ VueServlet/servlet/VueServlet`

Description: The URL where you can reach the AutoVue tunneling servlet over https.

▫ EDB-PVM-APPLET-BASE-URL

Value e.g.: https://<weblogic_server>:7104/VueLink

Description: The base https URL for the AutoVue viewer applet download.

Changing the Lightweight Report URL

The "Report_Service_URL" attribute in all application configuration files in
<ep_root>/init/<application>.xml must be changed.

Search the line:

<PLMPresentationServices Report_Service_URL="…"/>

and adapt "Report_Service_URL" to:

 https://<weblogic_server>:7104/AgilePlmWps/reporter/report

If you change this line it is valid for the complete application. An internal/external Client will also use this URL.

<div align="right">

Chapter 8

</div>

Component Based Installation

The e6 server J2EE components installation requires WebLogic server on the machine where the installation is performed. If you do not want to have the e6 native server processes running on the same machine where WebLogic is also running, you can install the e6 server on different machines with different installation components.

The installation where the J2EE components are installed will be called the "j2ee installation", and the installation where the native components like java,dtv daemon are installed will be called "native installation".

In general, follow the installation documentation for your operating system. "Installation Manual for Agile e6.1.2 Server on Windows" for windows installations or "Installation Manual for Agile e6.1.2 Server on UNIX" for unix installations.

This section describes the adaptations for a component based installation. Only the differences are described.

Requirements

Depending on your operating system, refer the Installation documentation - "Installing Agile e6 on Windows Server" for windows installations or "Installing Agile e6 on Unix Server"" for unix installation.

Installing e6 Native Components

Following components will be installed:

□   Administration Client

□   Java Daemon

□   FMS Java Daemon

□   DTV Daemon

□   Server libraries

□   Fileserver

□   Applications (plmref, plmtest)

1.   Login to the machine where you want to install "e6 Native Components".

2.   Follow the instructions in the installation documentation of your operating system.

Note   WebLogic server installation is not required.

Note   Oracle 32 bit database client installation is required.

As described in the section "Agile e6 Server" in Chapter "Installation with the installer" the

"Installer Component" mask opens. In this mask deselect "Server (J2EE Components) ".
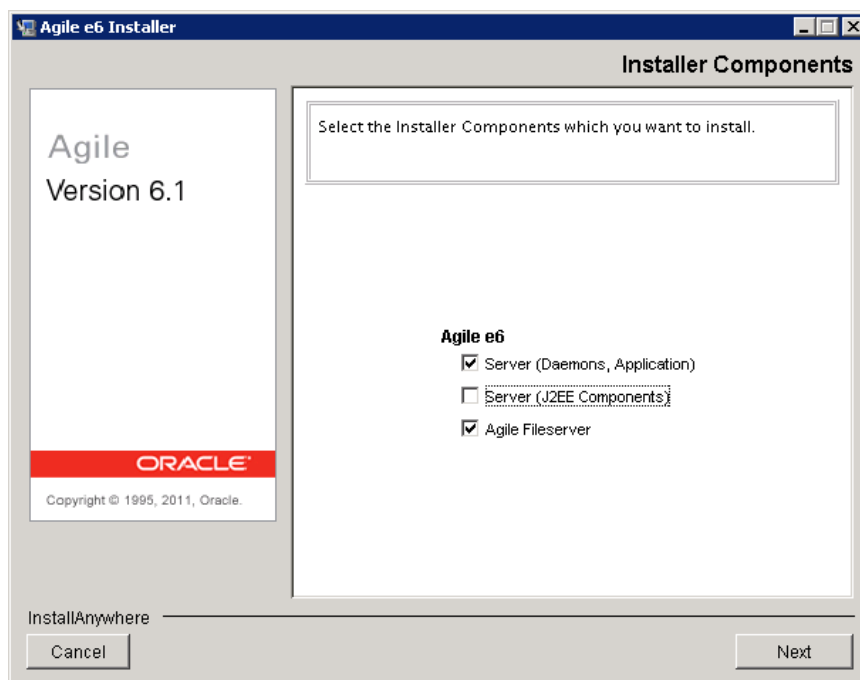
This will prevent the setup of the "e6 J2EE Components".

If you do not want the Agile Fileserver to be installed, deselect it.

3. Adapt the values in the Component Configuration mask.
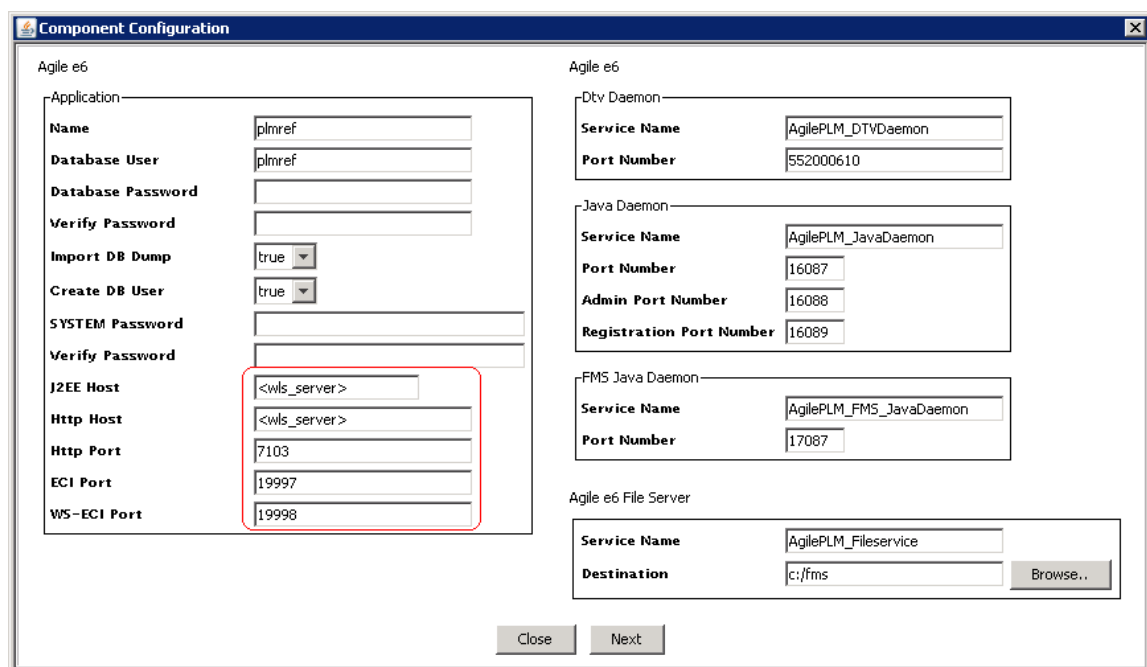
Following values must be identical for native and j2ee installation.

Setting	Description
Name	The name of the default application you want to create
Database User	The database user you want to use for the application.
Database Password	The password of the database user.

The J2EE Components are not installed at this point. You have to use the values marked in red later to setup the e6 J2EE components.

Following values must match the values of your J2EE installation. These are references to these values.

