

***AutoVue Integration SDK and Sample
Integration for Filesys DMS
Installation Guide***

ORACLE

February 28, 2011

Copyright © 1998, 2011, Oracle and/or its affiliates. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software-Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

PREFACE.....	5
Audience	5
Documentation Accessibility	5
Accessibility of Code Examples in Documentation	5
Accessibility of Links to External Web Sites in Documentation	5
TTY Access to Oracle Support Services	5
Related Documents.....	5
Conventions.....	6
INTRODUCTION	7
SYSTEM REQUIREMENTS	8
Required Software.....	8
Server.....	8
Windows	8
Linux.....	8
Client.....	8
Application Servers	8
INSTALLATION PREREQUISITES	9
Downloading Required Software	9
Oracle JDeveloper 11g	9
Eclipse	9
Oracle AutoVue	9
WebLogic Server.....	9
INSTALLATION.....	10
Integration SDK Package Checklist	10
Create FileSys Repository.....	11
Create Server Runtime Environment on IDE	11
Create Default Runtime Environment on IDE	11
Create Server Runtime on Eclipse.....	11
Creating Sample Projects on IDE	12
Creating Sample Projects on JDeveloper	12
Creating Sample Projects on Eclipse.....	13
Filesys Project Configuration.....	14
Copying AutoVue Jar Files	14
Configuring AutoVue Server.....	14
Configuring log4j.properties for Debugging	14
Configuring RootDir for the Filesys Repository	15
Configuring Embedded vs. Pop-Up Window	15
Configuring Markup Policy.....	16
Configuring User Control.....	16
Configuring Pick List	17
Configuring Thumbnail Display.....	17
Configuring Redirection	17
Configuring Real-Time Collaboration (RTC) Demo	18
Configuring Oracle Enterprise Visualization Framework (OEVF).....	19
Configuring New Sample Data.....	20
Configuring ISDK Skeleton Project	21
Copying AutoVue Jar Files	21

Configuring AutoVue Server.....	21
Configuring log4j.propertiesfor Debugging.....	21
Configuring ISDK Web Service Client Project.....	21
Copying AutoVue Jar files	21
Configuring AutoVue Server.....	21
Configuring log4j.properties for Debugging.....	21
Configuring SOAP Handler.....	22
Defining Location of BluePrint WSDL.....	22
Running Filesys Project	23
FEEDBACK	25
General Inquiries.....	25
Sales Inquiries.....	25
Customer Support	25

Preface

The *AutoVue Integration SDK and Sample Integration for Filesys DMS Installation Guide* describes the procedure for building and running ISDK samples on IDEs that support Java EE 5 and Web applications.

For the most up-to-date version of this document, go to the AutoVue Documentation Web site on the Oracle Technology Network (OTN) at <http://www.oracle.com/technetwork/documentation/autovue-091442.html>.

Audience

The *AutoVue Integration SDK and Sample Integration for Filesys DMS Installation Guide* is intended for third-party developers who want to run a dynamic Web project in the Eclipse Web Tools Platform (WTP) IDE for the Oracle AutoVue and the Filesys DMS sample

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

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.

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.

Related Documents

For more information, see the following documents in the Integration SDK documentation library:

- *Overview*
- *User Guide*
- *Technical Guide*
- *Release Notes*
- *Java Docs*
- *Security and Authentication Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in the text.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Introduction

This document describes the procedure for building and running a dynamic Web project in JDeveloper and Eclipse IDEs for the Oracle AutoVue.

System Requirements

The recommended system hardware configuration is:

- A system supporting the JDK/JRE version 6 with at least 2GB of main memory.
- At least 100MB of free disk space to install the software components and examples.

Required Software

- Oracle AutoVue 20.1

Server

The following operating systems have been certified with the Integration SDK:

Windows

- Windows 2008 64-bit (AutoVue running in 32-bit mode)
- Windows 2003 32bit

Linux

- Redhat Linux Enterprise AS version 5
- Oracle Enterprise Linux 5

Client

The following Java Virtual Machines have been certified with the Integration SDK:

- IDE that supports Java EE 5 and Web applications
 - Oracle JDeveloper 11g
 - Eclipse IDE 3.5 (Galileo) Edition
- Java JDK 1.5 for Filesys Sample and Skeleton
- Java JDK 1.6 for Web Service Client (recommended JDK 1.6_14)

The following Web browsers are certified with the Integration SDK:

- Microsoft Internet Explorer 7 or higher
- Mozilla Firefox 3.0 or higher

Application Servers

The following application servers are compatible with the Integration SDK:

- Oracle WebLogic Server 10gR3
- Oracle WebLogic Server 11gR1 (certified for Web Service security)
- Any other application server that supports Servlet 2.5 may work but are not certified by Oracle

Installation Prerequisites

This manual assumes you are familiar with Java development and with basic Web application development concepts, such as deployment descriptors and WAR archives. Understanding XML language is beneficial, but not mandatory.

