

VueLink for Open Text

***An Integration between AutoVue
Client-Server Edition and
Open Text Livelink***

***System Administrator
Manual***

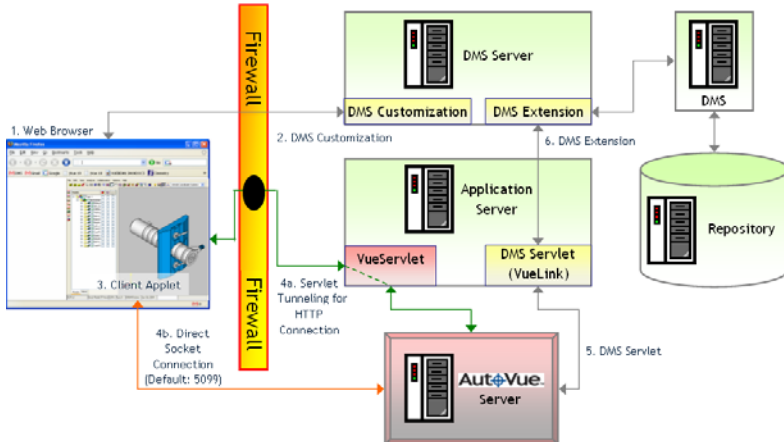
CONTENTS

INTRODUCTION	1
SYSTEM REQUIREMENTS	2
AUTOMATIC INSTALLATION	3
Running VueLink's InstallShield for Open Text	3
MANUAL INSTALLATION	18
DMS Customization	18
DMS Extension	18
VueLink Servlet	19
Configuration	19
Verification	19
To Manually Install VueLink for Livelink v9.5.0/v9.5.0Sp1	19
DMS Customization	19
DMS Extension	21
VueLink Servlet	22
Configuration	22
Mandatory	22
Optional	22
Installing VueLink for OpenText Module	23
Installing VueLink Servlet for the Servlet Engine	24
To deploy VueLink to a J2EE-enabled Application Server	24
To install VueLink for Jetty	24
CONFIGURING VUELINK	
SERVER-SIDE PARAMETERS	27
JVUESERVER Applet Parameter	27
DMS Applet Parameter	28
Verbose Applet Parameter	28
CodeBase Applet Parameter	28
LLServlet	29
Specifying the URL to LLServlet	29
Servlet IP	29
AutoVue Server Socket Port and Application Server ports	29
CONFIGURING VUELINK	
MIME TYPES ASSOCIATION	30
Settings for MIME Types Association	30

Register/Unregistered AutoVue Functionality Based on Document MIME Types	31
OTHER CONFIGURATION	32
Customizing Livelink Workspace for AutoVue	32
Setting up VueLink for Open Text for Single Sign-On (LDAP or NTLM)	34
Setting up VueLink for OpenText for Single Sign-On	34
Setting Parameters in vuelink.properties	35
Associating Markups with Document ID	36
Configuring for Xpression	36
Changing Access Level to Create Folders	36
Setting permissions for the Annotations Folder	37
Specifying Folder Names for Annotations/Viewables/Conversions Folders	39
Completing the Configuration for XRefs	39
For each Reference Document added to the Database	40
Printing DMS Attributes	41
Configuring for Remote Cache Servers	42
Configuring Remote Cache Servers for Optimal Performance	43
Enabling HTTPS/SSL	47
Updating java.security to Enable HTTPS/SSL	47
Enabling HTTPS/SSL Support with Application Server/Servlet Engine	47
VERIFICATION	49
Verifying that VueLink Servlet is Running Properly	49
Running VueLink Servlet in Debug Mode	50
FEEDBACK	51
General Inquiries	51
Sales Inquiries	51
Customer Support	51

Introduction

The VueLink servlet allows AutoVue Server to communicate with Open Text's Livelink Document Management System (DMS) using standard HTTP protocol. The following diagram illustrates a typical configuration of how AutoVue integrates with Livelink.



- 1 Log on to the DMS through a Web browser such as Microsoft Internet Explorer.
- 2 With Customization in place, clicking a Functions link opens a menu that lets you select **AutoVue**.
- 3 Clicking **AutoVue** launches the AutoVue applet and you can view the file inside the web browser window.
- 4 Depending on the AutoVue configuration, the AutoVue applet communicates with the AutoVue Server either through servlet tunneling for HTTP connection or through direct socket connection.
- 5 AutoVue Server then communicates with the VueLink servlet using standard HTTP connection.
- 6 With DMS Extension installed on the server, the VueLink servlet can communicate with Livelink to handle any request made by AutoVue Server, like file “fetching”.

To display a composite file (a file with external references), the VueLink servlet retrieves the file along with all its component files from DMS and makes them available to AutoVue Server. Then AutoVue server processes them and AutoVue client displays the composite file. From here you can mark up the file, create new Markups, save Markups into the DMS and open Markups from the DMS.

System Requirements

- AutoVue v19.1 (Note: All service packs of 19.1 are supported.)
- Open Text Livelink Server v9.5.0 or v9.5.0Sp1
- Livelink UNITE & Atlas 9.5.0 or 9.5.0.1 (optional)
- Livelink Explorer Pro v4.4.2 or 4.6.0 (optional)
- Livelink Directory Services v2.3.1 (optional)
- Livelink Remote Cache v3.2 (optional)
- Livelink CAD Manager for AutoCAD v2.0.1 or v3.0 (optional)
- Application Server / Servlet Engine
Cimmetry certifies the following application servers: Jetty, Tomcat 5.0.x,
and WebLogic 8.1 Service Pack 4.

Automatic installation

This document's installation instructions for the VueLink servlet support these Open Text Livelink server installations:

- Livelink v9.5.0
- Livelink v9.5.0Sp1

The following section consists of instructions for installing VueLink for Open Text with the help of the InstallShield Wizard. Instructions for manual installation follows.

Running VueLink's InstallShield for Open Text

The VueLink Installer for Open Text installs the VueLink servlet and copies custom components developed by Cimmetry Systems Corp. to appropriate folders within the Open Text installation.

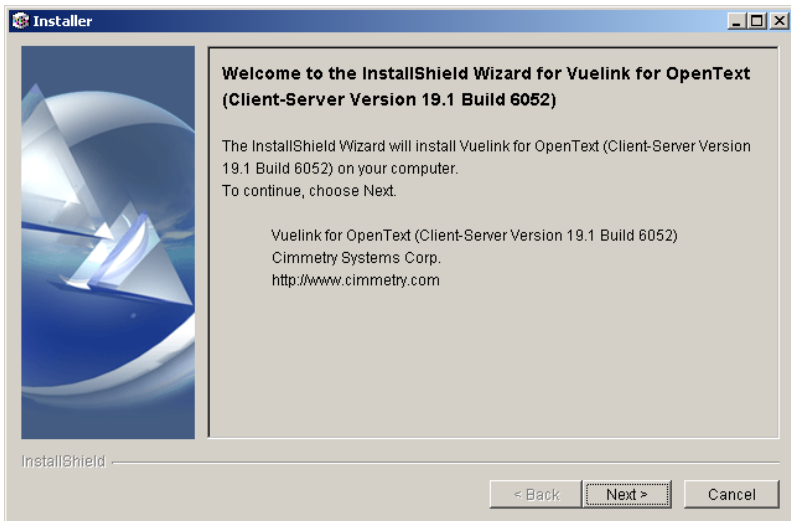
Note:

- After running the installer successfully, the VueLink for OpenText module needs to be deployed from the Open Text Livelink Administrator interface.
- If you have a previous copy of VueLink for Open Text installed, **uninstall** it before proceeding with the new installation.
- If you have a previous copy of **VueLink for OpenText module** installed, **uninstall** it before proceeding with the new installation. Uninstalling the VueLink does **not** uninstall the VueLink for OpenText module. You need to manually uninstall the VueLink for OpenText module.
- If you are installing VueLink on a Solaris machine, you must be logged on as **root** user and have **xterm** access.
- If you are installing on Solaris, run `setup_solarisSparc.bin` found on the distribution CD under **/opentext/vuelink_setup/solaris**.

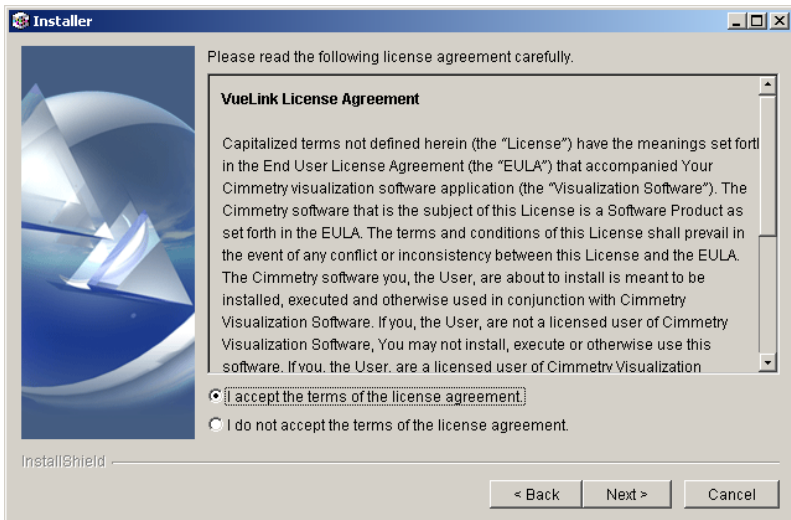
To run VueLink InstallShield on Windows:

- 1 Run **setupwin32.exe** found on the distribution CD under **\AutoVue_Client_Server_Edition\vuelink_setup\win32\opentext\vuelink_setup\win32**.

The installer starts the installation of VueLink for Open Text.

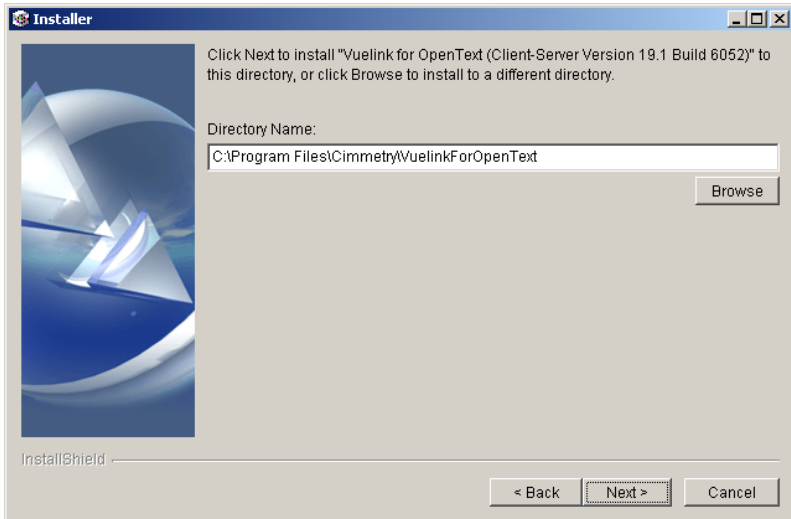


- 2 Read the license agreement and accept or decline the terms. If you decline, the installation aborts.