Setting	Description
J2EE Host	The host where WebLogic is running (the WebLogic server host name).
Http Host	The host where the Web client can be reached over http (the WebLogic server hostname).
Http Port	The port where the Web client can be reached over http (the HTTP port of WebLogic server).
ECI Port	The port where the business service will be configured to "wait" for requests.
WS-ECI Port	The port where the Webservices will be configured to "wait" for requests.

4. Follow the instructions in the installation documentation of your operating system

Note WebLogic domains will not be set up.

Note Batch deployment tasks must be executed on the j2ee installation only

Installing e6 J2EE Components

Following components will be installed:

▫ Administration Client

▫ Staging directory (See "Customer adaptations deployment" section in this document)

▫ Applications (plmref, plmtest..)

▫ WebLogic Domains containing deployments

To install e6 J2EE components:

1. Log in to the machine where you want to install "e6 J2EE Components".

   Note If WebLogic is already installed use the WebLogic installation user to install the e6 J2EE Components.

2. Follow the instructions in the installation documentation of your operating system.

   Note Oracle database client installation is not needed.

   As described in the section "Agile e6 Server" in Chapter "Installation with the installer" the

"Installer Component" mask openeds. In this mask deselect "Server (Daemons, Application) ".
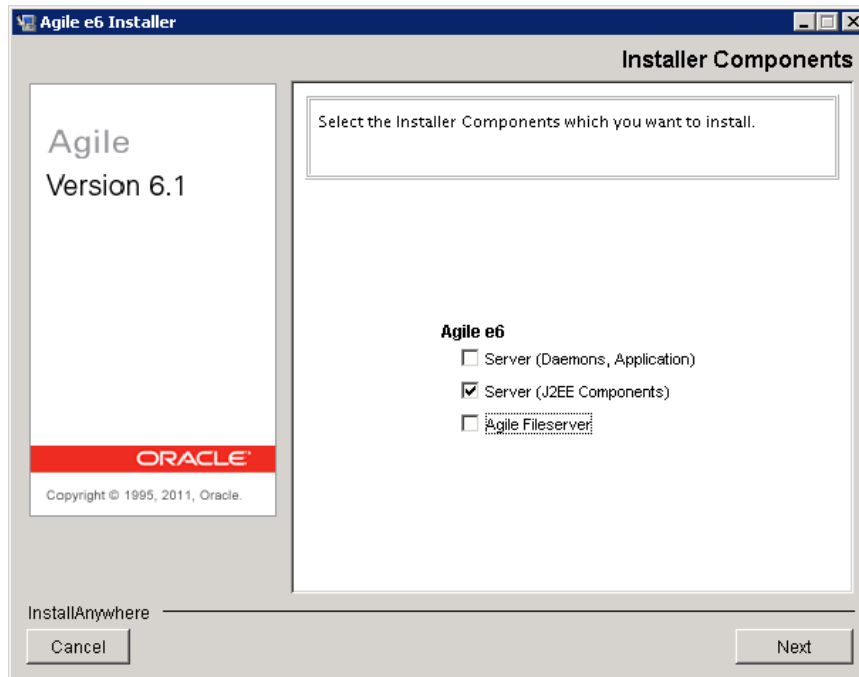
This will prevent the setup of the "e6 Native Components".

Following values must be identical for native and j2ee installation.

Setting	Description
Name	The name of the default application you want to create
Database User	The database user you want to use for the application.

Database Password	The password of the database user.

3. Apply the values of your native installation as given in fields marked in red.

    Following values must match the values of your native installation. These are references to these values.

Setting	Description
Java Daemon Host Name	The hostname where the Agile e6 Java Daemon is/will be running
Java Daemon Port	The Java daemon port
FmsJava Daemon Host Name	The hostname where the Agile e6 FMS Java Daemon is/will be running
FmsJava Daemon Port	The FMS Java daemon port

4. Follow the instructions in the installation documentation of your operating system.

Creating the Application

In a component based installation, every application has to be created on the native installation and on the j2ee installation. The Administration Client, however, is available for both installations. The Administration client checks whether he should manage a j2ee or a native installation.

Applications can be created as usual. Following two screenshots show the differences to the default web forms for the application creation with the Administration Client.

J2EE Application Creation:

NATIVE Application Creation:

User Authentication

LDAP Support

LDAP (Lightweight Directory Access Protocol) is an application protocol for querying and modifying directory services running over TCP/IP.

With Agile e6.1.2, LDAP-based authentication is supported. While a PLM user is logging on to the PLM system (through any supported client), the password of that PLM user is checked against an LDAP repository instead against the password which is normally stored in the PLM database.

The communication between the PLM server and the LDAP repository has to be set up. Every PLM user, as configured in the PLM database, has to exist in the LDAP repository in order to be authenticated upon login.

Note	Although LDAP support helps in use of a single password for different systems, this should not be mistaken with automatic Single-Sign-On (SSO) support, which would allow a user to log on automatically without even being asked to provide user login and password.

Prerequisites

▫ LDAP server (Oracle Internet Directory / MS Active Directory (does not work with encryption) / other LDAP server).

▫ Oracle LDAP client (part of the Oracle client installation).

▫ A site must have been defined in the DDM Site Vaults (see the module "General Replication" of the online help).

▫ A PLM user name and a LDAP user name must have been created.

Note	Please note that the PLM user name and the LDAP user name don't need to be identical. But it is required that the LDAP user name is configured in the PLM system.

User Authentication via LDAP
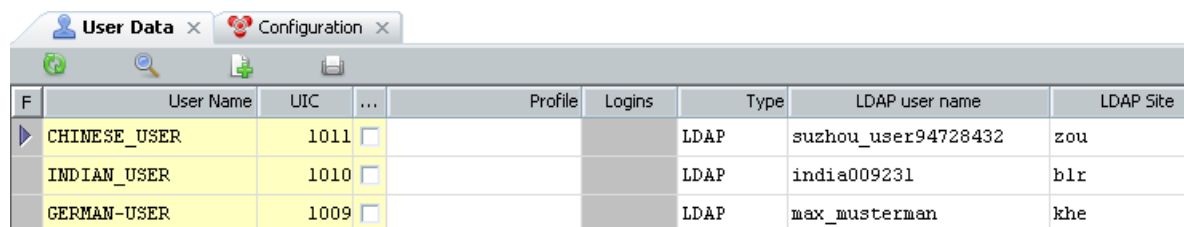
A LDAP directory is often used to manage users and organization units in a central environment. Products like the Oracle Internet Directory are able to manage users, groups and organization units in a standard LDAP environment and are compatible with the most other LDAP servers which are based on the LDAP standards.

With Agile e6.1.2 the LDAP authentication mechanism supports the authentication of a PLM user password against a LDAP repository.

The LDAP for Agile e6.1.2 uses the Base-DN for a direct access path to authenticate the user. LDAP does not support relative search paths.

Setup an LDAP User

To change the authentication mechanism of a user, select the "LDAP" entry as "Type" in the user list of Agile PLM e6.1.2.