The software products listed in the [System Requirements](#) must be installed and configured on your system according to the manufacturer's instructions.

Useful commands on UNIX and Linux:

- To extract files from a TAR file, use the tar command: `tar -xvk <tarfile>`
- To unzip GZIP files, use the gunzip command: `gunzip <gzip file>`
- To unzip ZIP files, use the unzip command: `unzip <zip file>`

Downloading Required Software

Oracle JDeveloper 11g

You can download Oracle JDeveloper Studio Version (jdevstudio11112install.exe) from <http://www.oracle.com/technology/software/products/jdev/htdocs/soft11.html>.

Eclipse

Oracle Enterprise Pack for Eclipse is a free set of certified plug-ins, enabling WebLogic developers to support Java EE and Web Service standards. The Oracle Enterprise Pack for Eclipse All-In-One installer includes a preconfigured version of Eclipse and the OEPE plug-ins. You can download the Eclipse 3.5 (Galileo) Edition for your desired platform from http://www.oracle.com/technology/software/products/oepe/oepe_11gR1.html. If you download [Eclipse IDE for Java EE Developers](#) from the Eclipse Web site, you must download the Oracle WebLogic Server plug-in separately when creating the server.

Oracle AutoVue

Oracle AutoVue 20.1 is available from <http://edelivery.oracle.com>. The description name is Oracle AutoVue 20.1. Select a Media Pack for your desired platform.

WebLogic Server

You can download WebLogic Server from the following location: http://www.oracle.com/technology/software/products/ias/htdocs/wls_main.html.

Installation

Integration SDK Package Checklist

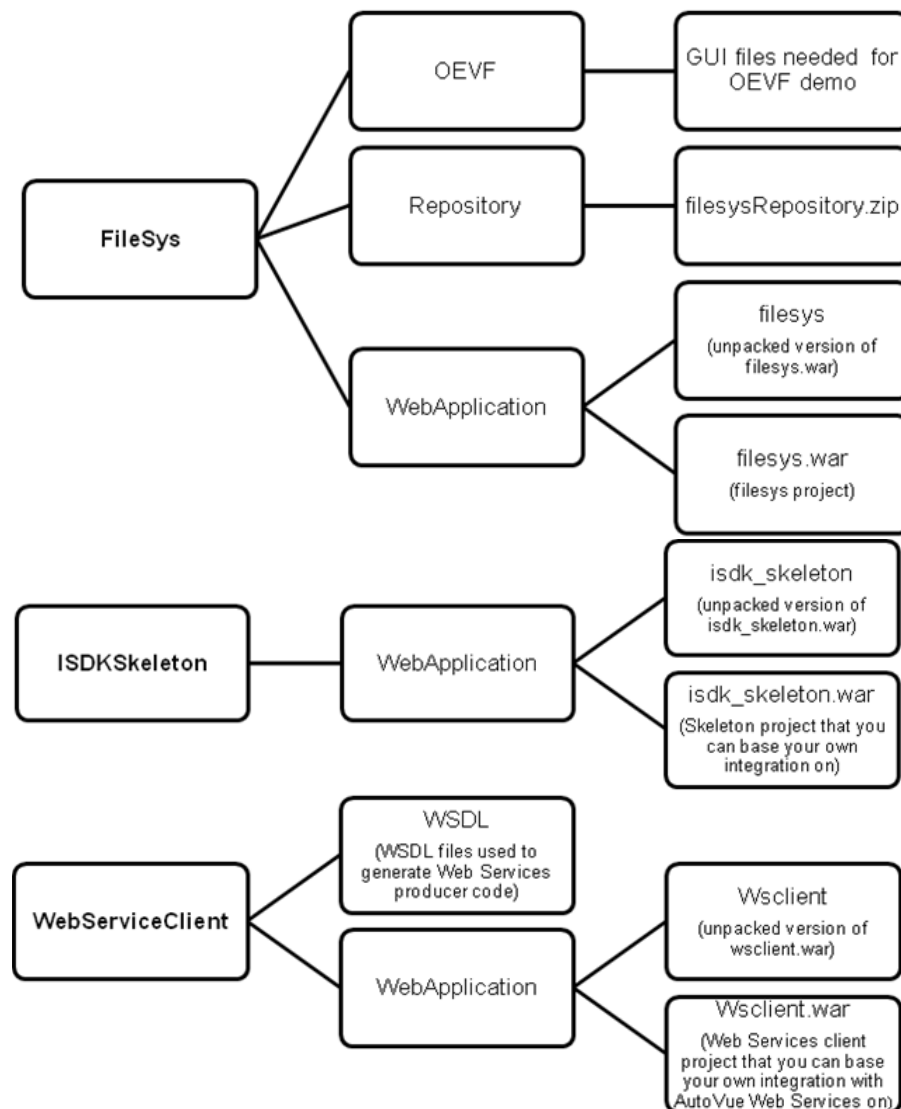
There are two folders included: win32 and linux. Each of these folders contains the installer of the ISDK for the corresponding platform:

- For **Windows** (2008 or 2003), go to the **win32** folder and launch the **setupwin32.exe** file.
- For **Linux** (Redhat Linux Enterprise 5), go to the **linux** folder and launch the **setuplinux.bin** file.