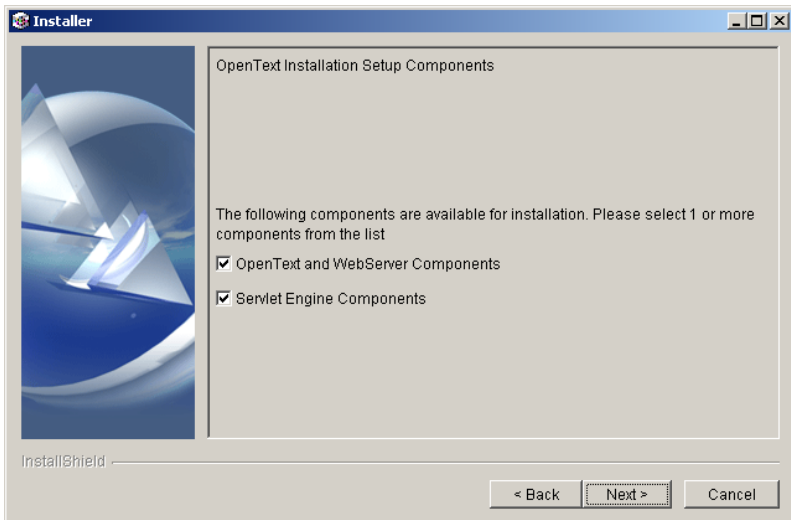
- 3 Enter the directory where you want VueLink for Open Text installed. The setup creates a copy of all files installed or updated as part of the installation in this directory.

Note: Make sure there are no extra spaces in the installation folder name.

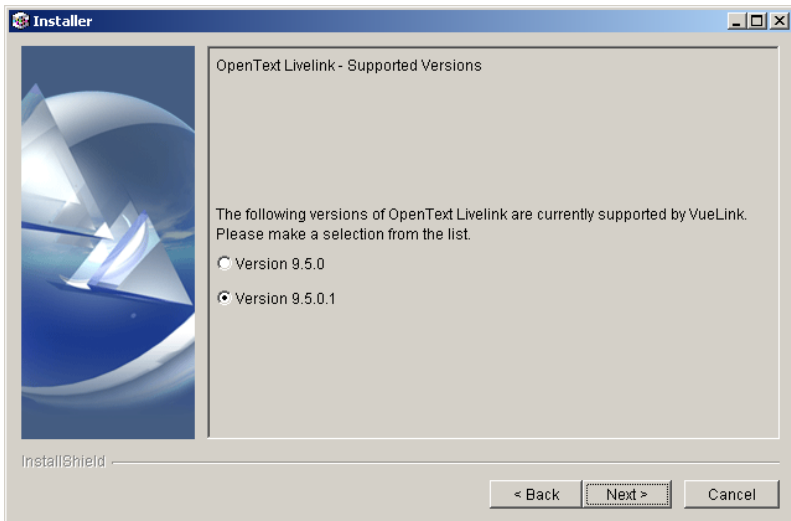


- 4 Choose the components that need to be installed.

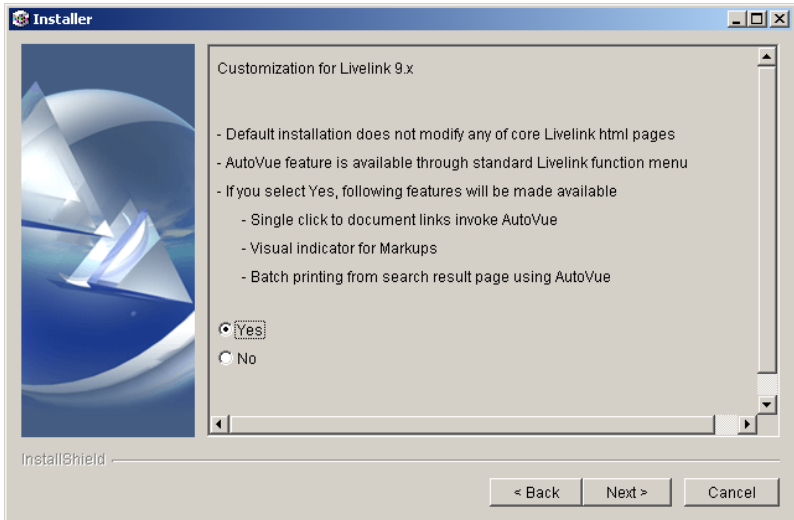
Note: The **Open Text and WebServer Components** option installs the custom module `csijvot.oll` and its HTML resources in addition to web-server components like `.cab` and `.jar` files.



- 5 Select the version of Open Text Livelink installed on the machine.

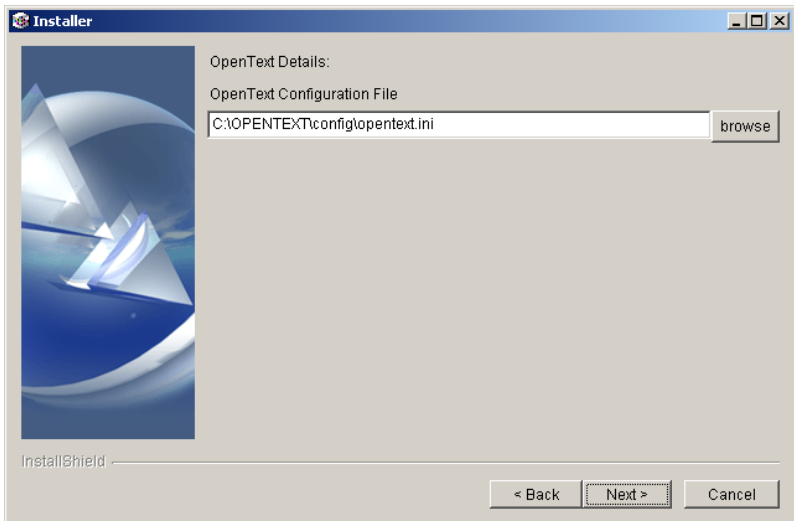


- 6 In the next screen, **Customization for Livelink 9.x**, select **Yes** if you want to update core Livelink HTML pages.
If you select **Yes**, AutoVue will be invoked when you click document links, documents with Markups will display a red bulb, and batch printing using AutoVue will be available from the Search results page.
If you select **No**, the default installation does not modify any of the core Livelink HTML pages and AutoVue is available through the standard Livelink function menu.



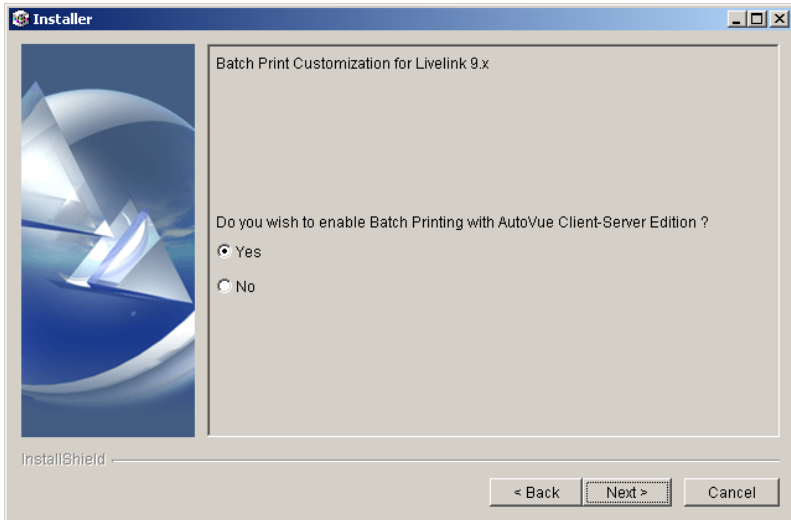
- 7 Locate the Open Text configuration file.

Note: The configuration file often appears by default. It is typically located at <'My Open Text Directory'\>config\opentext.ini.

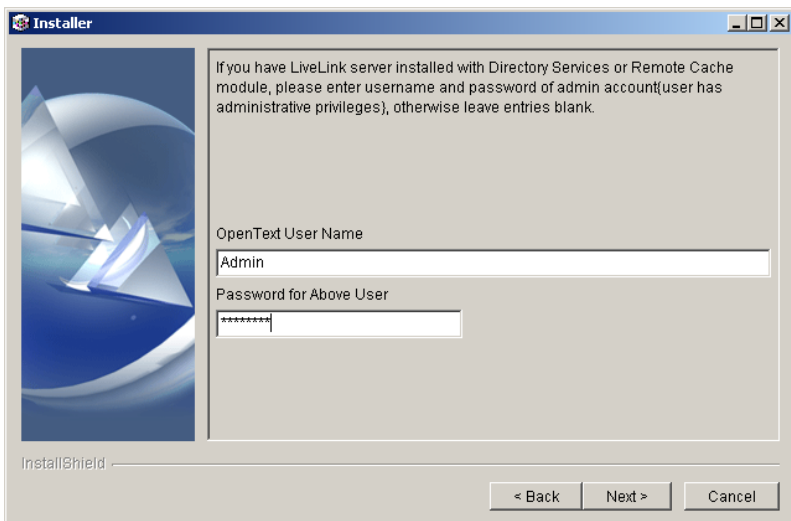


- 8 The installer asks if you want to install Batch Printing. To install Batch Print components, select **Yes**.

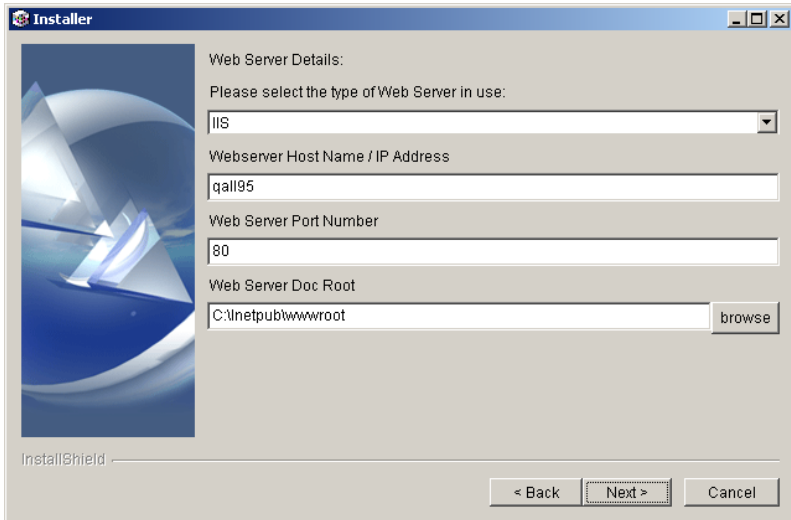
With BatchPrint components, you can use AutoVue to do batch printing from the Livelink search results page.



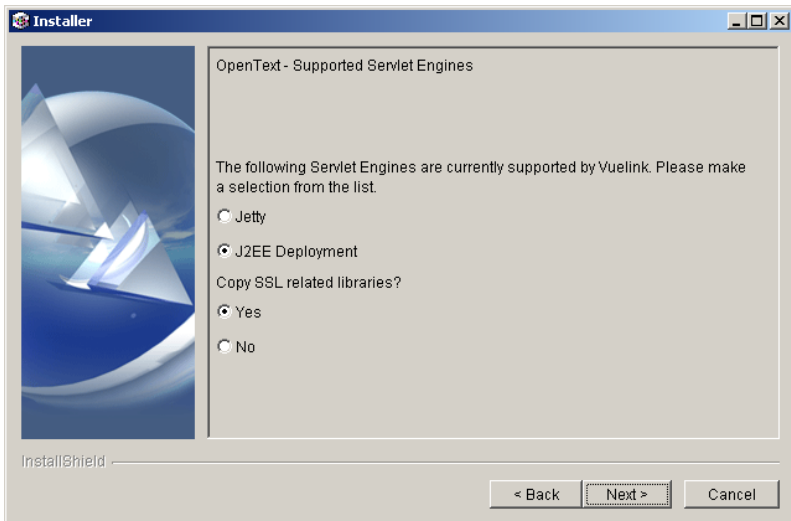
- 9 If you have the LiveLink server installed with Directory Services or the Remote Cache module, enter the user name and password of a Livelink user with administrative privileges. Otherwise, leave the fields blank.