F	User Name	UIC	...	Profile	Logins	Type	LDAP user name	LDAP Site
▶	CHINESE_USER	1011 ☐				LDAP	suzhou_user94728432	zou
	INDIAN_USER	1010 ☐				LDAP	india009231	blr
	GERMAN-USER	1009 ☐				LDAP	max_musterman	khe

Typically, an Agile e6.1 user has a different user name in the LDAP repository, therefore a LDAP user name field is supported to map the user names. To support LDAP multi-domains, the administrator can link each user to the site specific LDAP configuration.

The LDAP system takes care of the password policies (expiration and format).

Note The enhanced security module and the possibility to change the password within PLM are deactivated for LDAP users.

Configuration

The LDAP configuration used by the PLM system is stored in the database as configuration parameters (T_CFG_DAT).

Name	Default Value	Description
EDB-LDAP-HOST	<LDAP-host>	LDAP host name
EDB-LDAP-PORT	-	LDAP service port (=default port depends on encryption mode)
EDB-LDAP-BASEDN	cn=users, dc=agile, dc=com	Base DN of the user group
EDB-LDAP-ENCRYPTION	Yes	LDAP encryption mode (yes=SSL, no=clear text)

The configuration entries are site specific to support multi-domains. For each site different LDAP settings can be configured.

Example:

	ID	Type	Typ.!	Value	
	EDB-LDAP-BASEDN	SITE	blr	cn=users,dc=india,dc=sample,dc=com	Base DN of the user group group
	EDB-LDAP-ENCRYPTION	SITE	blr	yes	LDAP encryption mode (yes (yes=SSL, no=clear text)
	EDB-LDAP-HOST	SITE	blr	ldap.india.sample.com	LDAP host name
	EDB-LDAP-PORT	SITE	blr	-	LDAP service port (-=Def(=Default port depend on encryption mode)
	EDB-LDAP-BASEDN	SITE	khe	cn=users,dc=emea,dc=sample,dc=com	Base DN of the user group group
	EDB-LDAP-ENCRYPTION	SITE	khe	yes	LDAP encryption mode (yes (yes=SSL, no=clear text)
	EDB-LDAP-HOST	SITE	khe	ldap.germany.sample.com	LDAP host name
	EDB-LDAP-PORT	SITE	khe	-	LDAP service port (-=Def(=Default port depend on encryption mode)
	EDB-LDAP-BASEDN	SITE	zou	cn=users,dc=epac,dc=sample,dc=com	Base DN of the user group group
	EDB-LDAP-ENCRYPTION	SITE	zou	yes	LDAP encryption mode (ye: (yes=SSL, no=clear text)
▶	EDB-LDAP-HOST	SITE	zou	ldap.us.sample.com	LDAP host name
	EDB-LDAP-PORT	SITE	zou	-	LDAP service port (-=Def(=Default port depend on encryption mode)

Note   Please note that a language specific configuration – though selectable from the Type column – is not relevant for the LDAP support.

Office Suite - PDF Generator Installation

Note	Further information about how to install he Office Suite can be found in the Installation Manual for Agile e6.1.2 Client on Windows.

One main feature of the Office Suite is the PDF generation from a MS Office file. The Office Suite supports a PDF printer to convert Office documents into PDF files.

The Office Suite PDF Service is a MS Windows Service and does not need any GUI.

Installation

1. Extract the installation package to your installation directory.

    The installation package of the Office Suite PDF Service is located on the package directory and named officesuitepdf.zip.

    In this example the installation path is "C:\Program Files\Agile_e6\Office-Suite-PDF" (the Office-Suite-PDF path is the base path in the package).

    ```
    unzip z:\package\officesuitepdf.zip -d "c:\Program Files\Agile_e6"
    ```

    The installation package contains the following directories:

    ```
    -> PDF Service Root        (C:\Program Files\Agile_e6\Office-Suite-
    PDF)
     -> axalant
        -> cmd                 (scripts directory)
        -> pdf                 (PDF service files)
        -> bin
          -> intel-ms-nt5.0    (binaries like the FMS-Client)
          -> java              (e6.1 Java archives)
      -> ext
        -> bin
          -> intel-ms-nt5.0    (external binaries)
          -> java              (external Java archives)
      -> tmp                   (Logging directory)
    ```

2. Adapt the installation.

    You need to adapt the start-up script to setup the Java-Runtime and the installation path of the Office-Suite PDF Service. The script is located at the …\axalant\cmd sub directory of the installation.

    The pdf.bat script contains the following basic configuration settings:

    ```
    set JAVA_HOME=<JAVA_HOME>
    set ep_root=<ROOT DIRECTORY OF THE OFFICE SUITE PDF SERVICE>
    ```

    Example:

```
set JAVA_HOME=C:\Program Files\Java\jdk1.5.0_16
set ep_root=C:\Program Files\Agile_e6\Office-Suite-PDF
```

3. Adapt the service settings.

   The OfsPdf.properties file is located at the axalant/pdf sub directory of the installation. This file sets the environment variables needed by the Office Suite PDF Service.

   The following properties have to be adapted.

   Note   The other properties should not be changed.

```
#
# ECI connection
#
host=<HOSTNAME OF THE ECI DAEMON>
port=<PORT OF THE ECI DAEMON>
env=<PLM APPLICATION ENVIRONMENT>
#
# Directories
#
varenv.DATAVIEW_CROO=<BINARIY DIRECTORY OF THE PDF SERVICE>
varenv.ep_root=<ROOT DIRECTORY OF THE PDF SERVICE>
varenv.axalant_root=<AXALANT DIRECTORY OF THE PDF SERVICE>
varenv.$TMP=<PDF WORK DIRECTORY OF THE PDF SERVICE>
#
# Host names
#
varenv.CLI_HST=<NAME OF THE PDF SERVICE MACHINE>
varenv.CLI_SRV=<NAME OF THE PLM SERVER MACHINE>
#
# PLM Client
#
client1=<PLM USER>,<PASSWORD>,office.pdf.OfsPdf
```

Example:

```
#
# ECI connection
#
host=khe-plm
port=20001
env=plm_ref
#
# Directories
#
varenv.DATAVIEW_CROO=C:/Program Files/Agile_e6/Office-Suite-
PDF/axalant/bin/intel-ms-nt5.0
varenv.ep_root=C:/Program Files/Agile_e6/Office-Suite-PDF
varenv.axalant_root=C:/Program Files/Agile_e6/Office-Suite-PDF/axalant
```

```
varenv.$TMP=C:/officesuite/PDF_generation
#
# Host names
#
varenv.CLI_HST=pdfsrv
varenv.CLI_SRV=khe-plm
#
# PLM Client
#
client1=DEMOEP_M,not4test,office.pdf.OfsPdf
```

Note   The PDF work directory has to be created and entered in the PLM server configuration, PDF Service, and the PDF printer.

Office Suite PDF Service

The Office Suite PDF Service uses the same mechanism as the Java-Daemon to install, remove, start, and stop the service.

It can be either installed as a Windows Service or run as a console application.

Installing Office Suite PDF Service as a Windows Service

Note   The PDF Generator can be installed with the Windows Installer tool. After the PDF Generator is installed, an extra service is added to the Services list. Agile PLM Office Suite PDF (Control Panel > Administrative Tools > Services).
The Agile PLM Office Suite PDF Service is a MS Windows Service and does not need any GUI.