During the installation, provide the **Directory Name** where you want to install the AutoVue Integration SDK.

Note: If the Linux installer is unable to run in graphical mode, install the **libXp** package.

After running the installer, all the required files are created under your AutoVueIntegrationSDK installation directory with the following structure:



Create FileSys Repository

Unzip the fileysRepository.zip file to the location of your choice (<fileys data repository>). The <fileys data repository> is the repository that will contain the Filesys DMS data (documents, xrefs, markups, and so on).

Create Server Runtime Environment on IDE

Create Default Runtime Environment on IDE

JDeveloper has default IntegratedWebLogicServer configured.

Create Server Runtime on Eclipse

You can create a server to identify the runtime environment that you want to use in Eclipse 3.5. To create the WebLogic Server, complete the following steps. Please notice that your Oracle WebLogic Server domain needs to be created in development mode in order to create the server successfully in Eclipse.

You can create a server to identify the runtime environment that you want to use to test your Oracle AutoVue project.

- 1 From the **File** menu, select **New**, and then select **Other**.
- 2 Expand the **Server** folder, then select **Server**.
- 3 Click **Next**.

The Define a New Server wizard opens. This wizard lets you define a new server that contains information required to point to a specific runtime environment for local or remote testing, or for publishing to an application server.

Note: If you install Eclipse using Oracle Enterprise Pack for Eclipse Galileo Edition, Oracle WebLogic Server (10gR3 and 11gR1) will be listed in the New Server wizard under Oracle server type. If you download Eclipse 3.5 directly from Apache Web site, you need to click **Download Additional Server Adapters** and download Oracle WebLogic Server adapter from the Internet yourself.


- 4 You can select Oracle WebLogic Server 10gR3, click **Next** and go to step 5. You can select Oracle WebLogic Server 11gR1, click **Next** and go to step 6. It depends on which version of Oracle WebLogic Server you already have.
- 5 This step is for Oracle WebLogic Server 10gR3 on Windows.
 - a. You need to provide the domain directory at Specify a WebLogic domain directory dialog.
 - b. If you do not have the WebLogic domain available yet or you want to create a different one, click **Click Here to launch Configuration Wizard to create a new domain**. Write down the Domain Location for your created domain, for example, C:\bea\user_projects\domains\base_domain.
 - c. Now suppose you already have a domain directory available. You can input or browse to get it on your machine and click **Next**. Proceed to step 7.
- 6 This step is for Oracle WebLogic Server 11gR1 on Windows.

For Oracle WebLogic Server on Linux, follow similar steps.

 - a. First you need to input WebLogic Home, for example, C:\bea\wlserver_10.3 on Windows and /home/my/bea/wlserver_10.3 on Linux, at the Define WebLogic Runtime dialog and then provide the domain directory at Define a WebLogic Server dialog.
 - b. If you do not have the WebLogic domain available yet or you want to create a different one, click **Click Here to launch Configuration Wizard to create a new domain**. Write down the Domain Location for your created domain, for example, C:\bea\user_projects\domains\base_domain.
 - c. Now suppose you already have a domain directory available. You can input or browse to get it on your machine and click **Next**.

- 7 Select the projects from the available projects list in the Add and Remove dialog and add them to the configured projects list (click the **Add > button**). Then click **Finish**.

Then Oracle WebLogic Server 10gR3 or 11gR1 appears in the Servers view. You can start and stop the Server from this view.

- 8 Open Server view to verify the server has been created. You can click **Servers**  or click on **Window** from menu bar, then **Show View** and **Servers** to display the Server view.

Creating Sample Projects on IDE

The Sample Integration for Filesys DMS is a dynamic web application. In this section we describe how to create a FileSys project on JDeveloper and Eclipse IDEs.