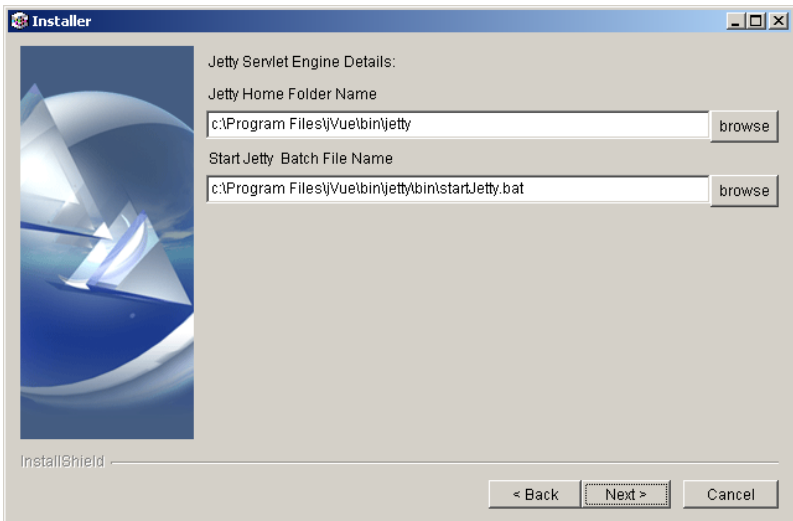
- The installer prompts for Web Server Details. Select the web server in use and enter the Host Machine Name, the Port for the web server and the web server doc root.



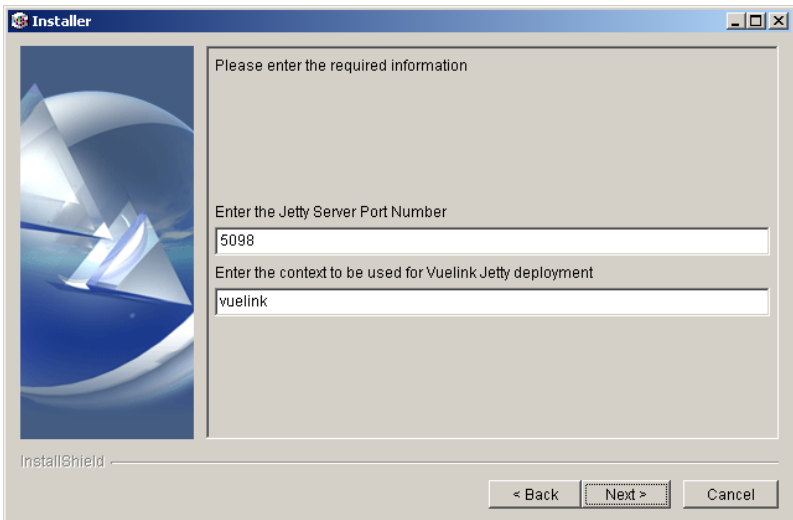
- Select the servlet engine you are using. If your servlet engine is not listed, see [Installing VueLink Servlet for the Servlet Engine \(page 24\)](#). Also specify if you want to copy SSL-related libraries. If you are using JVM 1.4 or higher, you do not need to copy these files.



- 12 If you selected **Jetty** as your servlet engine, enter the folder name of the Jetty installation in the appropriate text field.
- If Jetty was shipped with AutoVue:
- The default folder is **<Install_dir>\bin\jetty**, where **<Install_dir>** is the directory where AutoVue Server is installed. Enter the path and name of the batch file called to invoke Jetty in the appropriate text box.
 - The default Start Jetty Batch File Name is **<Install_dir>\bin\jetty\bin\startJetty.bat**, where **<Install_dir>** is the directory where AutoVue is installed.

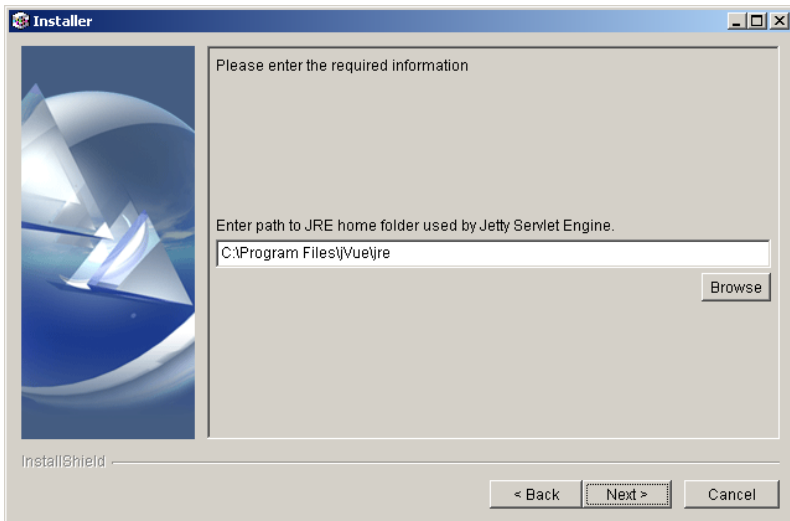


- c. Next, the installer prompts for the port used by Jetty. The default port is **5098**.



- d. Next, the installer asks if AutoVue Server is installed on the same machine. If you replied **Yes**, the installer then prompts you for the path to **jueserver.properties**.

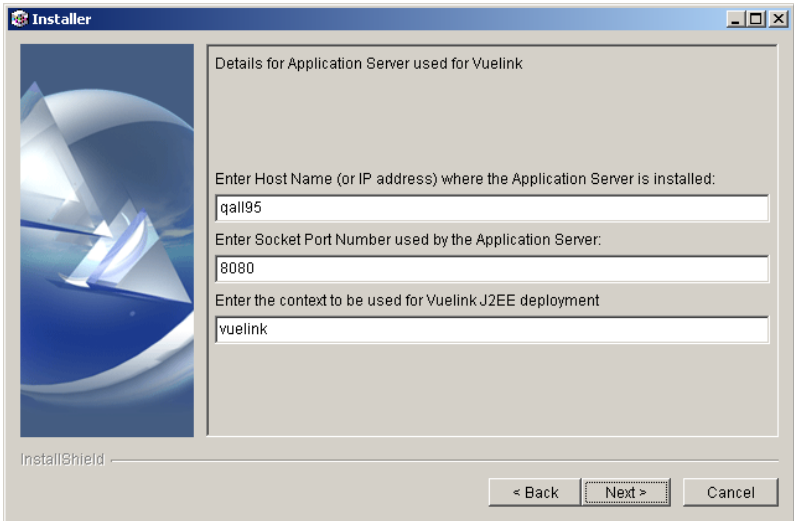
e. If you select Jetty, the installer prompts you for the jre folder Jetty uses.



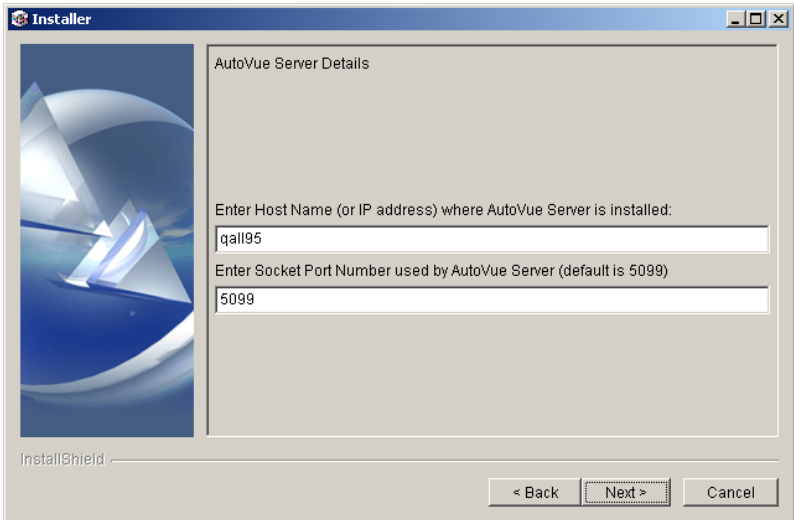
13 If you selected J2EE Deployment, the installer prompts for the Application Server Host name and the port number and the VueLink Context name.

Note:

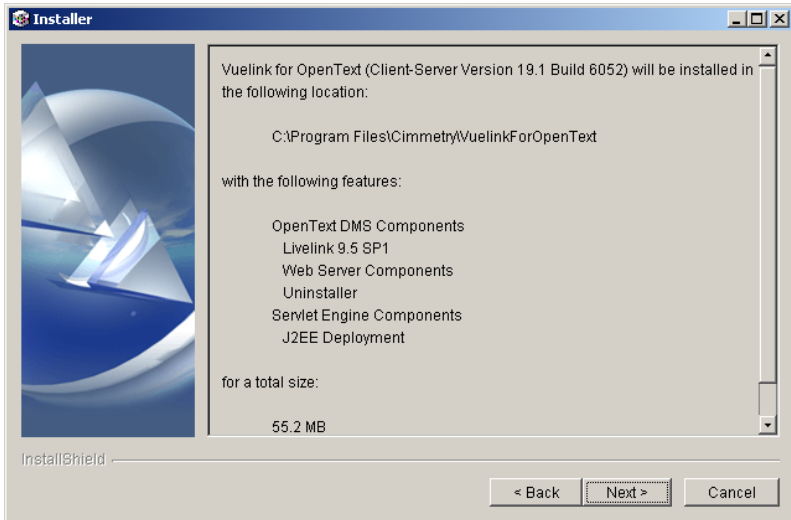
- For J2EE deployments, the installer creates a war file in **<VueLink Installation Folder>/vuelink4j2ee**.
- This war file has to be deployed with your application server. When you deploy **vuelink.war** with your application server, make sure you specify the same application name that was used when installing the VueLink.



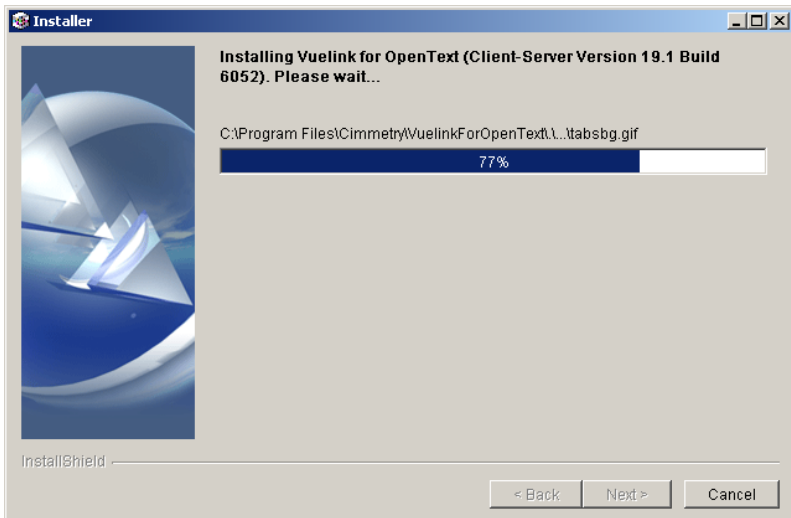
- 14 Enter the AutoVue Server host name/IP address and socket port number in the appropriate text boxes. Otherwise, accept the default values.