The configuration of the Windows Service registration can be found in the pdf_wrapper.conf file which is located in the …\axalant\pdf sub directory of the installation.

```
#********************************************************************
# Wrapper NT Service Properties
#********************************************************************
# WARNING - Do not modify any of these properties when an application
# using this configuration file has been installed as a service.
# Please uninstall the service before modifying this section.  The
# service can then be reinstalled.

# Name of the service
wrapper.ntservice.name=Agile PLM Office Suite PDF Service

# Display name of the service
wrapper.ntservice.displayname=Agile PLM Office Suite PDF Service

# Description of the service
wrapper.ntservice.description=PDF Generator for Agile PLM
```

```
# Service dependencies.  Add dependencies as needed starting from 1
wrapper.ntservice.dependency.1=

# Mode in which the service is installed.  AUTO_START or DEMAND_START
wrapper.ntservice.starttype=AUTO_START

# Allow the service to interact with the desktop.
wrapper.ntservice.interactive=false

wrapper.ntservice.account=.\axalantrt
wrapper.ntservice.password=********
```

To install the Office Suite PDF Service as Windows Service use the pdf.bat command script located in the …\axalant\cmd sub directory of the installation.

```
pdf.bat –i
```

Remove the Service

To uninstall the Office Suite PDF Service as Windows Service use the pdf.bat command script located in the …\axalant\cmd sub directory of the installation.

```
pdf.bat –r
```

Run as Console Application

To run the Office Suite PDF Service as console application use the pdf.bat command script located in the …\axalant\cmd sub directory of the installation.

```
pdf.bat –c
```

Configuration

The Office Suite needs some information to access the PDF printer and to exchange the files.

Printer Setup

The PDF printer has to fulfill the following prerequisites:

- Generate PDF file without user interaction (pure batch printing)

- Configurable PDF output folder

- Possibility to create a <PDF-Filename>.log file in the PDF output folder, after PDF file creation.

Note   The PDF Output Folder is the exchange folder between the PDF printer and Agile e6.1.2.

Note   Any folder can be used, but the configuration of the PDF printer has to match the configuration of the Agile PLM Office Suite PDF Service.

Post PDF Creation Script

Prerequisite for PDF printer is the <PDF-filename>.log file creation (see above) in the PDF output folder. This file is the "Ready" file for the batch client to trigger the PDF file checkin.

If your PDF printer does not create a <PDF-filename>.log file per default, there exists often the possibility to execute an application after printing. This could be used to create the <PDF-filename>.log file. It must be possible to set the filename "<PDF-filename>.log" as a parameter. The following steps describe the general setup of such a mechanism.

1. Create a file "createlog.bat" in the PDF output folder which looks like this:
   ```
   set FILENAME=%~n1
   set FILEPATH=%~dp0%
   echo Done > %FILEPATH%/%FILENAME%.log
   ```
2. Configure your PDF printer to execute the Windows cmd shell (e.g.: C:\WINDOWS\system32\cmd.exe) with the following parameters (e.g %f is the filename parameter) :
   ```
   /c C:\officesuite\pdf\createlog.bat %f
   ```

Setting up the PDF Printer

To setup the PDF printer

1. Select the PDF printer from the printer list (Manager > Office Suite > PDF Printer),

2. When you select the PDF printer, the following Office Suite configuration parameters are added to the Office Suite configuration table (System > OfficeSuite > Configuration table):

   □ GDM_PDF_BATCH

   □ GDM_PDF_DIRECTORY

   The value of this configuration parameter has to be the same as the value for TEMP in the OfsPdf.properties file. Add "*.pdf" at the end of it.

   □ GDM_PRINTER

   □ GDM_PRINTER_DRIVER

   □ GDM_PRINT_PORT

   Note GDM_PRINTER, GDM_PRINT_DRIVER, GDM_PRINT_PORT are set automatically.

3. The value for the Office Suite configuration parameter GDM_ARC_NOD has to be the same as the value for varenv.CLI_HST in the OfsPdf.properties file.

4. In the OfsPdf.properties file the following has to be set as well:

   □ host (e.g. plmhost)

   □ port (e.g. 20001)

   □ env (e.g. plmref)

5. In the OfsPdf.properties file the correct user has to be entered under "client1".

6. Link the PDF Generator to a lifecycle:

   The PDF Generator has to be linked to the respective transition states for the STD-DOC

lifecycle. The Office Suite provides a LogiView procedure (GdmBatch/Archive) to add a job into the PDF job list. This LogiView procedure is a transition procedure for the usage within the lifecycle of the Office document.

The following transition settings can be used:

Field	With change management	Without change management
From	220	120
State (From)	Approved	In Approval
To	230	230
State (To)	Release	Released
System	X	
For Change Management	X	
Pre-Action	xedbusr_tor_rr_vr	xedbusr_chk_no_obj_ews & xedbusr_tor_rr_vr & sig_cre_sgn
Post-Action	Xedbusr_tor_sa_uv & xchg_prd_unt_poa & GdmBatch/Archiv	Xedbusr_tor_sa_uv & xedbusr_rst_prd(260) & GdmBatch/Archiv

Thus, when an Office Suite document reaches this state, a PDF is created automatically.

7. Add "GdmBatch/Archiv" to the respective states in the Lifecycle mask.

> **Note**   After updating the lifecycle states, the client needs to be restarted.

8. For every file type, for which a PDF should be created, the 'Create Archive Format' checkbox has to be selected in the File type mask (System > Office Suite > File Types).

> **Note**   The PDF generation can only be used for file types of the storage type Single.
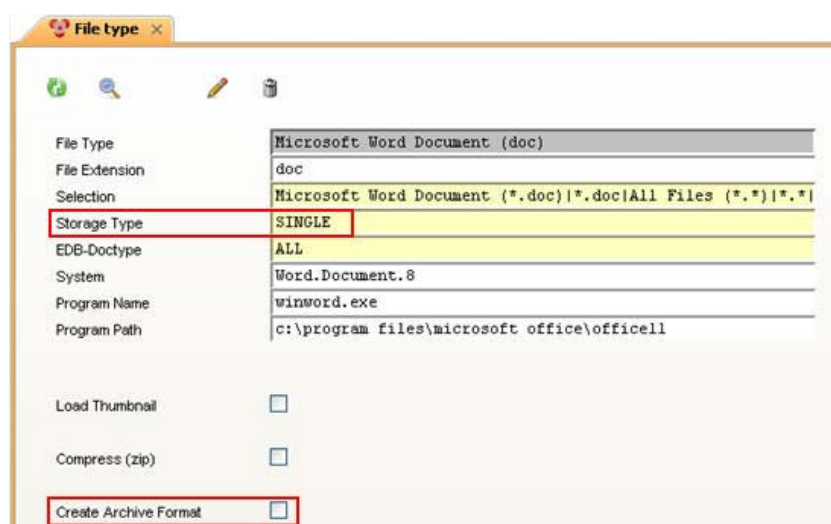
Configure the Agile PLM Office Suite PDF Service

The Agile PLM Office Suite PDF Service does require some information about the local environment, e.g. the exchange file between the PDF printer and Agile e6.1.2.