Creating Sample Projects on JDeveloper

- 1 Create an application if you do not have one yet. You can create an application by clicking on **File** from the menu, then select **New** to bring out the New Gallery dialog.
- 2 At this dialog, select **Applications** under the **General** category from the left panel and select **Generic Application** from the right panel.
- 3 Click **OK**. The Create Application dialog appears.
- 4 Complete the Create Application dialog to create an application with the Application Package Prefix field left empty.
- 5 Click **File and Import**. The Import dialog appears.
- 6 Select the WAR File in the Import dialog to bring out the Create Project from WAR File wizard.
- 7 You can finish the same task by right-clicking the application and selecting **New Project**. Select Project from WAR file from the right panel in the New Gallery dialog.
- 8 In the following Create Project from WAR file dialogs, input your project name, choose a directory to put your project, select the WAR file to import, and verify the location for Root Directory for Web Module.
- 9 Click **Finish** to finish the creation of your project.
- 10 In the Project view, browse to verify that your project has been created successfully.
- 11 Click on **Build** to make your project. There should be no compilation error.
- 12 Check Libraries and Classpath:
 - a. Right-click the project and select **Project Properties** to bring out the Project Properties dialog
 - b. Click on **Libraries and Classpath** in the left panel.
 - c. Check the JSP Runtime and JSF 1.2 are available under the Classpath Entries. If there are not available, you can add them manually in the following steps:
 - Click **Add Library** in the right panel
 - Select **JSP Runtime** under Extension from the pop-up window.
 - If you are going to deploy the project later to an external WebLogic Server instead of using the IntegratedWebLogicServer, you also need to add JSF 1.2 under Extension from the pop-up window.
 - Click **OK**.
- 13 This step is for the ISDK Web Service Sample project for WebLogic when the “WeblogicUserNameTokenHandler.java” (Username token profile security for WebLogic) is needed.

First, WeblogicUserNameTokenHandler.java.excluded needs to be renamed to WeblogicUserNameTokenHandler.java.

Second, add weblogic.jar to the project's build path if you see compilation error for WeblogicUserNameTokenHandler.java. The steps are:

- a. Right click the project and select **Project Properties** to bring out the Project Properties dialog.
 - b. Click on **Libraries and Classpath** in the left panel
 - c. Click on **Add JAR/Directory** in the right panel
 - d. In Add Archive or Directory dialog, browse to WebLogic Server's lib folder to select weblogic.jar and click on **Select**.
Weblogic.jar appears in the Classpath Entries.
 - e. Click **OK** to exit the Project Properties dialog.
 - f. Rebuild your project and there should be no compilation error.
- 14 To start the WebLogic Server. You can click on **Run** from menu bar and then click **Start Server Instance** to start or click on from the toolbar.

Creating Sample Projects on Eclipse

- 1 In the **File** menu select **Import**. The Import dialog appears.
- 2 In the Import dialog, expand Web and select **WAR file** and then click **Next** to bring out the WAR Import dialog.
- 3 Import the sample WAR files that you want to import by clicking the **Browse**.

Provide a name for your Web project.

If you have already configured Oracle WebLogic Server runtime, Oracle WebLogic Server 10gR3 or 11gR1 will be shown as Target runtime. If you have not created it yet, you can create one now by clicking on **New**. Please refer to "Create Server Runtime Environment on IDE" (page 11).

- 4 Click **Next**.
- 5 Accept the default at the WAR Import: Web libraries dialog and click on **Finish** to populate the Web project.
- 6 Click **Yes** if Eclipse asks you to open J2EE perspective for this project.
- 7 This step is for the ISDK Web Service Sample project for WebLogic when the "WeblogicUserNameTokenHandler.java" (Username token profile security for WebLogic) is needed.

First, WeblogicUserNameTokenHandler.java.excluded needs to be renamed to WeblogicUserNameTokenHandler.java.

Second, add weblogic.jar to the project's build path if you see compilation error for WeblogicUserNameTokenHandler.java. The steps are:

- a. Right click the project and select **Build Path**, then select **Configure Build Path** to open the Project's Properties dialog.
 - b. Click on the Add Library tab and click on **Server Runtime**, and then click **Next**.
 - c. If WebLogic appears, you can select and add.
 - d. If you cannot find the WebLogic runtime, then click on **Add External JAR** from the previous dialog to open the JAR Selection dialog. Browse to WebLogic Server's lib folder to select weblogic.jar and click on **Open**.
 - e. Weblogic.jar should appear in the Classpath Entries panel.
 - f. Click on **OK** to exit the Properties dialog.
 - g. Recompile the project and there should be no compilation error
- 8 This step is for using Oracle WebLogic 10gR3 Server runtime.
- Open the weblogic.xml file (located at the WebContent\WEB-INF folder) in a text editor.

- Comment out the following lines as they are not supported by WebLogic 10gR3.

```
<wls:session-descriptor>
    <wls:cookie-http-only>false</wls:cookie-http-only>
</wls:session-descriptor>
```

- 9 After completing all these steps, there should be no compilation error about java code in your project.