- 15 The Installer summarizes the installation options, and a panel shows the information. The following figure shows an example:

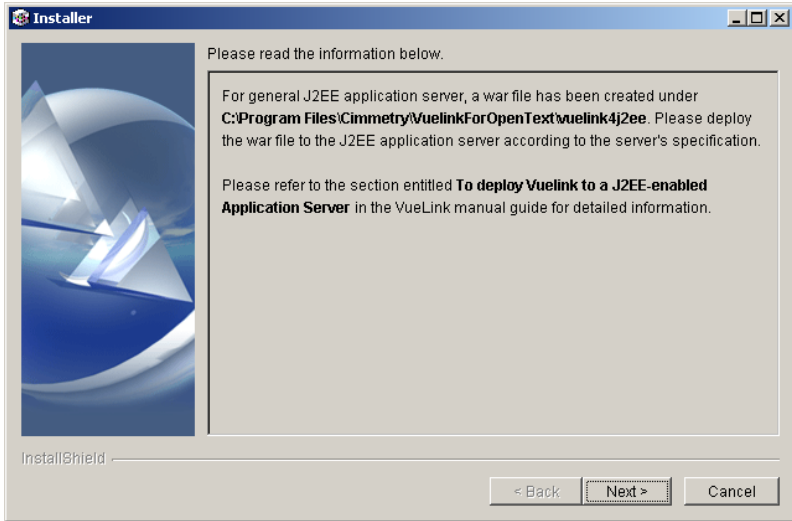


- 16 Click **NEXT**, then wait for the installer to complete.



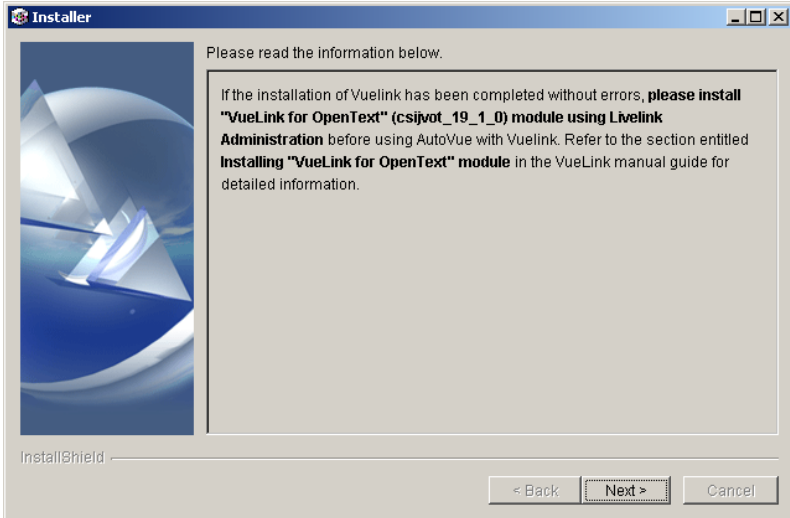
- 17 If you selected the general J2EE application server, a war file has been created under:
C:\Program Files\Cimmety\VuelinkForOpenText\vuelink4j2ee. Deploy the

war file to the J2EE application server according to the server's specification. See "To deploy VueLink to a J2EE-enabled Application Server" on page 24 for more information.

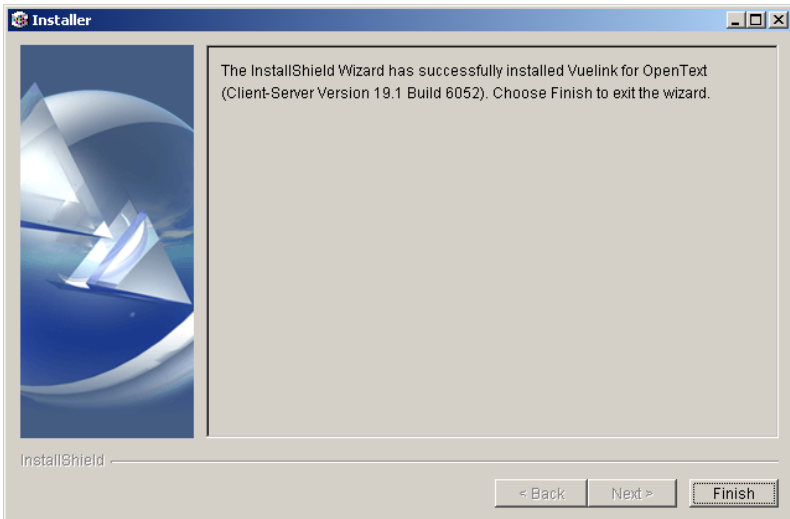


- 18 If the installation of VueLink is completed without errors, install "VueLink for OpenText" (csijvot_19_1_0) module using Livelink Administration

before using AutoVue with VueLink. See “Installing VueLink for OpenText Module” on page 23 for more information.



19 Click **Next** and then **Finish** to exit the installer.



20 Now you must register the VueLink for OpenText module with Livelink. Go to the section called [Installing VueLink for OpenText Module](#).

- 21 For J2EE Deployment, you must deploy **vumlink.war** with your application server. See the section [“To deploy VueLink to a J2EE-enabled Application Server” on page 24.](#)
- 22 If you have LDAP/NTLM authentication, refer to the section [“Setting up VueLink for Open Text for Single Sign-On \(LDAP or NTLM\)” on page 34.](#)
- 23 To configure VueLink server-side parameters such as AutoVue Server details, Application Server port, and LLServlet URL, see section [“Configuring VueLink Server-Side Parameters” on page 27.](#)
- 24 To configure XRefs, access levels for folders, and for other configuration instructions, see section [“Other Configuration” on page 32.](#)
- 25 To associate AutoVue with MIME types, see section [“Configuring VueLink MIME Types Association” on page 30.](#)

Manual Installation

The manual installation is divided into these sections: DMS customization, DMS extension, VueLink servlet, Configuration and Verification.

This document's installation instructions for VueLink servlet support these Open Text Livelink server installations:

- Livelink v9.5.0
- Livelink v9.5.0Sp1

DMS Customization

Follow the steps in this section to customize your DMS to:

- Update Core Livelink HTML pages.
- Invoke AutoVue when you click document links.
- Display a red light bulb beside documents with Markups in the Livelink document/version browse pages.
- Make batch printing using AutoVue available from the Search results page.

To activate these features, copy the customized files in the **dms_customization** folder on the distribution CD. Click a document link to launch AutoVue client from within DMS pages. VueLink enables the menu item **AutoVue** in the **Function** menu for aliases and generations. You can view documents marked as aliases or generations directly with AutoVue.

Skip the DMS Customization section if you do not want to modify any of the core Livelink HTML pages and you want AutoVue to be available through the standard Livelink function menu.

DMS Extension

Follow these steps to add functionality to your DMS. With DMS Extension installed on the server machine, the VueLink servlet can communicate with Open Text Livelink to handle requests coming from AutoVue Server. VueLink Servlet will call upon the files you copy from the **dms_extension** folder on the distribution CD to allow you to run various queries inside DMS.

VueLink Servlet

In this section you will install VueLink servlet into the servlet engine. The servlet engine is what loads VueLink Servlet and makes it run. VueLink servlet files are located in the **vueLink_servlet** folder on the distribution CD.

Configuration

Set up VueLink parameters.

Verification

Follow steps to verify that VueLink servlet is functioning properly.

Before integrating AutoVue with Open Text, ensure that Open Text Livelink and AutoVue Server are installed and configured according to the manufacturer's instructions. Verify that they are operating correctly. Try testing Livelink and AutoVue separately to verify that the installation was successful and that all functions are available and produce the expected results.

To Manually Install VueLink for Livelink v9.5.0/v9.5.0Sp1

DMS Customization

Note: Follow the DMS customization steps if you want the following:

- Core Livelink HTML pages to be updated.
- AutoVue to be invoked when you click document links.
- Documents with Markups to display a red bulb.
- Batch printing using AutoVue to be available from the Search results page.

Skip the DMS Customization section if you do not want to modify any of the core Livelink HTML pages and you want AutoVue to be available through the standard Livelink function menu.

Important: This section uses the following naming conventions:

<install_source>	Folder \AutoVue_Client_Server_Edition\dms_customization\opentext9.5.0\ on the distribution CD for Livelink 9.5.0. Folder \AutoVue_Client_Server_Edition\dms_customization\opentext9.5.1\ on the distribution CD for Livelink 9.5.0Sp1. Note: <install_source> varies depending on the version of Livelink.
<opentext_root>	Root folder of the livelink installation.
x_x_x	Version number of a module. When installing VueLink, you are required to replace x_x_x with the actual version number of the module.

- 1 Verify that Open Text Livelink server is installed properly.
- 2 Back up your existing copy of **browseproject.html** and copy the file **browseproject.html** from <install_source>\module\project_x_x_x\html to <opentext_root>\module\project_x_x_x\html.
- 3 Back up your existing copy of **cdview.html** and copy the file **cdview.html** from <install_source>\module\webdoc_x_x_x\html to <opentext_root>\module\webdoc_x_x_x\html.
- 4 Back up your existing copy of **browseview.html** and copy the file **browseview.html** from <install_source>\module\webnode_x_x_x\html to <opentext_root>\module\webnode_x_x_x\html.
- 5 Back up your existing copy of **infoversions.html** and copy the file **infoversions.html** from <install_source>\module\webnode_x_x_x\html to <opentext_root>\module\webnode_x_x_x\html.
- 6 Copy the file **browsecatalogs.html** from <install_source>\module\webnode_x_x_x\html to <opentext_root>\module\webnode_x_x_x\html.
- 7 Copy the file **icon_jvue.gif** from <install_source>\support\webdoc to <opentext_root>\support\webdoc.
- 8 Copy the file **csired.gif** from <install_source>\support\webnode to <opentext_root>\support\webnode.

- 9 Back up your existing copy of **wfbrowseview.html** and copy the file **wfbrowseview.html** from `<install_source>\module\webwork_x_x_x\html` to `<opentext_root>\module\webwork_x_x_x\html`.
- 10 Back up your existing copy of **favoritesview.html** and copy the file **favoritesview.html** from `<install_source>\module\home_x_x_x\html` to `<opentext_root>\module\home_x_x_x\html`.
- 11 Follow the next steps if you want to enable Batch Printing.
- 12 Back up your existing copy of **bottombuttons.html** and copy the file **bottombuttons.html** from `<install_source>\module\webbroker_x_x_x\html` to `<opentext_root>\module\webbroker_x_x_x\html`.
- 13 Back up your existing copy of **countandbuttons.html** and copy the file **countandbuttons.html** from `<install_source>\module\webbroker_x_x_x\html` to `<opentext_root>\module\webbroker_x_x_x\html`.
- 14 Copy the file **csibatchprint.html** from `<install_source>\module\webbroker_x_x_x\html` to `<opentext_root>\module\webbroker_x_x_x\html`.
- 15 Back up your existing copy of **narrative.html** and copy the file **narrative.html** from `<install_source>\module\webbroker_x_x_x\html` to `<opentext_root>\module\webbroker_x_x_x\html`.
- 16 Restart Livelink Services.