The configuration file of the Agile PLM Office Suite PDF Service has corresponding settings within the Agile e6.1.2 Office Suite configuration (System > Office Suite > Configuration).

The configuration table describes the settings of the configuration file of the Agile PLM Office PDF Service and the configuration settings of the PLM Office Suite configuration.

Office Suite configuration for the PDF service:

Variable	Value	Description
GDM_ARC_NOD	Hostname of the PDF Service machine (e.g. pdfsrv)	Host on which the PDF files are created (PDF batch client)
GDM_PDF_BATCH	RUN	Control parameter of the job 'PDF Generation'
GDM_PDF_DIRECTORY	PDF exchange directory (e.g. C:\officsuite\PDF_generation*.pdf)	Working file of the PDF generator (including absolute path and generic file name)

Runtime

The Agile PLM Office Suite PDF Service can be started via the Windows service manager. The service connects to a PLM server and starts the LogiView procedure GdmBatch/Spooler to process the PDF job list.

Note   The service should not be stopped via the Windows Service Manager, because the PLM server may not shut down.

To stop the Agile PLM Office Suite PDF Service, select shutdown in the no-select menu of the PDF job list. This operation sets the Office Suite configuration parameter GDM_PDF_BATCH to STOP. The LogiView procedure GdmBatch/Spooler checks that parameter and shuts down. During the shutdown of the spooler the Office Suite configuration parameter GDM_PDF_BATCH is set to RUN again.

The userexit sets the Office Suite configuration parameter:

Name	Runtime Setting	Shutdown Setting
GDM_PDF_BATH	RUN	STOP

<div align="right">

Chapter 11

</div>

Setting up the Online Help

There are two types of online documentation available:

▫   Online Help for Java Client and Web Client has to be setup separately.

> **Note**   This Online Help does not provide a full text search.

▫   Online Help for all Agile e6.1.2 features is available in the Native Windows Client. Because of restrictions caused by the Microsoft HTML Help format, this help is available only on Windows computers running Internet Explorer 6.0 or later.

> **Note**   The Online Help can be viewed only with Internet Explorer.

Java and Web Client

To make the Online Help for masks, fields, menus, userexits, etc. available in the Java and Web client, complete the following steps:

1.   Change to the following directory: <installation_media>/packages/.

2.   Extract the file clienthelphtml.zip in a temporary directory. This will create the directory structure ".\axalant\htd\htdocs\axalant\doc_ep\eng".

3.   Extract the file hlp_e6.zip in this directory.

> **Note**   At the moment the zip file is only available in English. Copy the file hlp_e6.zip from the "eng" directory to the "ger" directory and extract it there if the dump language is set to German.

4.   Copy the "axalant" folder under the htdocs to a web server of your choice (e.g. to the http server of your Oracle Application Server installation). Copy the axalant folder directly below your document root of your web server.

5.   Log in as a manager user (edbcusto, demoep_m).

6.   Select System > Other Parameters.

7.   In the configuration mask, search for Rubric EDB-HLP and select the folder Configuration parameter.

8.   Set the documentation root EDB-HLP-ROOT to: http://<http server name>:<http port>/axalant/

   The program adds /doc_ep/eng/ or /doc_ep/ger/, depending on which language is active, followed by the index file name index.html.

   •   English: http://<http server name>:<admin http port>/axalant/doc_ep/eng/index.html

   •   German: http://<http server name>:<admin http port>/axalant/doc_ep/ger/index.html

On UNIX Systems, Agile e6.1.2 starts Mozilla as a default. If the browser is not defined, follow the steps below:

1.   Select Manager > External Applications to specify different browsers on different operating

systems and client nodes.

This is where the default browsers for each available UNIX System are defined. (For Microsoft Windows the default system HTML browser is used.)

2. Copy the line for your hardware architecture and insert the hostname and browser name.

Note Make sure that all of the following are true:

- File type is equal to html
- Mode is equal to H (help).
- "-" is a wildcard and means every possible value (like *).

Native Windows Client

The Online Help for the Windows Client is installed together with the client software. No special setup is required.

If you want to use the HTML-based help (Java and Web Client) also in the Windows Client, you need to do the following:

1. Open the Windows client.

2. On the desktop, right click and select Open > System > Defaults.

3. Search for the variable EDB-HLP-TYPE, and change its value from MSH to HTTP to activate the use of the http protocol.

   As this variable is optional, it might not exist. Create the variable EDB-HLP-TYPE with the type String.

4. Proceed as described under Java and Web Client.

5. Unzip the file and set the respective defaults.

<div style="text-align: right">

Chapter 12

</div>

Appendix

Environment Configuration Parameters

Each axalant server process has an environment, which is set on startup. An axalant environment consists of environment variables which are set thru shell scripts and additional configuration values read by the axalant server process at startup from xml configuration files. Information about DataView server environment variables can be found in the DTV User's Guide.

Note The next two sections describe the internal Agile e6 startup of an axalant server process.

Startup Process on Windows

- Invoking axalant server startup thru DataView Daemon or Java Daemon

- %ep_root%\axalant\cmd\axalant_srv.cmd
  - %ep_root%\axalant\cmd\plm_env.cmd (read for all applications)
  - %ep_root%\axalant\cmd\plm_env_dev.cmd (if exists, read for all applications, not available in standard installation)
  - %ep_root%\axalant\cmd\plm_env_cust.cmd (if exists, read for all applications, not available in standard installation, customers should add their modifications for all applications here)
  - %ep_root%\init\%env_name%.cmd (read for specified application only)
  - %ep_root%\init\%env_name%_cust.cmd (if exists, not available in standard installation, read for specified application only, customers should add their modifications for all applications here)

Startup of axalant server process which reads:

- %ep_root%\init\axalant.xml (read for all applications which are defined in "%ep_root%\init\%env_name%.xml", which is the default)

- %ep_root%\init\%env_name%.xml

Startup Process on Unix

- Invoking axalant server startup thru DataView Daemon or Java Daemon

- ${ep_root}\axalant\scripts\axalant_srv
  - ${ep_root}\axalant\scripts\plm_env.sh(read for all applications)
  - ${ep_root}\axalant\scripts\plm_env_dev.sh(if exists, read for all applications, not available in standard installation)
  - ${ep_root}\axalant\scripts\plm_env_cust.sh(if exists, read for all applications, not available in standard installation, customers should add their modifications for all applications here)

- ${ep_root}\init\${env_name}.sh(read for specified application only)
- ${ep_root}\init\${env_name}_cust.sh(if exists, not available in standard installation, read for specified application only, customers should add their modifications for all applications here)

Startup of axalant server process which reads

▫ ${ep_root}\init\axalant.xml (read for all applications which have defined it in "${ep_root}\init\${env_name}.xml", which is the default)

▫ ${ep_root}\init\${env_name}.xml

Startup Shell Scripts

This section describes the environment variables which can be modified by the customer. If the environment variable should be set for all applications use "plm_env_cust.<extension>", if the environment variable is application specific then use "%env_name%_cust.<extension>". If the files do not exist, they must be created.

