

***AutoVue Document Print Service***  
***Release 20.1.1***  
***Deployment Guide***

ORACLE

September 2011

Copyright © 2008, 2011, Oracle and/or its affiliates. All rights reserved.

Portions of this software Copyright 1996-2007 Glyph & Cog, LLC.

Portions of this software Copyright Unisearch Ltd, Australia.

Portions of this software are owned by Siemens PLM © 1986-2008. All rights reserved.

This software uses ACIS® software by Spatial Technology Inc. ACIS® Copyright © 1994-1999 Spatial Technology Inc. All rights reserved.

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

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S.

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, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

# Contents

<b>PREFACE</b> .....	<b>5</b>
<b>Audience</b> .....	<b>5</b>
<b>Related Documents</b> .....	<b>5</b>
<b>Conventions</b> .....	<b>5</b>
<b>INTRODUCTION</b> .....	<b>6</b>
<b>Architecture and Technology</b> .....	<b>8</b>
<b>SYSTEM REQUIREMENTS</b> .....	<b>11</b>
<b>Prerequisites</b> .....	<b>11</b>
<b>CONFIGURING DPS</b> .....	<b>12</b>
<b>Document Print Services (DPS) Settings</b> .....	<b>12</b>
VueBean Pool Size .....	12
VueBean Availability Timeout.....	12
Parallel Print Jobs .....	12
Number of Print Jobs in the Queue .....	12
Memory Lower Bound .....	13
Upload Protocol.....	13
VueServlet Component.....	13
<b>AutoVue Server Settings</b> .....	<b>13</b>
<b>AutoVue User Profile Settings</b> .....	<b>14</b>
<b>Web Application Server Settings</b> .....	<b>14</b>
<b>DPS Client Code Settings</b> .....	<b>17</b>
<b>General Recommendations</b> .....	<b>17</b>
Deploying DPS .....	17
Installing New Printer for AutoVue Web Services .....	17
<b>DEPLOYING DPS</b> .....	<b>18</b>
<b>Multiple Machine Deployment (Clustering)</b> .....	<b>18</b>
Two Machine Deployment (Non-Clustered).....	19
Four Machine Deployment (Horizontal Cluster).....	20
<b>APPENDIX A: CONFIGURING AUTOVUE WEB SERVICES CLUSTERING ON WEB-LOGIC</b> .....	<b>21</b>
<b>Sample Horizontal Clustering Architecture</b> .....	<b>21</b>
<b>Configuring WebLogic for Horizontal Clustering</b> .....	<b>22</b>
<b>Creating and Deploying Applications</b> .....	<b>25</b>
<b>APPENDIX B: LIST OF SERVICES AVAILABLE IN AUTOVUE DPS</b> .....	<b>28</b>
<b>FEEDBACK</b> .....	<b>29</b>
<b>General Inquiries</b> .....	<b>29</b>
<b>Sales Inquiries</b> .....	<b>29</b>
<b>Customer Support</b> .....	<b>29</b>



# **Preface**

The *AutoVue Document Print Service Deployment Guide* describes the configuration and deployment of AutoVue Document Print Services (DPS).

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 Document Print Service Deployment Guide* is intended for third-party developers (for example, integrators) who want to complement their existing print server solutions by leveraging AutoVue's powerful printing capabilities within their broader enterprise applications.

## **Related Documents**

For more information, refer to the following documents:

- *Release Notes*

## **Conventions**

The following text conventions are used in this document:

---

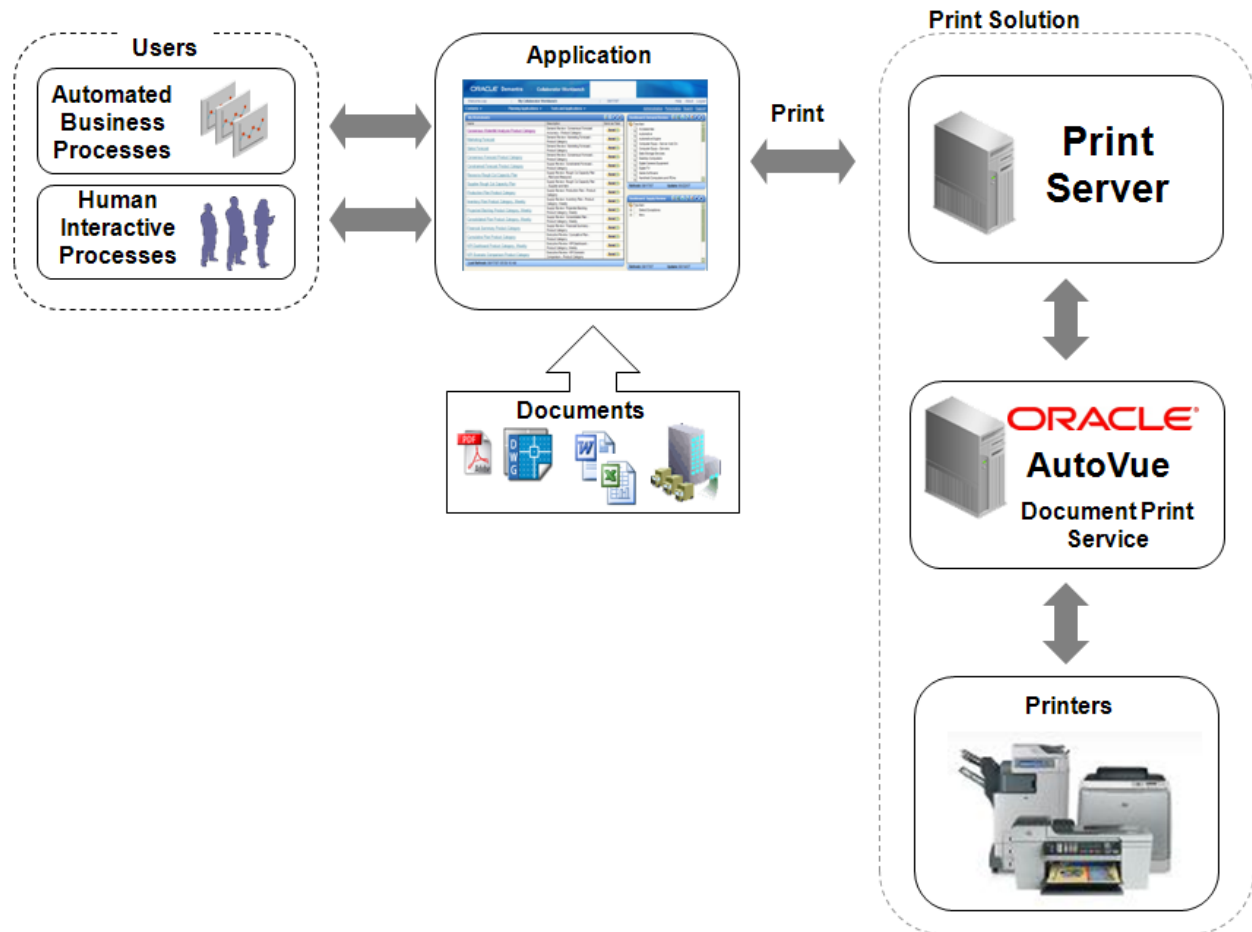
<b>Convention</b>	<b>Meaning</b>
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in the text.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace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

---

# Introduction

The AutoVue Document Print Services (DPS) products allow organizations to complement their existing print server solutions by leveraging AutoVue's powerful printing capabilities within their broader enterprise applications.

The following diagram illustrates a typical configuration of how AutoVue DPS is integrated with other components in your environment.



There are many benefits in incorporating DPS in your enterprise applications to build a comprehensive and integrated printing/batch printing solution. The following are some of the highlights:

- Automate high volume (batch) print capability without having to convert files to another format
- Print Office, 2D and 3D documents in both Linux and Windows environments
- Have a single print solution for all document types (business and technical)
- Enable user-driven or programmatic (background) printing
- Extend existing Content Management Systems, Document Management Systems and other enterprise applications and add support for enterprise printing
- Print documents that are stored in a content repository
- Take advantage of a flexible suite of products based on your business needs
- Priced independently of AutoVue to allow leveraging print-only capability of AutoVue for non-AutoVue users
- Easily integrate using open, industry standard APIs
- Programmatically control print options (header/footer/watermark) and document metadata
- Reduce risk and cost of manual print operations