DMS Extension

Note: For any DMS Extension files you copy, make sure you are copying from the folder named according to your system's platform, windows or solaris.

Follow these steps to install the VueLink for OpenText module into Livelink:

- 1 Copy the folder **csijvot_<release>_<build>** found on the distribution CD under the directory `\AutoVue_Client_Server_Edition\dms_extension\opentext9.5.x_<OS>\module` to `<opentext_root>\staging` where `<release>` is the release of AutoVue and `<build>` is the build number of AutoVue, `<opentext_root>` is the root folder where Livelink is installed and `<OS>` is the platform where Livelink Server is installed (winnt or solaris).
- 2 To install the VueLink for OpenText module from the Livelink Administration page see [Installing VueLink for OpenText Module](#).

VueLink Servlet

VueLink servlet files are found on the distribution CD under the directory **vueLink_servlet**. The VueLink servlet cannot run on its own; it requires a servlet engine which is responsible for loading the servlet and making it run. Instructions for installing VueLink into the servlet engine vary depending on the server used. However, the basics of the installation procedure are as follows:

- 1 Copy all the files found under the directory **vueLink_servlet** on the distribution CD to your servlet engine's "Servlet" directory.
- 2 Add these files to your servlet engine's class path: `vueLink.jar`, `vueServlet.jar`, `jaxp-api.jar`, `jcrt.jar`, `jnet.jar`, `jsse.jar`, `commons-httpclient.jar`, `commons-logging.jar`, `dom.jar`, `jce1_2_2.jar`, `local_policy.jar`, `sax.jar`, `sunjce_provider.jar`, `US_export_policy.jar`.
- 3 Copy the file **vueLink.properties** found in **vueLink_servlet** on the distribution CD to any folder on your application server/servlet engine machine.
- 4 Create an alias for **com.cimmetry.vueLink.opentext.DMS**.
- 5 Set init parameter properties to `<path>/vueLink.properties` where `<path>` is the folder where you copied **vueLink.properties**.
Note: If you want debug messages to appear in the servlet engine's console or log file, you must set the init parameter **Verbose** to **1**.
- 6 Restart the servlet engine for the changes to take effect.
For installing VueLink, Jetty (AutoVue's internal servlet engine) and a J2EE-enabled Application Server see [Installing VueLink Servlet for the Servlet Engine](#).

Configuration

You must complete the steps outlined under the following sections:

Mandatory

- [Configuring VueLink Server-Side Parameters](#)
- [Setting AutoVue applet parameters inside csibatchprint.html](#)

Optional

- [Customizing Livelink workspace for AutoVue](#)
- [Configuring VueLink for Single Sign-On \(for LDAP or NTLM\)](#)
- [Setting Servlet IP parameters](#)

- [Changing access level to create folders](#)
- [Setting Parameters in vuelink.properties](#)
- [Configuring for XPression](#)
- [Setting permissions for Annotations](#)
- [Completing the Configuration for XRefs](#)
- [Printing DMS Attributes](#)
- [Configuring for Remote Cache Servers](#)
- [Enabling HTTPS/SSL](#)
- [Configuring with LLServlet](#)

Installing VueLink for OpenText Module

In order for VueLink Servlet to work properly, you must install **VueLink for OpenText** module into Livelink. To do so, follow these steps:

- 1 Log on to Livelink Administration.
- 2 Scroll down the page until you see the heading **Module Administration**.
- 3 Click the **Install Modules** link under this heading.
- 4 Click **NEXT**.
- 5 Scroll down the page until you see the heading **Installable Modules**.
- 6 Select the module **VueLink for OpenText** from the list.
- 7 Click **INSTALL**.
Livelink Administrator installs the module.
- 8 Enter Admin **User**.
- 9 Enter the password for Admin User and click **LOGIN**.
You are prompted to restart your Livelink server before continuing.
- 10 Restart Livelink Services.
- 11 Click **Continue**.
- 12 You are returned to the Installed Modules list.

The VueLink for OpenText module is now displays under **Installed Modules**. Livelink places the module under the folder `<opentext_root>\module\csijvot_x_x_x\` where `<opentext_root>` is the root folder where Livelink is installed.

Installing VueLink Servlet for the Servlet Engine

In this section we describe installing VueLink servlet to run with several popular servlet engines:

- [J2EE-enabled Application Server](#)
- [Jetty](#)

To deploy VueLink to a J2EE-enabled Application Server

This section provides generic instructions to install VueLink Servlet for a J2EE Application Server.

If you selected **J2EE Deployment** when installing VueLink for OpenText, the installer creates **vueLink.war** in the **<VueLink Install Folder>\vueLink4j2ee**.

This **vueLink.war** should be deployed to your application server. This **vueLink.war** can be deployed to any application server according to its deployment requirements.

Please refer to the administration guide of your J2EE-enabled application server for instructions on how to deploy a war file.

Note: When you deploy **vueLink.war** with your application server, make sure you specify the same application name that was used when installing VueLink.

To install VueLink for Jetty

This section describes the steps necessary to install the VueLink Servlet to run with **Jetty**, the Servlet Engine that is shipped with AutoVue. Jetty is installed as part of AutoVue Server installation. Its default location is **C:\Program Files\j\Vue\bin\jetty** and its default web port is **5098**.

- 1 Copy all the files found under the directory **AutoVue_Client_Server_Edition\vueLink_servlet** on the distribution CD to AutoVue's default bin directory and to **jetty\lib** directory.
Example: C:\Program Files\j\Vue\bin\jetty\lib.
- 2 Follow these steps to register VueLink Servlet with Jetty:
Open the file **jetty.xml** located in the folder **\j\Vue\bin\jetty\etc**.

Add the following lines that appear in **bold**:

```
...
<Call name="addServlet">
<Arg>VueServlet</Arg>
<Arg>/servlet/VueServlet</Arg>
<Arg>com.cimmetry.servlet.VueServlet</Arg>
<Put name="JVueServer">HostName:5099</Put>
<Put name="Verbose">>true</Put>
<Put name="DebugLevel">0</Put>
</Call>

<Call name="addServlet">
<Arg>com.cimmetry.vuelink.opentext.DMS</Arg>
<Arg>/servlet/com.cimmetry.vuelink.opentext.DMS</Arg>
<Arg>com.cimmetry.vuelink.opentext.DMS</Arg>
<Put name="properties">__path__/vuelink.properties</Put>
<Put name="Verbose">0</Put>
</Call>
```

Set **__path__** to the folder where **vuelink.properties** is located.

The VueLink Servlet is now properly installed. Restart AutoVue Server.

By default Jetty is enabled on AutoVue. When you start up AutoVue Server, the servlet engine is also started.

If you want to start up Jetty independently, you must disable the startup command for Jetty.

Open the file **javueserver.properties** in the folder **lvuelbin**.

Comment the following lines:

```
servlet-engine.classpath= ....
servlet-engine.jre= ...
servlet-engine.cmdline= ...
```

Start AutoVue Server. Start Jetty from `<Install_dir>/bin/jetty/bin/startJetty.bat`.

Note:

- Complete steps outlined under [Enabling HTTPS / SSL](#) if you want the VueLink to access documents using HTTPS / SSL connection.
- You may choose any name (other than **com.cimmetry.vuelink.opentext.DMS**) for the Alias assigned to VueLink Servlet. However, any

name you choose must match the name set in the [Configuring VueLink Server-Side Parameters](#) param for the AutoVue client.

- You may set Verbose=1 if you wish to see debug messages displayed by VueLink Servlet.

Configuring VueLink Server-Side Parameters

VueLink for OpenText provides you with an administrative page to configure VueLink parameters. You can access this page from the Livelink Administration page.

- 1 Login to the Livelink Administration page.
- 2 Scroll down to the **Cimmetry VueLink Configuration** section.
- 3 Click the **Configure VueLink Server-Side Parameters** link.

The Configure Server-side Parameters page lets you set the following parameters:

- [JVUESERVER Applet Parameter \(page 27\)](#)
- [DMS Applet Parameter \(page 28\)](#)
- [Verbose Applet Parameter \(page 28\)](#)
- [CodeBase Applet Parameter \(page 28\)](#)
- [LLServlet \(page 29\)](#)
- [Servlet IP \(page 29\)](#)
- [AutoVue Server Socket Port and Application Server ports \(page 29\)](#)

Changes you make to these parameters apply to csijVue.html and csibatchprint.html files.

Note: Make sure you have installed the VueLink for OpenText module in step 1 of Installation, DMS Extension.

Note: You will need to restart the Livelink service once you have completed all of your changes in order for them to take effect.

JVUESERVER Applet Parameter

Specify the connection parameter for the AutoVue Server. This could be a direct socket connection or a tunneling through the VueServlet.

The syntax for the parameter is as follows:

```
socket://<JVUE-SERVER>:5099;http://<APPLICATION-SERVER>:<port>/servlet/VueServlet
```

This specifies a semi-colon separated list of connection methods to use to communicate with the AutoVue Server.

See “AutoVue – Installation and Administration Guide” for more information.

VueServlet Setting `http://<APPLICATION-SERVER>:<port>/servlet/VueServlet` where `<APPLICATION-SERVER>` is the Name of application server/servlet engine plus Port Number running VueServlet tunneling servlet.

Socket Setting `Socket://<JVUE-SERVER>:<Socket Port>` where `<JVUE-SERVER>` is Name of the host machine running AutoVue server. Default `<Socket Port>` is 5099.

DMS Applet Parameter

Specify the URL of the VueLink Servlet.

The syntax of the URL is as follows:

```
http://<APPLICATION-SERVER>:<port>/vuelink/com.cimmetry.vuelink.opentext.DMS
```

Example: `http://jupiter:8989/vuelink/com.cimmetry.vuelink.opentext.DMS`

The URL points to VueLink Servlet `http://<APPLICATION-SERVER>:<port>/Vuelink/com.cimmetry.vuelink.opentext.DMS`, where `<APPLICATION-SERVER>` is the name of the host machine where the application server and/or servlet engine is installed, and where `<port>` is the port number where the Vuelink DMS servlet is running.

Verbose Applet Parameter

Change client-side (AutoVue applet) verbosity to either True or False. If you want debug messages to appear in the servlet engine's console or log file, you must set the parameter to True.

CodeBase Applet Parameter

Specify the codebase for the AutoVue Client. This is the location of `jvue.cab` and `jvue.jar`.

The syntax is: `<WEB-SERVER>/jVue`

This specifies a URL location to `WEB-SERVER` where AutoVue files (`jvue.cab` and `jvue.jar`) are located.

Set `<WEB-SERVER>` to the host name of your Web server.

Example: `qall95sp1vpc/jVue`

LLServlet

LLServlet is a Java servlet that runs on an application server. You can choose to use it if you want to provide external access to Livelink without having to run the Livelink CGI outside your firewall. When using Single Sign on (NTLM/LDAP), specify the URL pointing to the Livelink J2EE Web application.

Specifying the URL to LLServlet

The syntax of the URL is: `http://<application server:port>/livelink/livelink.exe`.

Note: UNIX versions omit the `.exe`.

Example: `http://host_name:port/livelink/livelink.exe`

Refer to Livelink documentation for more information on configuring LLServlet or on configuring Livelink as a web application.

Servlet IP

The Servlet IP points to the IP Address of the machine running VueLink Servlet.

AutoVue Server Socket Port and Application Server ports

Specify the ports for AutoVue Server and the Application Server.