Environment Variable	Description	Values	Optional	Example
axalant_tmp	Path to the PLM server log files.	$ep_root/tmp	no	$ep_root/tmp or %ep_root\tmp
axalant_data	Path to the PLM root	$ep_root	Not used in code! Maybe obsolete	$ep_root or %ep_root
ORACLE_HOME	Path to the oracle client home directory.	-	no	D:\oracle\product\11.1\db_1
NLS_LANG	National Language Set for the oracle client to use for the axalant server process.	-	no	AMERICAN_AMERICA.WE8MSWIN1252
TNS_ADMIN	Path to the tnsnames.ora for the oracle client to use for the axalant server process.	-	no	%ORACLE_HOME%/network/admin
EP_DEBUG	A comma separated list containing the modules that should generate debug output. Special entries are _all_ to debug all modules, Main for the main routine and 0 to turn off debug output. If debug output is enabled, each e6 process creates an axalant-<hostname>-<pid>.err and .out file to capture stderr and stdout. Be aware that using the value _all_ will have a negative	_all_, 0, or a comma separated list of module IDs (as defined in axalant.xml or ebd_mid.h).	Yes	Main,Mod,Epq,Lgv

Environment Variable	Description	Values	Optional	Example
	performance impact on the e6 server, and it will generate huge log files.			
EDB_LOGDIR	Directory for the log files created by the EP_DEBUG setting. Default is **${ep_root}/tmp**.	Directory with path	Yes	/my/log/dir
EDB_LOGSIZEBUFFER	Buffer size in bytes for stderr output. If 0 is specified, the output will be unbuffered. A value of 0 should only be used if crashes occur and the log buffer is not flushed to disk. Maximum size used is currently 1024, values greater than 1024 are ignored.	0 to 1024	Yes	1024
EDB_TRC_ALL	Deprecated. Use **EP_DEBUG** with **Epq** instead.			
EDB_TRC_DEBUG	Deprecated. Use **EP_DEBUG** with **Epq** instead.			
PATH	PATH to use for the axalant server process.	-	no	PATH=%ORACLE_HOME%/bin;%PATH%

Startup Configuration Files

In e6.1, the application configuration files are in XML format and can be found in the <ep_root>/init directory (e.g. <application>.xml and axalant.xml). They replace the former *.edb files in the same directory. Some basic attributes in the application specific <application>.xml file can be modified over the admin client web interface. If you need to change the enhanced attributes the files can be edited manually with an editor.

All changes not performed over the admin client web interface will be lost if you use the admin client web interface again to change the values of the application.

<application>.xml

The following section describes the attributes of the xml nodes that can be defined in the xml file. In a standard installation not all attributes are defined in the xml file and default values are used.

General Node

```
<General SignalFlag="1" ModuleConfig="axalant.xml" TraceConfig=""
UseCommonTraceFile="1"/>
```

Attribute	Description	Values	Optional	Example
SignalFlag	If the entry is 1, e6 catches runtime errors	0 or 1	Yes	1

Attribute	Description	Values	Optional	Example
	and stops the server process in a controlled manner. For instance, all database connections are disconnected. This might cause hanging server processes and is therefore deactivated by default.			
ModuleConfig	Contains the name of the configuration source which is used as the primary source for the e6 module definitions. All modules of this source are registered. From the original configuration source, only the modules entered in **[Modules\Custom]** will be registered. The module IDs must be unique over all sources.	File name of the primary module source: **[X:][path]filename.xml** **[F:][path]filename.edb** If specified without type F or X, the same type as the original source is assumed. If specified without path, ${ep-root}/init is used.	Yes	axalant.xml
TraceConfig	The Path to the C++ trace configuration file.	F:<path to trace config file>	Yes	F:D:\oracle\plm61\axalant\ini\trace.edb
UseCommonTraceFile	If the entry is 1, the PLM server writes all common traces (SQL, C/C++ and LogiView) into the standard server trace.	0 or 1	Yes	1

Database Node

```
<Database Library="epq10c_ora111" Vendor="Oracle" Version="111"
User="yin@melon" Pwd="yin"
      DbBlobLocation="edb_lob"
      ParallelConnect="NO_PARALLEL_CONNECT"
      ParallelConnectTimeout="10"
      DbWuqSP="Static"
      BindMode="All"
      Wildcards="?%"
      Querymode="MIXED">
   </Database>
```

Attribute	Description	Values	Optional	Example
User	The schema name and the sqlnet connect string of the database to which the application should connect to.	e.g. plmref@plm61	n	-
Pwd	The password of the database schema to which the application should connect to. Can encrypted but also cleartext passwords are possible.	e.g.: plmref or encrypted e.g: RSA-PUBLIC-BASE64:JoMHOs.....NpMCDcytN+DJI=	n	-
DbBlobLocation	Tablespace for the database Blob fields	edb_lob		
ParallelConnect	This setting manages the ORACLE database connection behavior when a data record is going to be updated with the function **"epqupdpar()"**. The major use case for this functionality is the number server so that drawing a new number from a number cycle is done in a parallel connection. The concurrency of the requested data modification is vastly improved by enabling a parallel connection. Important: This setting does not influence the standard behavior of any database transaction, it is only important for a very small number of functions explicitly using **"epqupdpar()"** instead of **"epqupd()"**	NO_PARALLEL_CONNECT - Update of the data record via **"epqupdpar()"** will be executed within the current transaction. The concurrency of an application transaction is low and this setting should be used only for small installation size and/or small number of statements in a transaction. PERMANENT_CONNECT - A **permanent** parallel database connection will be opened when an update of a data record via **"epqupdpar()"** is requested. This connection will require additional memory resources in favor of a fast response time for an update. The concurrency of an application transaction is high. ON_DEMAND_CONNECT - A **temporary** parallel database connection will be opened when an	Y, Default in EPQ: 'NO_PARALLEL_CONNECT' Default in Installation: 'ON_DEMAND_CONNECT'	-

Attribute	Description	Values	Optional	Example
	update of a data record via **"epqupdpar()"** is requested. This connection will require temporarily additional memory resources and will also take more time for execution due to the time required to open and close a parallel connection. Closing the parallel connection is influenced by the **"ParallelConnectTimeout"**. The concurrency of an application transaction is high. This setting is the best one if the memory resources are very limited.			
ParallelConnectTimeout	Determines when the parallel connection in mode **"ON_DEMAND_CONNECT"** will be closed after <ParallelConnectTimeout> seconds due to inactivity. After each read-only statement a check occurs when an update in the last parallel connection has happened and if the timeout has been exceeded the parallel connection is closed.	Value in seconds e.g. 10 for 10 seconds inactivity A value of **0** will close immediately the parallel connections after it has been used.	Y, Default in EPQ: 0 Default in Installation: 10	-
DbWuqSP	Defines which algorithm will be used for a "Where Used Query" or "Structure Explosion". There is a "dynamic" version available which created during runtime temporarily a stored procedure which is suited to the requested query for a specific table. The other version uses a stored procedure which is static and part of the standard database dump.	Static - Static stored procedure in database dump Dynamic - Stored procedure which will be created during runtime	Y, Default is: 'Static'	