Filesys Project Configuration

Copying AutoVue Jar Files

Overwrite the files jvue.jar, jogl.jar and gluegen-rt.jar from the directory <AutoVue Installation directory>\bin to your Filesys project's WebContent\jvue folder (for Eclipse) or public_html\jvue folder (for JDeveloper).

Overwrite the file vueservlet.jar from the directory <AutoVue Installation directory>\bin to your Filesys project's WebContent\WEB-INF\lib folder (for Eclipse) or public_html\WEB-INF\lib folder (for JDeveloper).

Configuring AutoVue Server

- 1 Open the web.xml file in a text editor.
- 2 Locate the following block.

```
<servlet id="csi_servlet_2">
<servlet-name>VueServlet</servlet-name>
<servlet-class>com.cimmetry.servlet.VueServlet</servlet-class>
<init-param>
<param-name>JVueServer</param-name>
<param-value>localhost:5099</param-value>
</init-param>
<init-param>
```

- 3 Update the default location of JVueServer "localhost:5099". You must replace localhost with the host name/IP address of the machine that is running AutoVue Server, and replace 5099 with the socket port number that AutoVue Server is listening to (default is 5099).

Configuring log4j.properties for Debugging

The location of log4j.properties file is defined in web.xml. By default, it is located at WEB-INF/lib folder.

```
<init-param>
    <param-name>log4jInitFile</param-name>
    <param-value>/WEB-INF/lib/log4j.properties</param-value>
</init-param>
```

To configure log4j.properties for debugging, do the following:

- 1 With your preferred file editor open the log4j.properties file.
- 2 Set the location and the filename of your log4j logging file, for example, C:/tmp/filesys.log.

```
# setting the logging file
log4j.appender.R.File=<Your logs directory>/<logfile>.log
```

- 3 You can change the level and location of output by modifying this file, for example, log4j.logger.com.cimmetry.vuelink=DEBUG.

The following table shows the different levels of logging available.

Will Output Messages Of Level		DEBUG	INFO	WARN	ERROR	FATAL
Logger Level	DEBUG					
	INFO					
	WARN					
	ERROR					
	FATAL					
	ALL					
	OFF					

 : No

 : Yes

- If you set Logger Level to FATAL, then only output messages of level FATAL are logged in log4j file.
- If you set Logger Level to ERROR, then only output messages of level ERROR or FATAL are logged in log4j file.
- If you set Logger Level to DEBUG, then output messages of any level are logged in log4j file.]

For more information on log4j capabilities, refer to log4j documentation.

Configuring RootDir for the Filesys Repository

- 1 With your preferred file editor open the web.xml file located at WEB-INF folder.
- 2 Replace the param-value for "RootDir". For example, if you have unzipped the Filesys Repository to folder c:\tmp on Windows, the param-value for RootDir will be c:\tmp\filesysRepository.

```
<!-- context parameters are available to all servlets -->
<context-param>
  <param-name>RootDir</param-name>
  <param-value>Put path to repository here</param-value>
</context-param>
```

Configuring Embedded vs. Pop-Up Window

AutoVue applet can be launched in a pop-up window or embedded inside the caller's browser window.

By default, Filesys demo uses embedded mode and RTC demo uses pop-up mode.

For OEVF demo, you decide the mode by providing `embedded=0` or `embedded=1` request parameter in the launching url. See `java/OEVFDemo.html`.

To change the mode in Filesys demo:

- 1 Open `java/frmApplet.jsp`
- 2 Change the line
`boolean embedded = true;`
to
`boolean embedded = false;`

To change the mode in RTC demo:

- 1 Open `java/RTCDemo_init.jsp` and `java/RTCDemo_join.jsp`
- 2 Change the line
`boolean embedded = false;`
to
`boolean embedded = true;`

Configuring Markup Policy

The location of `MarkupPolicy.xml` file is defined in `web.xml` that controls both online and offline markup operation in the Mobile package file. By default, it is located at `WEB-INF/lib` folder.

```
<init-param>
<param-name>CSI_MarkupPolicyDefLocation</param-name>
<param-value>/WEB-INF/lib/MarkupPolicy.xml</param-value>
</init-param>
```

If you need to update the Markup Policy file, please refer to the *Oracle AutoVue User's Manual*. On Windows, the link is <http://localhost/jVue/help/en/AutoVueOnLineHelp.html>. If the link does not work, check whether there is a virtual directory "jVue" with IIS. It is created during AutoVue Server installation.

Configuring User Control

By default, ISDK filesys bundles a file called `credential.txt` that contains valid user information for authentication. The location of `credential.txt` file is defined in `web.xml`.

```
<init-param>
<param-name>CredentialInfoLocation</param-name>
<param-value>/WEB-INF/lib/credential.txt</param-value>
</init-param>
```

To add new users or modify existing user name or password, update `credential.txt`. Each line of the file contains an entry for a user and its password. The field separator is colon ":".