For example, the Application Server Port at RemoteCache Host can be 5098, and the AutoVue Server Socket Port at RemoteCache Host can be 5099.

Note: The ports you specify here are for the primary server and for remote servers (if you are using Remote Cache). The assumption is that the ports are the same for AutoVue/VueLink on all remote servers and the primary server. For more information regarding how VueLink works in a Remote Cache Configuration, see [“Configuring for Remote Cache Servers” on page 42](#).

Configuring VueLink MIME Types Association

VueLink for OpenText provides you with an administrative page to associate MIME-types with VueLink. You can access this page from the Livelink Administration page.

- 1 Login to the Livelink Administration page.
- 2 Scroll down to the **Cimmetry VueLink Configuration** section.
- 3 Click the **Configure VueLink MIME Types Association** link.

Note: You will need to restart Livelink service once you have completed all of your changes in order for them to take effect.

Settings for MIME Types Association

Using the Configure VueLink MIME-Types page, you can choose to associate AutoVue with MIME types. When you select Enable, the AutoVue menu item in the functions menu appears only for MIME types that are registered with AutoVue. When you select Disable, the AutoVue menu item in the functions menu appears independently of any MIME type.

You can also choose to show the AutoVue link when the MIME type is not registered in Livelink. When you select Enable, the AutoVue menu item appears in the functions menu for any MIME type that is not registered in Livelink. When you select Disable, the AutoVue menu item will not appear in the functions menu unless the MIME type is explicitly registered with AutoVue. See the following section about registering and unregistering MIME types.

Once you have chosen to enable or disable these options, you can click the Submit button.

Register/Unregistered AutoVue Functionality Based on Document MIME Types

When a MIME type does not appear in the unregistered list, you can enter it in the text field and then click **Append MIME to Unregistered List**.

The Configure VueLink MIME-Types page displays unregistered and registered MIME types in two separate lists. To register MIME types that are in the unregistered list, select the unregistered MIME type and click the **Register MIME-Types** button. Similarly, to unregister MIME types that are in the registered list, select the registered MIME type and click the **Unregister MIME-Types** button. Ctrl-click to select more than one MIME type. Shift-click to select a range in the list.

Other Configuration

- [Customizing Livelink Workspace for AutoVue \(page 32\)](#)
- [Setting up VueLink for Open Text for Single Sign-On \(LDAP or NTLM\) \(page 34\)](#)
- [Setting Parameters in vuelink.properties \(page 35\)](#)
- [Associating Markups with Document ID \(page 36\)](#)
- [Configuring for Xpression \(page 36\)](#)
- [Changing Access Level to Create Folders \(page 36\)](#)
- [Setting permissions for the Annotations Folder \(page 37\)](#)
- [Specifying Folder Names for Annotations/Viewables/Conversions Folders \(page 39\)](#)
- [Completing the Configuration for XRefs \(page 39\)](#)
- [Printing DMS Attributes \(page 41\)](#)
- [Configuring for Remote Cache Servers \(page 42\)](#)
- [Enabling HTTPS/SSL \(page 47\)](#)

Note: See “[Configuring VueLink Server-Side Parameters](#)” on page 27 to make changes to csibatchprint.html file.

Customizing Livelink Workspace for AutoVue

When installing the VueLink for OpenText, you have the option of customizing Livelink’s core html pages. If you selected this option when installing the VueLink, you can further customize Livelink workspace.

The following table lists INI options that are set in **csijvot.ini**:

INI Option	Default	Description
The following options go under the section heading [Markups] in csijvot.ini .		
EnableMarkupIndicator	True	Set to true if you wish a red light bulb (markup indicator) to appear next to a file if the file has markups.

INI Option	Default	Description
EnableMarkupIndicatorLastVersionOnly	True	Set to false if you wish to see markup indicator for files whose previous versions have markups, even if the current version does not have markups.
UseNodeIDOnly	True	Set to true if you wish to associate markups with Document ID. Set to false if you wish to associate markups with Document name. If markups are associated with ID, if a document is renamed or moved, the links between the base document and its markups are preserved.
EnableMarkupPermissionBaseFile	False	Set to true to set Markup file permissions are the same as base file permissions.
The following options go under the section heading [Options] .		
EnableAutoVueLink	True	Set to true if you wish to launch AutoVue by clicking the file link.
ApplyRemoteCacheOptimalConfigSetting	False	When using remote sites, set to true if you wish to force optimal settings. See Configuring for Remote Cache servers

Note:

- To modify any of these settings, update csijvot.ini located in <Open Text root>\module\csijvot_x_x_x.
- Restart Livelink service.

Setting up VueLink for Open Text for Single Sign-On (LDAP or NTLM)

When Livelink is configured with Single-Sign On, extra configuration has to be done for the VueLink/AutoVue Server.

When a user tries to access Livelink, the Web/HTTP server first validates the user against the LDAP database and then passes user information to Livelink. Livelink then checks the LDAP database to verify that the user exists before permitting logon.

When a user selects a file to view with AutoVue, AutoVue prompts for authorization. The authorization request appears because the Web server requires NTLM/LDAP authentication and the AutoVue server has not been authenticated. To work around this, VueLink for Open Text communicates with Livelink via the LLServlet using existing user credentials (LLCookies). Users are not prompted to enter logon information again and the AutoVue client displays the file.

The following section, [Setting up VueLink for Open Text for Single Sign-On](#) describes how to update csjVue.html to use LLServlet.

Note:

- LLServlet must be configured to accept anonymous connections.
- LLServlet is used by the VueLink only for authentication.
- These points also apply to NTLM authentication.

Setting up VueLink for OpenText for Single Sign-On

- 1 Install LLServlet.
- 2 In a text editor such as Notepad, open the file **csjvue.html** found in the directory **<opentext_root>/module/csjvot_x_x_x/html**, where **<opentext_root>** is the root folder where Livelink is installed.
- 3 Under "jVue Configuration Section", set sLLServlet to the URL pointing to the LLServlet.
Example: ;String sLLServlet="http://qall92:7001/LLServlet/livelink.exe"
- 4 In a text editor, open the file **resultlist.html** found in the directory **<opentext_root>/module/websbroker_x_x_x/html**.
- 5 Set sLLServlet to the URL pointing to the LLServlet.

- 6 In a text editor, open the file **narrative.html** found in the directory **<opentext_root>/module/websbroker_x_x_x/html**.
- 7 Set **sLLServlet** to the URL pointing to the **LLServlet**.
- 8 Restart Livelink Services for the changes to take effect.
- 9 Edit file **vueLink.properties** in a text editor such as Notepad. This file is located in the **servlets** directory of your application server/servlet engine. Set **DSUserName** to a valid Livelink user name that has system administrator privileges.
Set **DSUserPassword** to the password for the Livelink user specified in the previous step.
Note: This user information is used by the VueLink to authenticate with Livelink. Any other operations within Livelink are based on permissions of the user who originally logged into Livelink.

Setting Parameters in **vueLink.properties**

- 1 Open the file **vueLink.properties** that you copied to the **Servlets** folder of your **Application Server/Servlet Engine**.
- 2 Modify the following settings if needed.

Setting	Syntax	Description
DSUserName		Set DSUserName to a valid livelink user name with System Admin privileges. This user information is used by the VueLink to authenticate with Livelink. Any other operation within Livelink is based on the permission sets of the original user that logged into Livelink. This value is encrypted.
DSUserPassword		Set DSUserPassword to the password of the user specified in DSUserName. This value is encrypted.

Associating Markups with Document ID

As of release 18, markups are associated with Document ID instead of Document name. An advantage of this is that if a document is moved or renamed, the links between the base document and its markups remain intact.

If you wish to change this behavior and associate markups with document name, edit the following INI option in **csijvot.ini**.

Setting	Default	Description
UseNodeIDOnly	true	Set to true if you wish to associate markups with Document ID. Set to false if you wish to associate markups with Document name.

Configuring for Xpression

If Xpression is installed, additional configuration is required for AutoVue to open files from Livelink. Take the following steps to configure Livelink:

- 1 Log on to Livelink.
- 2 Go to the Xpression Forbidden page.
Example: <http://myserver/livelink/livelink.exe?func=xpression.forbidden>
- 3 Enter 'ObjAction' under **Name** column and 'csijVue' under **Value** column.
- 4 Click **Add**.

These steps let you disable Xpression formatting to the html page that contains AutoVue client.

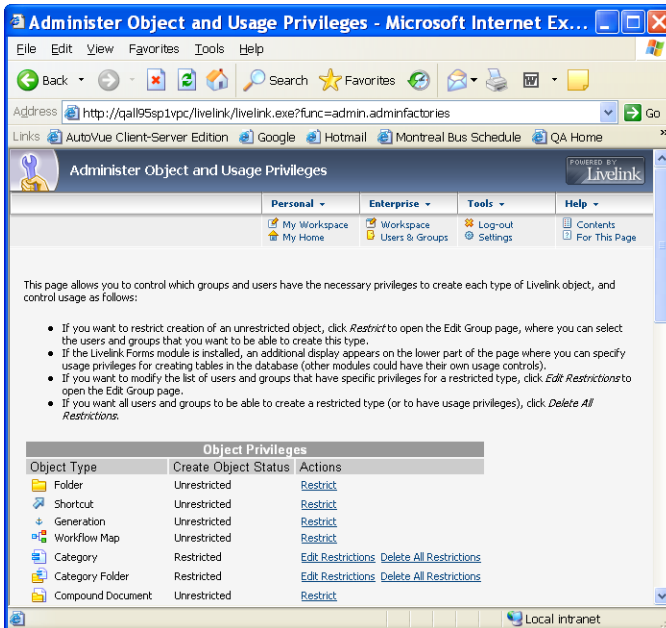
Changing Access Level to Create Folders

Users need to be given permission to create folders to be able to save folders under the Annotation folder.

- 1 Log on to Livelink Administrator.
- 2 Scroll down the page until you see the heading **System Administration**.
- 3 Click Administer Object and Usage Privileges.

- 4 Enter Admin **User**.
- 5 Enter your password and click **LOGIN**.
- 6 Next to **Folder** Object Type, you may remove any restrictions for creating folders.

For example, Create Object Status should display “Unrestricted” to give all users the privilege to create folders. Otherwise, you can edit the restriction and allow only certain users or groups to have the privilege to create folders.



Setting permissions for the Annotations Folder

VueLink automatically creates a folder called Annotations at the Enterprise Workspace, Personal Workspace, Project Root, and Workflow Root levels.

Every time a Markup is created, VueLink recursively searches for the Annotations folder within the workspace that contains the relevant document. If the search fails, VueLink attempts to create a folder called Annotations at the workspace level. The System Administrator can set the ACL (Access Control List) for this folder from within the **csiSaveDoc.html** file.

Note: Make sure that users who need to mark up drawings have full access to create items and folders inside the Annotations folder. For more information, see [Changing Access Level to Create Folders](#).

- 1 In a text editor such as Notepad, open the file `csisavedoc.html` found in the directory `<opentext_root>\module\csijvot_x_x_x\html\csiSaveDoc.html`, where `<opentext_root>` is the root folder where Livelink is installed. For full permission, make sure there is no comment at the beginning of the following line:

```
; newEntry.pWorldPerm = DAPI.FULL_PERM_MASK
```

This line grants **All Users** of Livelink **Complete Access** (Read/Write/Modify) to the Annotations folder.