Attribute	Description	Values	Optional	Example
	Note This is an internal parameter, please change it only after advised by a support engineer.			
BindMode	Manages the usage of place holders in the dynamic SQL statements. Using place holders is important to avoid the need to parse SQL statements if many SQL statements have the same structure but different values for variables. Not using place holders will add the parsing time on database server side to the execution time for each SQL statement. **Note** This is an internal parameter, please change it only after advised from a support engineer.	**Value is an integer handled as a bit mask; each value is specific for a condition value in a WHERE-clause. The value in a clause is the SQL keyword:** Intervall (between): 1 Equal (=): 2 Like (like): 4 Less than or greater than: (<,>):8 Less equal or greater equal: (<=,>=): 16	Y, Default is 31, for Oracle 8 Databases it has been only Equal(2)	Using place holders for all conditions:31 Using place holders only for an Interval and Equal conditions:3
Wildcards	Sets the characters to define the single and the multi wildcard in the database. A single wildcard is a place holder in a query for exactly one character; a multi wildcard is a placeholder for zero to many characters. There are many places in the application to define wildcards, it is important to understand in which order the wildcard definitions are evaluated. Please see here the order of evaluation (first line is also first evaluated): Configuration parameter (this entry here) **Wildcards** Command line parameter: **-w** DataView Defaults SYSTEM parameter: **WILDCARD** DataView Defaults USER parameter: **WILDCARD** To simplify the wildcard	Any combination of 2 ANSI characters, special characters are recommended	Y, Default: '?%'	Setting an asterisk for multi wildcard like on a Windows or UNIX shell: '?*'

Attribute	Description	Values	Optional	Example
	definition it is strongly recommended to use the **WILDCARD** parameter in the DataView defaults only.			
Querymode	Defines the mode how a query for data will be handled. Queries for exact matches and wildcard matches are distinguished. An exact match is containing no wildcard e.g. 'abcdef' while a wildcard match contains any combination of wildcards e.g. 'a?cd%f'. There are many places in the application to define the query mode, it is important to understand in which order the query mode definitions are evaluated. Please see here the order of evaluation (first line is also first evaluated): Configuration parameter (this entry here) **Querymode** Command line parameter: **-q** DataView Defaults SYSTEM parameter: **QUERYMODE** DataView Defaults USER parameter: **QUERYMODE** To simplify the query mode definition it is strongly recommended to use the **QUERYMODE** parameter in the DataView defaults only.	**SENSITIVE** - Queries will distinguish upper and lowercase character. This will provide the best database performance. **INSENSITIVE** - Queries will not distinguish between upper and lower case characters. **MIXED** - Queries will use 'SENSITIVE' for exact matches and 'INSENSITIVE' for wildcard matches. This provides the best balance between performance and usability of a query	Y, Default in EPQ: 'SENSITIVE' Default in DTV:'MIXED'	A query for 'abc' match in SENSITIVE: only 'abc' INSENSITIVE: 'abc', 'Abc', 'ABC', 'aBc', ... MIXED: for exact matches only 'abc', for wildcard matches 'abc', 'Abc', 'ABC', 'aBc', ...

Security

```
<Security KeyStoreFile="file://D:/oracle/plm61/init/ewallet.p12"
      Wallet="cwallet.sso"
      KeyAlias="C=DE,ST=Baden,L=Karlsruhe,O=Oracle,OU=Agile PLM,CN=PLM"
      TicketKey="RSA-PUBLIC-
BASE64:D8VyVSBxMfgXcZ8AXhOSZMI6Agh4IVQdU49RgszulDGm+z7dQbSIBYRWbpdfsYgP
s4GmjQL//tVYLdttGvLw6n2uN4/iwLFjGO93PtzGuX7TGqWZQkgXR4pGw7M2KjXMDNN/nIL
9rwlWyRKYLOzHZka1ZMgaopEuwPRmsqoQ2lU=">
   </Security>
```

Attribute	Description	Values	Optional	Example
KeyStoreFile	The location of the Oracle Wallet file used by the PLM server.	file://%ep_root%/init/ewallet.p12	no	file://D:/oracle/plm61/init/ewallet.p12
Wallet	The name of the Oracle Wallet file used by the PLM Java applications.	cwallet.sso	no	Do not change the default value!
KeyAlias	The identification of the PLM server key.	C=DE,ST=Baden,L=Karlsruhe, O=Oracle,OU=Agile PLM,CN=PLM	no	Do not change the default value!
TicketKey	The encrypted key for the ticket module.	RSA-PUBLIC-BASE64:........	no	Do not change the default value!

IPC Node

```
<IPC AbsEciUrl="eci://www.example.com:19997" SecurityLevel="process"
TicketLifeTime="600"
        Protocol = "1"
        Node = "www.example.com">
    </IPC>
```

Attribute	Description	Values	Optional	Example
AbsEciUrl	The url where the business service can be reached on the oracle application server. The port must match the "Port" entry in the ABS_<env>.ini file on the oracle application server.	eci://www.example.com:19997	-	-
Protocol	(also configurable in the command line) If the configuration file or command line contains both entries and if the command line also contains the resource to be used, e6 is started in ECI Server mode and can be contacted by ECI clients via the specified parameters (see ECI Manual).	<ECI Protocol ID>	Yes	1
Node	The host name of the e6 server when running as an ECI server.	Fully qualified host name	Yes	example1.oracle.com

Attribute	Description	Values	Optional	Example
SecurityLevel	This entry specifies the security level to be used for IPC connections. The default value of **connection** should be used whenever possible, to ensure that no unauthorized access is possible. Use the other values only if a legacy integration is not capable of passing credentials during an ECI connect. An e6 environment using a value other than **connection** should be secured by firewalls, so that only the legacy system has remote access to the e6 server.	One of: **unrestricted**: no authorization required to establish the connection. **process**: first IPC connection needs to authorize by passing credentials. **connection**: each IPC connection needs to authorize by passing credentials. **[default]**	Yes	process
TicketLifeTime	The life time value of PLM tickets for multiple ECI connections.	expire value in seconds	Yes	21600 (default)

Note   In case of long processes, as e.g. batch processes, it is recommended to set the time value for the TicketLifeTime to "0", thus defining an indefinite timeout.

Modules Node

```
<Modules>
      <Core>
          <Class Name="Classification" Library="epsrv_edb"
Type="embedded" Startup="immediately"/>
      </Core>
      <Custom>
          <Sample Name="Sample Custom module" Library="sample"/>
      </Custom>
   </Modules>
```

The **Modules** element contains two child elements called **Core** and **Custom**. Each of these child elements may contain any number of module definitions, where the element name is used as Module ID.