Configuring Pick List

This list is for controlling the content of pick list for Intellistamp DMS properties. You can remove/modify existing values or add new values for the <Status> and <RelatedInfo> elements in WEB-INF/lib/picklist.xml, but you are not supposed to delete these two elements or add new elements directly under <Data> element.

Configuring Thumbnail Display

If you want to show thumbnails based on bmp renditions when browsing the Filesys Repository, you can do the following configuration.

- 1 For Windows system, create a virtual directory on IIS (Internet Information Services) for the Filesys Repository. For example, if you have unzipped the Filesys Repository to folder c:\tmp on Windows, you can create a virtual directory with alias filesysRepository and the location path c:\tmp\filesysRepository. Suppose IIS is available at the default port 80.
- 2 For Linux system, if Apache Server is available, do the following configuration.
 - Open Apache's httpd.conf file, locate the line: DocumentRoot "/var/www/html".
 - Copy this line and comment out the original one.
 - Change the copied line to, for example
DocumentRoot "/home/ucm/tmp"
Suppose your filesys repository is unzipped to /home/ucm/tmp folder and your /home/ucm/tmp/filesysRepository folder allow executing file as program. If your DocumentRoot has already been used, you need to put your filesys repository under the existing DocumentRoot folder in order to preview thumbnails.
 - Save the file and restart Apache Server.
- 3 Replace the param-value for RootURL in web.xml. This URL is mainly used for thumbnail displaying. But you need to put a URL (for example, http://localhost) there even if thumbnail displaying is not intended. With the configuration sample in Step 1, the param-value for RootURL will be http://localhost/filesysRepository. Pay attention to the case sensitivity of IIS.

```
<!-- This URL is only needed to construct thumbnail URLs -->
<context-param>
<param-name>RootURL</param-name>
<param-value>http://localhost/filesysRepository </param-value>
</context-param>
```

Configuring Redirection

In order to experiment redirection functionality in Filesys, you need to install IDE and deploy the Filesys sample project on two machines—a main server and a remote server—and finish generic configuration and other configurations based on your needs. You must then perform the following configurations for redirection.

- 1 On the main server machine, change the folder permission for the filesys repository to "Full Control" for all users.
- 2 On the remote server machine, create a network mapping drive to the filesys repository directory on the main server machine. In Filesys demo, both remote server and the main server will use the same filesys repository data.

- 3 On the main server, modify web.xml to comment out the blocks "RemoteVueLink", "RemoteVueServer" and "RemoteVueServlet". Specify the param-values for these three parameters.

Param-name	Description and param-value
RemoteVueLink	URL to the remote vuelink The param-value is http://host:port/context/servlet/FilesysVuelink where host is the remote host name or IP address, port is the remote IDE's server runtime port number, context is the Filesys project name on the remote IDE.
RemoteVueServer	Hostname or IP address of the remote AutoVue server The remote server can use another AutoVue Server instead of the one running on the main server.
RemoteVueServlet	URL to the remote VueServlet The param-value is http://host:port/context/servlet/VueServlet

For Example:

```

<context-param>
<param-name>RemoteVuelink</param-name>
<param-value> http://sremote:7001/ISDK_Remote/servlet/FilesysVuelink</param-value>
</context-param>

<context-param>
<param-name>RemoteVueServer</param-name>
<param-value>sremote</param-value>
</context-param>

<context-param>
<param-name>RemoteVueServlet</param-name>
<param-value>http://sremote:7001/ISDK_Remote/servlet/VueServlet</param-value>
</context-param>

```

Configuring Real-Time Collaboration (RTC) Demo

Verifying RTC Demo

Make sure the WEB-INF/lib/credential.txt has an entry for user "rtc" and "rtc1". Although every valid user can initiate and join a meeting, but by default the meeting is initiated as user "rtc" and joined by user "rtc1" and the AutoVue applet is named after the username.

Creating or Updating meetingfiles.txt

Verify that the meetingfiles.txt under your <filesys data repository>/Meeting folder exists. If this file does not exist, you need to create it manually.

If you want to change the files shown in the Meeting File drop down list when initiating a RTC meeting from RTCDemo_init.jsp page similar to the following figure, then you need to update the meetingfiles.txt file.

Each entry in the meetingfiles.txt file represents one meeting file; it starts with "/" and reflects one viewable document file in the Filesys data repository.

Configure AutoVue Server and Restarting

Open jvueserver.properties file in AutoVue server's bin folder and locate the following line:

```
dms.vuelink.version=19.3
```

Comment out this line in order to avoid an Authorization pop-up when initializing or joining a RTC meeting using ISDK 20 RTC demo:

```
# * To be used with 19.3 VueLinks
# dms.vuelink.version=19.3
```

Restart AutoVue Server after the above modification.