To grant permission to all users in the group, remove the comment from the following line:

```
//newEntry.pGroupPerm = DAPI.FULL_PERM_MASK
```

This line grants Complete Access only to the users in that group. All other users only have default access to the Annotations folder.

Note:

- Of the two entries in `csiSaveDoc.html` that control permissions on the Annotations folder at the time of its creation, one is commented by default.
- Only one of the two entries should be activated at any given time.
- To deactivate an entry, add `"/"` after the semi-colon (`;`) and to activate an entry remove the `"/"` after the semi-colon (`;`).
- The entries in `csiSaveDoc.html` are accessed only at the point where VueLink creates the Annotations folder.
- The permissions are configurable by the Livelink Administrator, who may set or reset them at a later point using the Livelink Administration interface.

Specifying Folder Names for Annotations/Viewables/Conversions Folders

By default, the VueLink stores Markups in the Annotations folder, the generated tiff files in the Conversions folder, and the cached metafiles in the Viewables folder. The Livelink administrator has the option to select different folder names.

- 1 In a text editor such as Notepad, open the file **csiSaveDoc.html** found in the directory **<opentext_root>/module/csi/vot_x_x_x/html**, where **<opentext_root>** is the root folder where Livelink is installed.
- 2 Search for the following block of code:

```
;if IsDefined ( Str.LocateI(sQueryString, "csiType=Markup" ) )
; sFolderName = "Annotations"
;elseif IsDefined ( Str.LocateI(sQueryString, "csiType=Viewable" ) )
; sFolderName = "Viewables"
;else
; sFolderName = "Conversions"
;end
```

- 3 Make changes and save the file.
- 4 Restart Livelink Services for the changes to take effect.

Completing the Configuration for XRefs

In order for Livelink to recognize external reference files (XRefs) linked to a drawing, specify the linked files in the **Description** field of the base document.

Note: If Livelink is integrated with CADManager and XRef definitions have been stored by CADManager, AutoVue retrieves XRef information as saved by CADManager. No additional configuration is required.

- 1 From the browser, start Livelink.
- 2 Make sure you have owner/modification rights to the base document to which you will be adding external references.
- 3 Click the drop-down **Function** menu corresponding to the base document.

Note: From the menu, click **Info** and then **General**. This contains the link to the General Info page for that document.

- 4 In the **Description** text box, enter a description for each XRef associated with the active drawing. Use the following format:
 - **X** tells Livelink that you are about to specify a link to an external reference file.
 - **n** is the number Livelink associates with the external reference document (the nodeID of the document), and **<filename>** represents the file name of the external reference document.

Use **colons** to separate each element of the description. The **semi-colon** at the end of the XREF information is also required.

You have a sample file called **chair_3d.dwg**. This file contains an XREF called **adesk_b.dwg**. First, import both drawings into the Livelink database through the **Document** option of the **Add New Item** menu. Livelink assigns each document a unique numeric identifier, or document number.

Example:

chair_3d.dwg may be assigned the identifier 2313 and **adesk_b.dwg**, the identifier 2424.

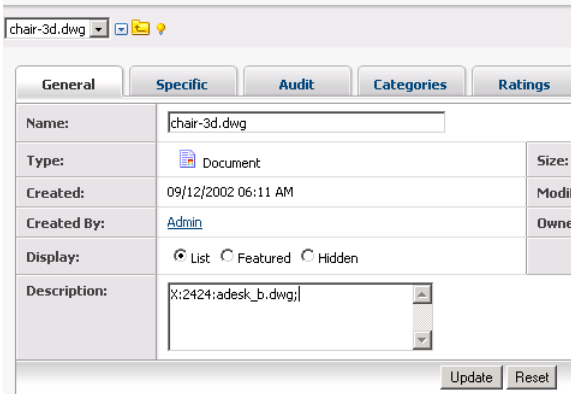
The actual numbers vary from system to system.

For each Reference Document added to the Database

- 1 Move the mouse over the document's name.
- 2 Right-click and select **Copy shortcut** from the pop-up menu.
- 3 Open a text editor, such as Notepad.
- 4 Right-click and select **Paste**.

In the URL that you pasted, the `nodeId` is the document number you need to make a note of, and `docTitle` is the name of the document.
- 5 After importing the two documents, edit the profile of the base document (**chair_3d.dwg**) by adding a description for each XREF associated with the active drawing in the Description text field (as explained above).

Example: In the **Description** text field, enter: **X:2424:adesk_b.dwg;**



- 6 To add multiple external references to the current document, add the XREF information in the **Description** text field of the current document. Each "XREF" is separated by a semicolon (;).

Example: In the **Description** text field, enter:

X:8594:xc101.dwg;X:8482:xcr01.dwg;X:8486:xsk01.dwg;

Assuming the current base document lv01.dwg has the following three external references:

- X:8594:xc101.dwg (XREF Name is xc101.dwg, while "8594" is the nodeID provided by Livelink for the document.)
- X:8482:xcr01.dwg
- X:8486:xsk01.dwg

Printing DMS Attributes

Additional attributes may be added to "AttributeCols" in csiGetProperties.html. To add more attributes, edit the file **/OPENTEXT/module/csi/vot_x_x_x/html/csiGetProperties.html**. Append the additional attributes to AttributeCols.

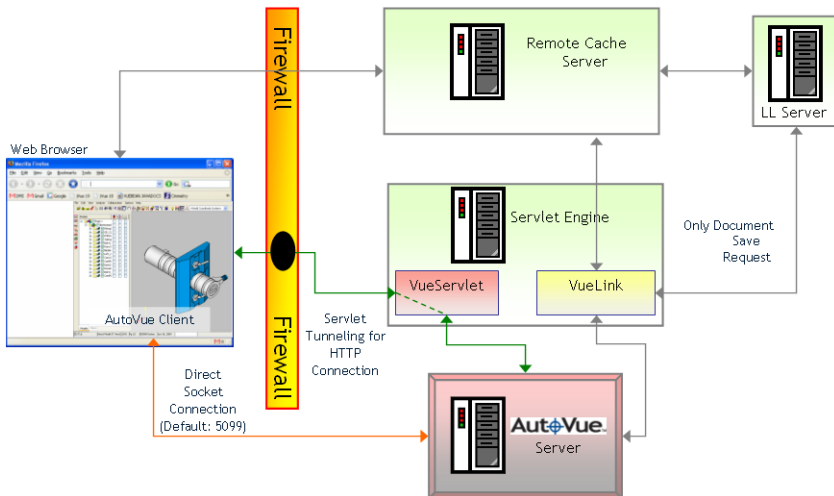
The following is a list of additional attributes to append to AttributeCols:

Attribute Name	Description
CSI_DocName	The descriptive title of the document.
CSI_DocSize	The size of the document in bytes (read only).

CSI_DocDateCreate	The date the document was created (read only).
CSI_DocDateLastModified	The date the document was last modified (read only).
CSI_DocAuthor	Document author/owner.
CSI_UserName	User name.
CSI_Version	The version number of the document.

Configuring for Remote Cache Servers

The VueLink servlet allows AutoVue Server to communicate with Remote Cache Server using standard HTTP protocol. The following diagram illustrates a typical configuration of how AutoVue integrates with Livelink Remote Cache Server.



When the components depicted in the preceding diagram are too far from one another, it can create performance issues. Installing AutoVue Server and VueLink too close to Livelink Primary Server is a common integration setup that prevents optimal performance.

To achieve optimal performance, install AutoVue Server and VueLink close to the Remote Cache Server. You also need to update the applet parameters to point to the relevant VueLink and AutoVue Server instances.

Configuring Remote Cache Servers for Optimal Performance

- 1 Install AutoVue Server on the Remote Cache host machine, or on a machine that is quickly accessible from the network.
- 2 Install VueLink for Open Text (servlet engine component) on the RemoteCache Host machine, or on a machine that is quickly accessible from the network.

Note: You do not need to install the **Open Text and Web Server Components** option. The **csijvot_x_x_x** module should be installed on the Primary Livelink Server.

- 3 In a text editor such as EMACS or Notepad, open the file **csijVue.html** to update the applet parameters. This file should be located in the directory **<PrimaryServer_OpenTextInstallation>module\csijvot_x_x_x\html**, where **<PrimaryServer_OpenTextInstallation>** is the directory where Livelink Primary Server is installed.
- 4 Locate the jVue Configuration Section. The following block of text is an example of the jVue Configuration Section.

```

:/**
:/**          jVue Configuration Section
:/**
;Strings  ARCHIVE          = "jvue.jar"
;String   sCABINETS       = "jvue.cab"
;String   sCODEBASE       = sServerProtocol + "__WEBSERVER__/jVue"
;String   sVERBOSE        = "false"
;String   sJVUESERVER     = "http://__WEBSERVER__/servlet/VueServlet;
                        socket://__JVUESERVER__"
;String   sDMS            = "http://__WEBSERVER__/servlet/com.cimmetry.
                        vuelink.opentext.DMS"
;String   sLLServlet      = ""
:/** Set "ServletIP" to the IP Address of the machine running VueLink Servlet
;String   ServletIP       = "__SERVLET_IP__"
:/**

```

- 5 To reassign the applet parameters with new values, add one of the three following code samples, according to your configuration for integrating AutoVue with Open Text Livelink Primary and Remote Cache Server.

Code Sample 1:

AutoVue Server and VueLink are running on default ports:

5099 for AutoVue and **5098** for VueServlet and VueLink.

AutoVue Server and VueLink are running on the same machine as Remote Cache Server.

```

;Boolean      bApplyRemoteCacheOptimalConfigSetting  = TRUE
;if ( .RemoteClient () && bApplyRemoteCacheOptimalConfigSetting)

    ;String sHostName = request.HTTP_HOST

    ;// Setting for the jVue Server and the VueLink
    ;// e.g
    ;//sCODEBASE = sServerProtocol + "__WEBSERVER__jVue"
    ;//sJVUESERVER = "http://__WEBSERVER__/servlet/VueServlet;socket://__
    JVUESERVER__"
    ;//sDMS = "http://__WEBSERVER__/servlet/com.cimmetry.vuelink.opentext.
    DMS"

    ;// The following settings for AutoVue and VueLink assume that
    ;// AutoVue Server and VueLink are on the same machine as the
    ;// RemoteCache Host Server. This is the optimal setup:
    ;// AutoVue gets the document and its properties from
    ;// the RemoteCache through VueLink. Being on the same machine
    ;// or in close proximity means quicker handling of all
    ;// requests and a shorter response time for Viewing/Marking up
    ;// documents.

    ;sCODEBASE = "http://" + sHostName + "/jVue"
    ;sJVUESERVER ="http://" + sHostName + ":5098/servlet/VueServlet;socket://" +
    sHostName + ":5099"
    ;sDMS = "http://" + sHostName + ":5098/servlet/com.cimmetry.vuelink.opentext.
    DMS"

;end

```

Code Sample 2:

In some cases it may not be suitable to run AutoVue Server and VueLink on the same machine as the Remote Cache Server. Add Code Sample 2 for integrations where one machine is dedicated to AutoVue Server and VueLink, and for each Remote Cache Server instance.

```

;Boolean      bApplyRemoteCacheOptimalConfigSetting  = TRUE
;if ( .RemoteClient ( ) && bApplyRemoteCacheOptimalConfigSetting)

    ;String sHostName = request.HTTP_HOST

    ;// To further extend the RemoteCache Server / Primary Server
    ;// configuration and to add adaptability to any scenario.
    ;// This sample code shows how the applet can be pointed to
    ;// AutoVue on any machine and pointed to VueLink based on certain
        conditions, like
    ;// the RemoteCache Host.
    ;// e.g.
    ;//
    ;// if (Str.CmpI (sHostName, '<REMOTECACHEHOST_1>') == 0)

        ;/sCODEBASE = sServerProtocol + "__WEBSERVER_1jVue"
        ;/sJVUESERVER = "http://__WEBSERVER__1/servlet/VueServlet;socket://
            __JVUESERVER__1"
        ;sDMS ="http://__WEBSERVER__1/servlet/com.cimmetry.vuelink.opentext.
            DMS"

;elseif (Str.CmpI (sHostName, '<REMOTECACHEHOST_2>' == 0)

    ;sCODEBASE = sServerProtocol + __WEBSERVER_2jVue"
    ;sJVUESERVER = "http://__WEBSERVER__2/servlet/VueServlet;socket://__
        JVUESERVER__2"
    ;sDMS = "http://__WEBSERVER__2/servlet/com.cimmetry.vuelink.opentext.
        DMS"

    ;end
;end

```

With the code in Code Sample 2, requests made by "**<REMOTECACHEHOST_1>**" and "**<REMOTECACHEHOST_2>**" are handled separately. The appropriate **WEBSERVER** and **JVUESERVER** values are assigned for each parameter.