The attributes of each element are as follows:

Attribute	Description	Values	Optional	Example
Name	A short description of the module. This is displayed in the library list of e6.	A human readable name	No	Sample Module
Library	The name of the library to be loaded, if possible without path and	Library name	No	epsrv_sample

Attribute	Description	Values	Optional	Example
	operating-system-specific suffix			
Entry	The name of the entry function of the module. This information is provided by the supplier of the respective module. If the entry does not exist, it is derived from the module ID as follows: **<mod-id>_Entry**	Function name	Yes	Sample_Entry
Startup	Controls the automatic start-up of a module. If the entry does not exist, the module is only loaded, initialized and started on demand.	One of ▫ **immediately**: Start the module immediately during server startup. ▫ **onDemand**: Start if a userexit requests the module [default] ▫ **disabled**: Do not start the module.	Yes	onDemand
Type	Specifies the module type, standard is dynamic.	One of: Static ▫ **embedded**: Only used for internal e6 modules. ▫ **dynamic**: Standard value for external modules. **[default]**	Yes	dynamic

LogFileMgr/CpsVerify Node

```
<LogFileMgr>
      <CpsVerify Mode="append" Prefix="CpsVerify">
      </CpsVerify>
   </LogFileMgr>
```

Attribute	Description	Values	Optional	Example
Mode	Logfile mode. For CpsVerify the value is overwritten by user specific DTV default "CPS_LOG_FIL_MOD". So adapt the DTV default instead of changing value here.	▫ **trunc** = truncate log message file before writing log messages ▫ **append** = append log messages to file	yes	Mode="append"
Prefix	Prefix for logfile. Not evaluated yet for CpsVerify, prefix is always set to	CpsVerify	yes	Prefix="CpsVerify"

Attribute	Description	Values	Optional	Example
	value "CpsVerify".			

PLMPresentationServices Node

```
<PLMPresentationServices
Report_Service_URL="http://www.example.com/reporter/report"/>
```

Attribute	Description	Values	Optional	Example
Report_Service_URL	Lightweight reporting URL, generated automatically. URL on the configured oracle application server.	http://www.example.com:7777/reporter/report		

Environment Node

```
Environment>
      <Windows DATAVIEW_TBSP="edb" DATAVIEW_IXSP="edb_idx" ...
DATAVIEW_DUMP="<EP_ROOT>\axalant\dmp" ...>
          <intel-ms-nt5.0/>
      </Windows>
      <Unix ... >
          <ia32-linux-sles10/>
          <rs6000-ibm-aix5.3/>
          <sparc-sun-solaris10/>
          <hppa-hp-hpux11.23/>
          <ia64-hp-hpux11.31/>
      </Unix>
   </Environment>
```

Environment variables for the axalant server process can be set here, or in the startup scripts defined above.

These values will overwrite all previously defined values from e.g. startup scripts.

Ports

The *Internet Assigned Numbers Authority* (IANA) administrates the port numbers in the range of 0 to 65535.

When it comes to assigning port numbers for services that are not registered, only port numbers of the so-called dynamic (private) range of 49152 to 65535 should be assigned in order to meet minimum security requirements. However, conflicts with already installed applications can occur.

But in practice already numbers from 1024 on are used for only the range of 0-1023 is protected. For example, operating systems should only allow processes with appropriate privileges the opening of server ports that are within that range.

Prior to any installation, you should contact your system administrator in order to evaluate the ports that are already in use by the system and applications. Thus conflicts can be avoided when assigning ports that are used by Agile e6. Additionally, an existing firewall needs to be configured

accordingly.

In case a system administrator is not available, a list of currently used TCP- and UDP –ports can be created with the command `netstat –a`. The RPC ports that are available through PortMapper can be determined using the command `rpcinfo –p`.

Range of ports

The port numbers are divided into three ranges:

1. the well known ports

2. the registered ports

3. the dynamic and/or private ports.

The well known ports are those from 0 through 1023. DCCP well known ports should **not** be used without IANA registration. The registration procedure is defined in (RFC4340), section 19.9.

The registered ports are those from 1024 through 49151. DCCP registered ports should **not** be used without IANA registration. The registration procedure is defined in (RFC4340), section 19.9.

The dynamic and/or private ports are those from 49152 through 65535.

> **Note** Unassigned port number should not be used. The IANA will assign the number for the port after your application has been approved.

> **Note** Assignment of a port number does not in any way imply an endorsement of an application or product, and the fact that network traffic is flowing to or from a registered port does not mean that it is "good" traffic. Firewall and system administrators should choose how to configure their systems based on their knowledge of the traffic in question, not whether there is a port number registered or not.

Well known port numbers

The well known ports are assigned by the IANA and on most systems can only be used by system (or root) processes or by programs executed by privileged users.

Ports are used in the TCP [RFC793] to name the ends of logical connections which carry long term conversations. For the purpose of providing services to unknown callers, a service contact port is defined. This list specifies the port used by the server process as its contact port. The contact port is sometimes called the "well-known port".

To the extent possible, these same port assignments are used with the UDP [RFC768]. The range for assigned ports managed by the IANA is 0-1023.

Registered port numbers

The registered ports are listed by the IANA and on most systems can be used by ordinary user processes or programs executed by ordinary users.

Ports are used in the TCP [RFC793] to name the ends of logical connections which carry long term conversations. For the purpose of providing services to unknown callers, a service contact port is defined. This list specifies the port used by the server process as its contact port.

The IANA registers uses of these ports as a convenience to the community. To the extent possible, these same port assignments are used with the UDP [RFC768]. The Registered Ports are in the range 1024-49151.

Dynamic and/or private ports

The Dynamic and/or Private Ports are those from 49152 through 65535.

Range of values and Dependencies

Service	Ports (default value)	Dependencies
Sun Portmapper (RPC)	111	Under Unix always present, under Windows a component of the e6 delivery
Admin Server	□ HTTP (8027) □ HTTPS (8028)	
DataView Daemon	□ RPC Port (552000566) □ one free port per application server	Sun Portmapper
Java Daemon	□ StandardPort (16077) □ AdminPort (16078), only local □ RegistrationPort (16079), only local □ One free port from the port range per application server	
FileServer	□ RPC port (804257548) □ Web File Service (8088) □ one free port per client connection	Sun Portmapper FELICS Web Presentation Service
EDM Server	Per session one port assigned from the daemon and a free port for the CORBA communication to the Business Service.	Sun Portmapper FELICS Business Service FileServer
Web Presentation Service	Ajp12 Port (7077), integration TomCat in Apache Shutdown Port (8005) Web Client (8088) Web Report Service (8088) ViewCafé (2099)	Java Daemon
Business Service	CORBA Name Service (2571) JMX (12808) ECI Port (19997) one free port per connection to the e6 server	Java Daemon e6 Server Zugriff auf Mail-Port (25).
Java Client	ECI Topic (4444) Needs to be distinct for every client	Java Daemon e6 Server

Service	Ports (default value)	Dependencies
	call und can be set with the start.	
DataView Client	ECI Topic (DDE: eci_dde_loop) Needs to be distinct for every client call und can be set with the start. Instead of DDE also TCP/IP is possible.	Sun Portmapper DataViewDaemon e6 Server
Workflow Editor		Business Service (ECI Port)
Office Suite	DDE/OLE/COM	DataView Client e6 Server
EIP	Queue Port (9001) Admin Port (9876) Log Port (4445) Web Server (8080) synchronous: ECI Server Port (19997) Note Here there's a conflict with the standard ECI port of the Business Service.	Java Daemon e6 Server