Configuring Oracle Enterprise Visualization Framework (OEVF)

Defining OEVFInfoLocation in web.xml

By default, ISDK filesys bundles a file called oevf.xml which defines the mapping of document IDs with assetIDs and workflowIDs. The default location of oevf.xml is under the folder WEB-INF/lib. If you move the file to another location, then you need to specify the full path for the parameter "OEVFInfoLocation" in web.xml.:

```
<!--
# the location of xml file which contains all the info about assetIDs, workflowIDs
# and full path of the latest revision in FileSys DMS
-->
<init-param>
<param-name>OEVFInfoLocation</param-name>
<param-value>/WEB-INF/lib/oevf.xml</param-value>
</init-param>
```

Updating oevf.xml

This step is required if you want to establish new or update existing mappings of document IDs with assetIDs and workflowIDs.

The root element of the oevf.xml file is <data>. The direct elements under <data> are <file> elements that contain the definition for files. Each <file> element represents one file. If you want to add mapping relationships for a new file, then you need to add a new <file> entry.

A `<file>` element can include multiple `<revision>` elements that represent the multiple revisions of the file. If you want to add a new revision section to an existing file, then you must add one new `<revision>` entry.

Each `<version>` element includes a `<docID>`, `<assetIDs>`, `<workflowIDs>` and `<version>` elements. The value for `<version>` element is the revision number. The value for `<docID>` element is the relative path to a file in the Filesys data repository. It starts with `"/"`. For example, `"/2D/MicroStation.dgn/MicroStation.dgn(2)/MicroStation.dgn"`.

The `<assetIDs>` element can contain multiple `<assetID>` elements and the `<workflowIDs>` element can contain multiple `<workflowID>` elements. You can add or delete an `assetID` that is associated with one revision of a file by adding or deleting element a `<assetID>` element. You can add or delete a `workflowID` that is associated with one revision of a file by adding or deleting a `<workflowID>` element.

Updating OEVFDemo.html

This step is needed to add new or modify existing test cases for OEVF.

The launching OEVF url defined inside `<a>` tag calls `".../jvue/frmApplet.jsp"` page combined with some of the following parameters.

URL Request Parameter	Value and Description
aID	A Value defined for <code><assetID></code> element in <code>oevf.xml</code> .
docID	A value defined for <code><docID></code> element in <code>oevf.xml</code> .
wID	A value defined for <code><workflowID></code> element in <code>oevf.xml</code> .
embedded	0 or new such parameter: AutoVue applet appears in a new window. 1: AutoVue applet is embedded in the caller's browser window.
goBack	Work together with <code>embedded=0</code> . 0 or no such parameter: The caller's browser displays an empty page with the launching OEVF URL. 1: The caller's browser displays the <code>OEVFDemo.html</code> page.
guiFile	Name of the AutoVue GUI to be used.

You can pass in only aID, only wID, aID with wID, aID with dID, wID with dID, aID with wID and dID in addition with `embedded` or `goBack` or `guiFile` param. Refer to *OEVFDemo.html* for the meaning of different combinations.

Copying OEVF GUI files to AutoVue

Copy `assetView.gui` and `assetEdit.gui` files from inside the ISDK installation `"AutoVueIntegrationSDK/FileSys/OEVF"` folder to the folder `"bin/Profiles"` under AutoVue Server installation folder. If the `"Profiles"` folder does not exist, create one before copying.

Configuring New Sample Data

You can add new data to the existing sample Filesys repository. It is recommended not to rename the folder name or file name, or delete existing data, because the sample data is preconfigured to demonstrate certain functionalities (for example, for RTC Demo and OEVF demo).

Refer to the section "Add new data to the document repository" in the *User Guide* for information on how to manually, or using JDeveloper or Eclipse, add new data after you create the Filesys project.

Configuring ISDK Skeleton Project

Copying AutoVue Jar Files

Copy the files jvue.jar, jogl.jar and gluegen-rt.jar from the directory <AutoVue Installation directory>\bin to your Filesys project's WebContent\applet folder (for Eclipse) or public_html\applet folder (for JDeveloper).

Copy the file vueservlet.jar from the directory <AutoVue Installation directory>\bin to your Filesys project's WebContent\WEB-INF\lib folder (for Eclipse) or public_html\WEB-INF\lib folder (for JDeveloper).

Configuring AutoVue Server

Configuring the AutoVue Server for the ISDK Skeleton project follows the same steps as "Configuring AutoVue Server" (page 14).

Configuring log4j.properties for Debugging

Configuring the log4j.properties for debugging follows the same steps as "Configuring log4j.properties for Debugging" (page 14).