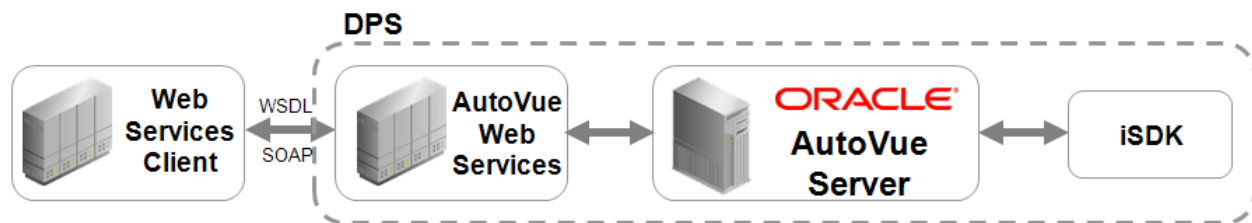
AutoVue Document Print Service is available in three product flavors, depending on your file formats requirements:

- Oracle AutoVue Office Document Print Service: Supports printing of Office formats only.
- Oracle AutoVue 2D Document Print Service: Supports printing of 2D CAD and Office formats.
- Oracle AutoVue 3D Document Print Service: Supports printing of 3D CAD, 2D CAD, and Office formats.

Each Document Print Service bundle contains the following components:

- Oracle AutoVue (Office, 2D, or 3D)
- AutoVue Web Services
- AutoVue Integration SDK (ISDK)

ISDK is optional and is only required if you plan to print files that are stored inside a content repository for which Oracle does not provide an out-of-box VueLink. In this case, you can build a connector based on ISDK framework. For more information, refer to ISDK documentation.



**Note:** Each component of DPS has its own installer program which is available for Windows and Linux platforms.

For installation instructions, please, refer to the installation guide for each component:

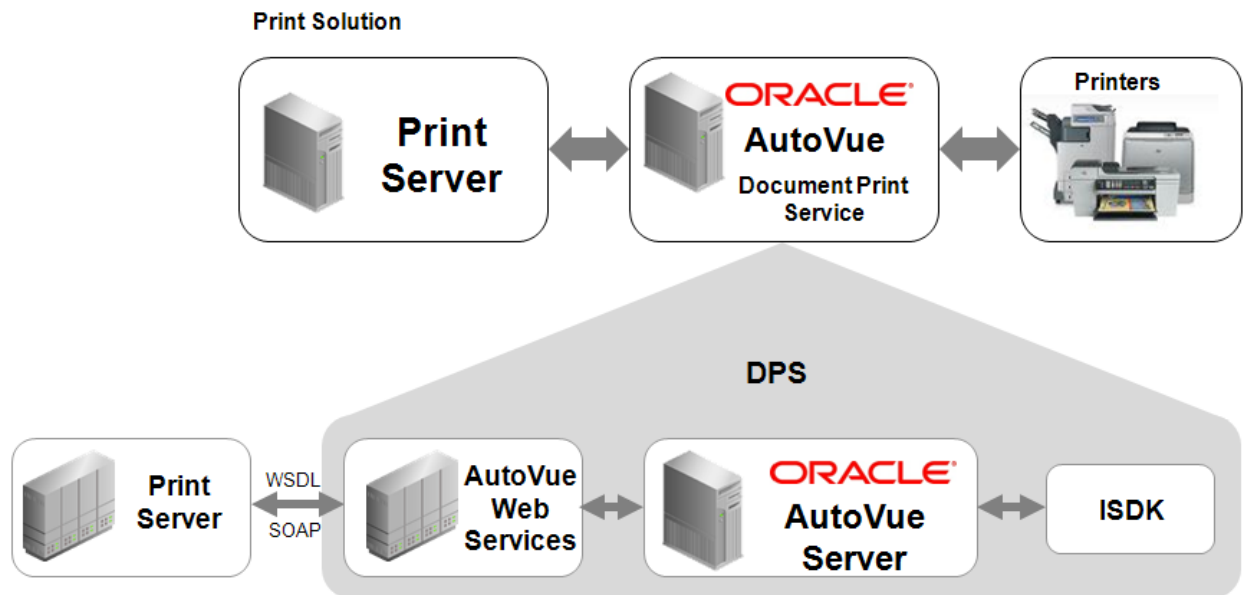
- AutoVue (Office or 2D or 3D): *Installation and Configuration Guide*
- AutoVue Web Services: *Oracle AutoVue Web Services Installation and Configuration Manual*
- AutoVue Integration SDK (ISDK): *AutoVue Integration SDK and Sample Integration for Filesys DMS Installation Guide*