Code Sample 3:

There can also be a scenario where one machine with AutoVue Server and VueLink is dedicated for a group of Remote Cache Servers. The code is similar to Code Sample 2, but the different components are grouped together using the logical OR statement (||).

```

;Boolean      bApplyRemoteCacheOptimalConfigSetting  = TRUE
;if ( .RemoteClient ( ) && bApplyRemoteCacheOptimalConfigSetting)

    ;String sHostName = request.HTTP_HOST

    ;// In the case where one machine with AutoVue and VueLink is intended
    ;// to serve one or more RemoteCache Server instances, use a grouped
    ;// comparison.
    ;// e.g.
    ;//
    ;// if ( (Str.CmpI (sHostName, '<REMOTECACHEHOST_1>')
    ;//      == 0) || \
    ;//      (Str.CmpI (sHostName, '<REMOTECACHEHOST_2>')
    ;//      == 0) || \
    ;//      (Str.CmpI (sHostName, '<REMOTECACHEHOST_3>')
    ;//      == 0) )

    ;//sCODEBASE = sServerProtocol + "__WEBSERVER__jVue"
    ;//sJVUESERVER = "http://__WEBSERVER__/servlet/VueServlet;socket://
    ;//__JVUESERVER__"
    ;sDMS ="http://__WEBSERVER__/servlet/com.cimmetry.vuelink.opentext.
    DMS"

    ;end
;end

```

Once the values for "**__WEBSERVER__**" and "**__JVUESERVER__**" are set, the preceding code sample dedicates a machine with AutoVue Server and VueLink to serve "**__REMOTECACHEHOST_1>**", "**__REMOTECACHEHOST_2>**" and "**__REMOTECACHEHOST_3>**".

- 6 After you have finished making the appropriate changes, save csjVue.html.
- 7 Restart the Primary Livelink Server for the changes to take effect.

Enabling HTTPS/SSL

Updating java.security to Enable HTTPS/SSL

The VueLink supports connections to **HTTPS** (Secure HyperText Transfer Protocol) using **SSL** (Secure Socket Layer).

- 1 In a text editor such as Notepad, open the file **java.security** found in the directory **<jre home>\jre\lib\security** where **<jre home>** is the root folder where Sun's Java Runtime Environment (JRE) is installed.
- 2 Locate the line **security.provider.2** under the '# List of Providers' section.
- 3 Below this line add the following line that appears in **bold**:

```
#  
# List of providers and their preference orders (see above):  
#  
security.provider.1=sun.security.provider.Sun  
security.provider.2=com.sun.rsa.jca.Provider  
security.provider.3=com.sun.net.ssl.internal.ssl.Provider
```

- 4 If you already have an entry with **security.provider.3**, replace the 3 with the next available number.

Enabling HTTPS/SSL Support with Application Server/Servlet Engine

If you obtained a Trial SSL ID from VeriSign (CA), then you must install Test CA Root into the JRE used by your Application Server /Servlet Engine. VeriSign has implemented this step to prevent fraudulent use of Test Server IDs. When you purchase a regular Secure Server ID, you will not have to perform this step. For example, if you have obtained Trial SSL ID for Microsoft IIS Web server, then do the following:

- 1 Import the SSL id you obtained from CA into Internet Explorer and export it in Base-64 encoded format X.509.
Note: The SSL id should be the same id that is installed on the Web Server.
- 2 Import the Base-64 SSL id to cacerts file using keytool.exe.

Example:

```
C:\jdk1.3.1\bin>keytool -import -alias otserver -file c:\ssl3.cer -trustcacerts
-v -keystore C:\jdk1.3.1\jre\lib\security\cacerts
```

password: changeit

Serial number: 257bc618dbdcbf7befc81c2fce739a8a

Valid from: Wed Apr 10 20:00:00 EDT 2002 until: Thu Apr 25 19:59:59 EDT 2002

Certificate fingerprints:

MD5: E5:0B:A7:36:B7:B9:76:71:72:D8:D7:7A:EB:57:5D:13

SHA1: E9:CE:9C:35:7F:28:68:7D:6D:1C:0B:4D:18:26:87:63:0C:54:05:98

Trust this certificate? [no]: yes

Certificate was added to keystore

[Saving C:\jdk1.3.1\jre\lib\security\cacerts]

Note: You need to repeat step 2 to import the SSL id into cacerts file for JRE used by AutoVue server. The location for cacerts file is <Install_dir>\jre\1.3\lib\security\cacerts where <Install_dir> is the directory where AutoVue Server is installed.

Example:

```
C:\Program Files\jVue\jre\1.3\bin>keytool -import -alias otserver -file
c:\ssl3.cer -trustcacerts -v -keystore C:\Program
Files\jVue\jre\1.3\lib\security\cacerts
```

- 3 Restart your Application Server/Servlet Engine for changes to take effect.
- 4 To verify that the SSL id was successfully imported into cacerts, use the keytool utility.

Example:

```
C:\jdk1.3.1\bin>keytool -list -keystore C:\jdk1.3.1\jre\lib\security\cacerts
```

Note: When HTTPS is enabled, make sure the file csijvue.html is updated with the right web port for sCODEBASE. Open the file csijvue.html and locate the string sCODEBASE. Specify a valid port after the web server. For IIS, the default port number is 443.

Example:

```
;String sCODEBASE = sServerProtocol + otserver:443/jVue
```

Verification

Verifying that VueLink Servlet is Running Properly

In order to verify that VueLink servlet [com.cimmetry.vuelink.opentext.DMS] is running properly, launch your Web browser and enter the URL of the servlet alias name which you assigned when installing **Vuelink.jar** into the application server. See [Installing VueLink servlet for the servlet engine](#).

Example of a URL:

http://127.0.0.1:8080/vuelink/com.cimmetry.vuelink.opentext.DMS

The following screen shot is a sample response if VueLink is running properly. If you do not receive a similar response, refer to the installation steps outlined in [Installing VueLink servlet for the servlet engine](#).

AutoVue Client Server Edition

Version= "19.1.0"

Build= 6052

Build Date= 2006-07-28

URL= http://qall95sp1vpc:8000/vuelink/com.cimmetry.vuelink.opentext.DMS

RemoteUser= null

QueryString= null

ServletPath= /com.cimmetry.vuelink.opentext.DMS

SessionID= null

AuthorizationType= null

RequestContentLength= -1

ServletEngine= Apache Tomcat/5.0.28

Servlet Init Parameters:

Param: dns.action.Open=com.cimmetry.vuelink.opentext.action.ActionOpen

Param: dns.getprops.CSI_ListItems=com.cimmetry.vuelink.opentext.action.GetPropCSI_ListItems

Param: dns.action.SetProperties=com.cimmetry.vuelink.opentext.action.ActionSetProperties

Param: Verbose=True

Param: dns.action.Save=com.cimmetry.vuelink.opentext.action.ActionSave

Param: dns.getprops.Default=com.cimmetry.vuelink.opentext.action.GetPropDefault

Param: dns.getprops.CSI_Search=com.cimmetry.vuelink.opentext.action.GetPropCSI_Search

Param: dns.getprops.CSI_Versions=com.cimmetry.vuelink.opentext.action.GetPropCSI_XREFS

Param: dns.context=com.cimmetry.vuelink.opentext.ActionContext

Param: dns.action.Download=com.cimmetry.vuelink.opentext.action.ActionDownload

Param: log4jInitFile=C:\Program Files\Cimmetry\VuelinkForOpenText\vuelink4j2ee\WEB-INF\log4jInitFile.txt

Param: dns.getprops.CSI_DocSize=com.cimmetry.vuelink.opentext.action.GetPropCSI_DocSizeOrDate

Param: dns.getprops.CSI_Markups=com.cimmetry.vuelink.opentext.action.GetPropCSI_Markups

Param: dns.getprops.GUI=com.cimmetry.vuelink.opentext.action.GetPropCSI_GUI

Param: dns.getprops.CSI_XREFS=com.cimmetry.vuelink.opentext.action.GetPropCSI_XREFS

Param: dns.getprops.CSI_Renditions=com.cimmetry.vuelink.opentext.action.GetPropCSI_Renditions

Param: dns.getprops.CSI_ListAllProperties=com.cimmetry.vuelink.opentext.action.GetPropCSI_XREFS

Running VueLink Servlet in Debug Mode

VueLink for OpenText uses apache log4j package for logging. Default configurations are set in log4j.properties inside WEB-INF\lib folder of the VueLink application.

You can change the level and location of output by modifying this file. The following figure shows the different levels of logging available:

Will Output Messages Of Level

	DEBUG	INFO	WARN	ERROR	FATAL
Debugger Level	DEBUG	INFO	WARN	ERROR	FATAL
DEBUG	Green	Green	Green	Green	Green
INFO	Red	Green	Green	Green	Green
WARN	Red	Red	Green	Green	Green
ERROR	Red	Red	Red	Green	Green
FATAL	Red	Red	Red	Red	Green
ALL	Green	Green	Green	Green	Green
OFF	Red	Red	Red	Red	Red

For example, if you want to elevate log to DEBUG level then set **log4j.logger.com.cimmetry.vuelink=DEBUG** inside the previously-mentioned file.

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

Important: You will need to startup Jetty independently of the AutoVue Server. To startup Jetty independently, shutdown AutoVue Server, run **StartJetty.bat** that is located in **<AutoVue Installation Directory>\jetty\bin**, then restart AutoVue server.

Feedback

Cimmetry Systems products are designed according to your needs. We would appreciate your feedback, comments or suggestions. Contact us by fax, e-mail or telephone. We added a feedback button to our Web page that activates an easy-to-use feedback form. Let us know what you think!

General Inquiries

Telephone: +1 514-735-3219

Fax: (514) 735-6440

E-mail: info@cimmetry.com

Web Site: <http://www.cimmetry.com>

Sales Inquiries

Telephone: +1 514-735-3219 or 1-800-361-1904

Fax: (514) 735-6440

E-mail: sales@cimmetry.com

Customer Support

Telephone: +1 514-735-9941

Web Site: <http://www.cimmetry.com/support>