Configuring ISDK Web Service Client Project

Copying AutoVue Jar files

Configuring the AutoVue jar files for the ISDK Web Service client project follows the same steps as "Copying AutoVue Jar Files" (page 14).

Configuring AutoVue Server

Configuring the AutoVue Server for the ISDK Web Service client project follows the same steps as "Configuring AutoVue Server" (page 14).

Configuring log4j.properties for Debugging

Configuring the log4j.properties for the ISDK Web Service client project debugging follows the same steps as "Configuring log4j.properties for Debugging" (page 14).

Configuring SOAP Handler

- 1 Locate and uncomment the following block in web.xml and update the param-value for param wsclient.WSHandler.

```
<!-- the SOAP handler class must extend com.cimmetry.vuelink.wsclient.backend.WSHandler -->
<init-param>
    <param-name>wsclient.WSHandler</param-name>
    <param-value>com.cimmetry.vuelink.wsclient.backend.UserNameTokenHandler</param-value>
</init-param>
```

- 2 Replace the param-value for wsclient.WSHandler with your desired handler.

Here is a list of handlers delivered with Filesys Sample inside the com.cimmetry.vuelink.wsclient.backend package.

Handler Name	Usage
WSHandler	No security implementation.
HTTPBasicHandler	HTTP basic authentication.
UserNameTokenHandler	Generic username token profile security.
WeblogicUserNameTokenHandler	Username token profile security for WebLogic. Use this one if the generic UserNameTokenHandler does not work on Oracle WebLogic Server. You need to rename the source code named "WeblogicUserNameTokenHandler.java.excluded" to "WeblogicUserNameTokenHandler.java" and add "weblogic.jar" to the project's class path.

Defining Location of Blueprint WSDL



Locate the following block in web.xml and update the <param-value> for param WSDL.

```
<!-- Define the location of Blueprint WSDL -->
<init-param>
    <param-name>WSDL</param-name>
    <param-value>... </param-value>
</init-param>
<load-on-startup>1</load-on-startup>
</servlet>
```

A sample param-value for WSDL is:

`http://mymachine:7001/WSClient/Blueprint?wsdl`

Running Filesys Project

- 1 Run the AutoVue Server.
- 2 Deploy project and start WebLogic server on Eclipse:
 - a. Go to the Servers view by clicking **Servers** .
 - b. Right-click on the Oracle WebLogic Server and then click **Add and Remove**. In the Add and Remove dialog, select your project from the left panel, click **Add** to add the project to the right panel, and then click **Finish** to exit.
 - c. Click  to start the server.
- 3 Start WebLogic server and deploy project on JDeveloper:
 - a. From Application Server Navigator, right-click IntegratedWebLogicServer and click **Start Server Instance** to start WebLogic Server.
 - b. Right-click the project, click **Deploy** and click your project's name. The Deploy dialog appears.
 - c. In the Deploy dialog, select **Deploy on Application Server**, click **Next**, then select **IntegratedWeblogicServer**.
 - d. Accept the default setting and click **Next**.
 - e. At the last page, click **Finish**.
 - f. Check the server's Deployment log and get the URL context root for testing the application. For example, `http://10.10.1.1:7101/ISDKSamples-filesys-context-root`.
- 4 Launch a Web browser and enter the URL address `http://host:port/context` to launch the home page for ISDK Demo. For example, the URLs can be `http://localhost:7001/filesys` for Eclipse and `http://10.10.1.1:7101/ISDKSamples-filesys-context-root` for JDeveloper.
- 5 If you run into an issue when launching the project, verify that the FilesysVueLink and VueServlet servlets are running properly using the following URLs:
 - `http://<host:port>/context/servlet/FilesysVuelink`
 - `http://<host:port>/context/servlet/VueServlet`

Replace the `<host:port>` using your own host name, WebLogic server's port.

Replace `context` with the context for Filesys project on IDE.

If VueLink and VueServlet are running properly, the URLs load and displays their respective version and build information, and in the case of the VueServlet, whether the connection state is OK. If you do not get a successful response, perform the following verifications:

- Verify that the AutoVue server is running.
- Verify that your project is installed deployed correctly.
- Verify that web.xml is configured properly.
- Verify that your application server is running and functioning properly.

Feedback

Oracle products are designed according to your needs. We would appreciate your feedback, comments or suggestions. If at any time you have questions or concerns regarding AutoVue Integration SDK, call or email us. Your input is an important part of the information used for revision.

General Inquiries

Telephone: +1.514.905.8400 or +1.800.363.5805

E-mail: autovuesales_ww@oracle.com

Web Site: <http://www.oracle.com/us/products/applications/autovue/index.html>

Sales Inquiries

Telephone: +1.514.905.8400 or +1.800.363.5805

E-mail: autovuesales_ww@oracle.com

Customer Support

Web Site: <http://www.oracle.com/support/index.html>