## Architecture and Technology

The AutoVue Document Print Service is a Web Services-based interface. It allows printing of a wide range of file formats and is intended for third-party developers who want to build a print solution that integrates AutoVue's printing capabilities with their applications. Clients that consume DPS can be written in any programming language (for example, Java or any language in a .NET environment) as long as they understand Web Services Description Language (WSDL) and communicate using Simple Object Access Protocol (SOAP).

A print solution consists of the following three main components:

- Print Server
- AutoVue DPS
- Printers

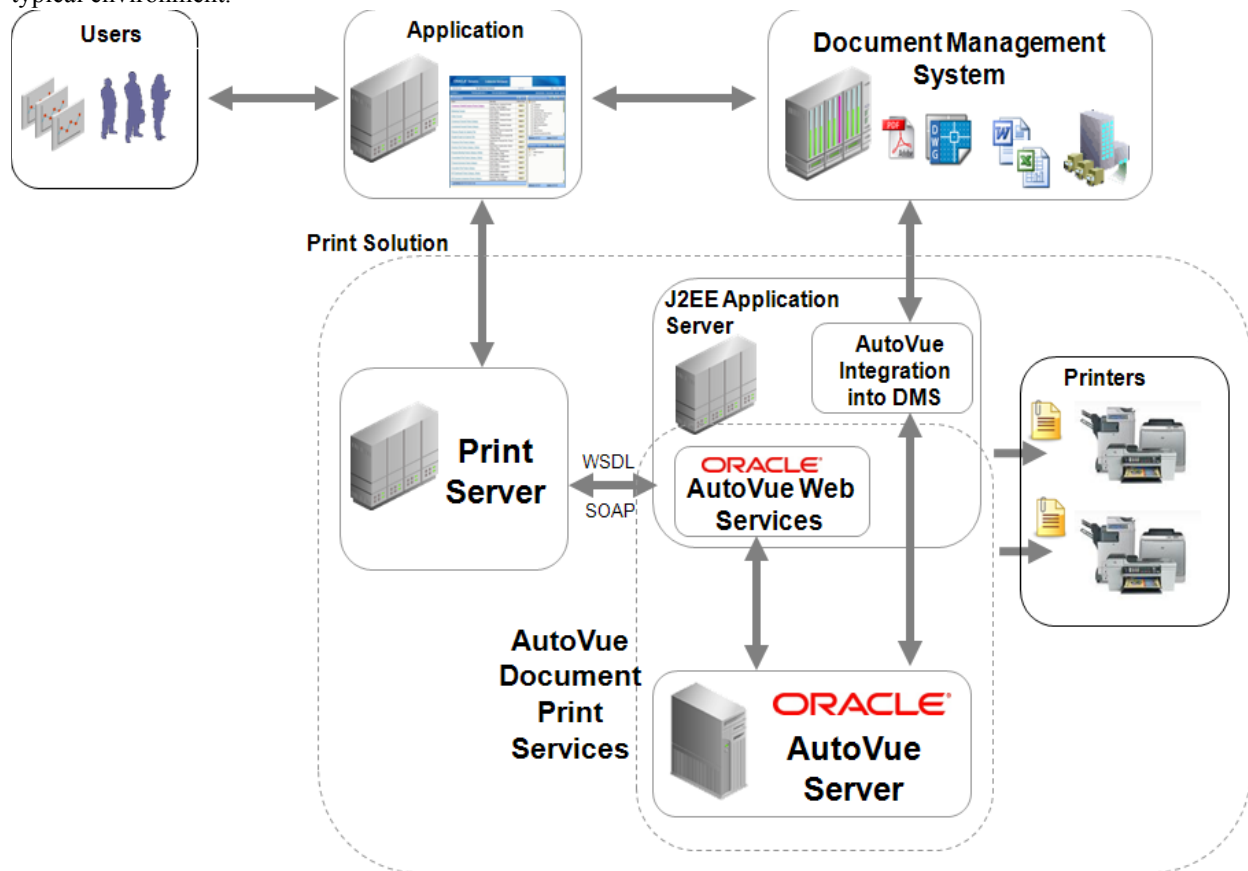


In order to build a complete print solution, you must develop your own print server. The role of a print server may be as follows:

- Interact with your application to identify:
  - The files that need to be printed.
  - The printer to send each file.
  - The print options needed for each file. This includes paper size, page orientation, footers / headers, and so on.
- Collate print job requests.
- Call AutoVue DPS to perform actual printing.
- Poll printer for status.



The following figure is a high-level architectural diagram displaying how the DPS is related to other components in a typical environment.



As shown, you may trigger printing from your application either programmatically through automated business processes or manually through human interaction.

If documents that need to be printed from your application are stored inside a content repository/Document Management System (DMS) such as Oracle Universal Content Management System (UCM), then it is the job of your Print Server to identify the list of documents and pass the ID of each document to the AutoVue DPS to print. In this case, AutoVue DPS leverages the VueLink (a connector to your DMS) to fetch documents out of backend DMS system for printing.

If a connector to your DMS does not exist, you can build one using AutoVue Integration SDK (ISDK). Refer to the ISDK documentation for more information.

If documents that need to be printed from your application are not stored inside a content repository, then it is the task of your Print Server to ensure files are made available to the AutoVue DPS. In this case, you can have your Print Server fetch files out of your application or you can build an extension to your application to fetch files and make them available to the AutoVue DPS. Either way, you may use any of the following methods to pass file information to the AutoVue DPS:

- Copy files into a shared folder location and use either `Server://` or `Upload://` protocol and pass the file path to AutoVue DPS.
- Copy files into a Web server document folder and use `HTTP://` protocol and pass the file URL to AutoVue DPS.

For more information on these protocols, refer to the *Oracle AutoVue Web Services Installation and Configuration Manual*.

**Note:**

- Printers should be visible to the machine on which AutoVue Web Services is installed.
- Files passed to DPS using HTTP:// or Server:// protocols should be accessed directly by AutoVue Server. However, files passed to DPS using the Upload:// protocol should be accessed directly by AutoVue Web Services.
- Files cannot be uploaded directly from the client to AutoVue Web Services.
- Files stored inside a repository should be passed using a DMS protocol. For more information, refer to AutoVue Web Services documentation.

---

# **System Requirements**

Refer to the "System Requirements" section of the *Release Notes* for each of the following components:

- Oracle AutoVue (Office or 2D or 3D)
- AutoVue Web Services
  - Note:** The DPS is certified with Oracle WebLogic 11g (10.3.4)
- AutoVue Integration SDK (ISDK)

## **Prerequisites**

You must first install AutoVue Server and AutoVue Web Services before deploying DPS. Additionally, you must verify that you have all requisite software installed prior to deploying DPS. For a complete platform-specific list, refer to the "System Requirements" section of the *Oracle AutoVue Installation and Configuration Guide*.

# Configuring DPS

AutoVue DPS deployment includes several options that may be modified in accordance with the requirements and capabilities of the underlying platform. This section demonstrates these options and features.

## Document Print Services (DPS) Settings

### VueBean Pool Size

The AutoVue DPS maintains a pool of VueBean objects. Each Web Service request is handled by a VueBean object that is borrowed from this pool. The VueBean object is returned to the pool for reuse after the completion of the request. You can modify the pool size by changing the `maxPoolSize` parameter in the `web.xml` of AutoVueWS Web application component of AutoVue Web Services.

If you want the Web Services to address more requests simultaneously, you may increase the pool size. However, you must consider the memory size, CPU speed, the average size and type of the documents, and the configuration on the AutoVue server side. The size of the VueBean pool directly affects the performance and the load on the AutoVue server and the underlying hardware system. The default value is set to 1. It is possible to increase this number if only small files are being printed. Note that it is required that the size of the VueBean pool does not exceed the size of the AutoVue server process pool (default is 4).

### VueBean Availability Timeout

When all the VueBean objects are busy processing requests, a new request cannot be processed until one of the VueBean objects becomes available. In this case, a new request is put on hold for a certain period of time and waits for a VueBean object to become available in the pool. If a VueBean object is not available by the end of that period, then the request is rejected with a timeout error message. The default value for this timeout period is 7200 seconds and it is set as `maxWait` inside the `web.xml` file of AutoVueWS Web application.

If there are more clients than VueBean pool size, you have the option to change the wait time. If your client wishes to wait for a shorter period of time when the server is busy, you can reduce this number. To set a longer request wait time for a VueBean to be available, you can increase this number. In either case you can always retry the request after it is rejected with a server busy error message. The sample code, `SampleClient.java`, provided in the `<AutoVue DPS installation folder>\sample_client` directory demonstrates a persistent retry as long as the server is busy.

### Parallel Print Jobs

The `maxParallelPrint` job option in `web.xml` of AutoVueWS Web application specifies the number of active print jobs per printer. The default value for this parameter is 1. When set to the default value, at any given time, only one VueBean can print on the same printer.

**Note:** The maximum value for this parameter is the VueBean Pool Size. In general, we do not recommend increasing this value even if you increase the Vuebean Pool Size. Since all of the VueBeans share the same memory, it is better to keep the `maxParallelPrint=1` to limit the possibility of memory starvation during the VueBeans print action. Memory starvation may cause unexpected results in the printout, so changing this value is not recommended unless you always deal with small files.

### Number of Print Jobs in the Queue

The `maxPrinterJobBuffer` options in `web.xml` of the AutoVueWS Web application controls the flow of print jobs. The default value for this parameter is 5. DPS delays the sending of print jobs to the printer if the number print jobs in the printer queue exceeds the default value.

Note that this does not guarantee that the queue will never be more than the value of `maxPrinterJobBuffer` if multiple printing clients are printing at the same time.

This parameter is useful when dealing with slow printers or if there is a limit in the size of printer buffer. If a limit is not needed for the print queue, you can set this parameter to a negative value and eliminate any queue checking in the print service.

## Memory Lower Bound

The `minMemory` parameter in `web.xml` of the AutoVueWS Web application sets the minimum amount of memory required before a `VueBean` is allowed to open a document. If there is not enough memory in a system, and multiple `VueBeans` are being used, the print service does not allow the opening of a new file and throws an exception. This behavior can be controlled by changing the value of this parameter. It is recommended to keep the default value (128 MB), but if you are dealing with big files you might need to increase the value. You must make sure that this number is not more than 25% of the total memory allocated to the Virtual Machine. For example, if the Web application server is set to use 1024 MB then the `minMemory` should not exceed 256 MB. On the client-side code you can always try sending the request again if this exception is returned by print services. It is good practice to delay for couple of minutes before resending a request.

## Upload Protocol

By default, the upload protocol is disabled in the DPS for security reasons. From the performance point-of-view, it is not recommended to use an upload protocol. However, if necessary, the upload protocol can be enabled by setting the `isUploadProtocolEnabled` inside `web.xml` of the AutoVueWS Web application to `TRUE`.

If you are enabling upload protocol, a printing client can print any file that is accessible to the AutoVueWS Web application. So, if not all users should have access or some should have partial access to those files, then the file access management should be addressed in the print service client code and only authorized print service client connection should be allowed to access AutoVue print service.

It is recommended to not use the upload protocol, always use a document management system for storing documents, and use a `VueLink` along with print services to access the documents by passing the user's credentials.

In case using DMS and `VueLink` is not feasible, then choose either an FTP or Server protocol with the AutoVue and print service. Please refer to AutoVue documentation on how to setup the Server protocol.

## VueServlet Component

All the communication between `VueBean` objects and the AutoVue server go through the `VueServlet` component. The `VueServlet` component comes as a separate Web application inside the AutoVue Web Services package. It must be deployed on a separate server instance than the AutoVue Web Services (AutoVueWS Web application) in order to avoid any possible request queuing problems. You must make sure that the initial `JVueServer` parameter inside the `web.xml` file of the AutoVueWS Web application points correctly to your `VueServlet` Web application URL location. Inside the `VueServlet` Web application, update its `web.xml` for the `JVueServer` parameter to point to the correct host name and port number that the AutoVue server is running on.

## AutoVue Server Settings

AutoVue server can be configured through its option files. For more information and recommendations on setting these option files, refer to the *Oracle AutoVue Installation and Configuration Guide*. One recommendation from DPS point of view is to disable the Streaming file option. This is particularly important if large queues of requests are being sent to Web Services. To disable streaming file generation, set the `jvueserver.metacache.enable` to `false` from inside `jvueserver.properties` file (`jvueserver.metacache.enable=false`).

Another option to disable is collaboration. Since the AutoVue server is not being used for collaboration when it is dedicated to AutoVue Web Services, disabling collaboration reduces the overhead and improves the performance. You can do that by setting `javueserver.collaboration.enable` to `false` inside `javueserver.properties` file (`javueserver.collaboration.enable=false`).

AutoVue properties file, `javueserver.properties`, is the main location for modifying AutoVue server settings.

For more information on available options and their effects, refer to the *Oracle AutoVue Viewing Configuration Guide*.

## AutoVue User Profile Settings

You can specify default user profile settings for all users that connect to the AutoVue server. When users connect to AutoVue for the first time (for example, when profiles are created for the first time), the contents of `default.ini` are copied to the users' own INI file ("username".ini). If you want all users that connect to AutoVue to have the same initial default options, specify these options in `default.ini`. When users connect to the AutoVue server, options set in `allusers.ini` are transferred to the user profile. The following are recommended

In addition to this file, the `allusers.ini` file can be used to set global user related settings. Some of the settings are format/type-specific. Depending on your environment and documents, you might need to use the format/type-specific options as well.

For example, we recommend setting the following option for the IFC file format inside `allusers.ini` file under the [Options] section:

```
IFCLoadInvisibleSpaces=0
```

Setting this option disables loading of internal spaces boundary geometry and improves both time and memory usage when files contain a large number of invisible spaces.

We also recommend that you set the `TILELIMIT` INI option to a positive number (hard limit). For example, if the scaling settings generate an output for a single page that spans multiple pages, then setting this option to a positive number limits the output to fit to a single page. Since there is no user-interface with DPS, setting this option is highly recommended. For more information, refer to the *Oracle AutoVue Viewing Configuration Guide*.

## Web Application Server Settings

Oracle WebLogic is the certified Web application server for the deployment of AutoVue Web Services package. Since some of the processes may take a few minutes to complete (for example, printing a large file), it is recommended to increase the **Stuck Thread Max Time** parameter for the servers that are deploying Web Services component to 7200 seconds. To do so, you must set the **Stuck Thread Max Time** parameter in two locations: the Tuning tab and the Overload tab (both are located under the Configuration tab).

- 1 From the WebLogic server that you are using for AutoVue DPS select the **Configuration** tab, then select the **Tuning** tab, and then set the **Struck Thread Max Time**.


Settings for AdminServer

**Configuration** Protocols Logging Debug Monitoring Control Deployments Services Security Notes

General Cluster Services Keystores SSL Federation Services Deployment Migration **Tuning** Overload


Save


Use this page to tune the performance and functionality of this server.

 **Enable Native IO**

**Socket Readers:**

**Maximum Open Sockets:**

 **Stuck Thread Max Time:**


 **Stuck Thread Timer Interval:**

**Accept Backlog:**

**Login Timeout:**

**SSL Login Timeout:**

**Reverse DNS Allowed**

 **Low Memory GC Threshold:**

- 2 From the WebLogic server that you are using for AutoVue DPS select the **Configuration** tab, then select the **Overload** tab, and then set the **Struck Thread Max Time**.


Settings for AdminServer

Configuration Protocols Logging Debug Monitoring Control Deployments Services Security Notes


General Cluster Services Keystores SSL Federation Services Deployment Migration Tuning **Overload** Health Monitoring


Save


Use this page to configure how WebLogic Server should react in the case of an overload or failure condition.


 **Shared Capacity For Work Managers:**


**Failure Action:**

 **Panic Action:**

 **Free Memory Percent High Threshold:**

 **Free Memory Percent Low Threshold:**

 **Max Stuck Thread Time:**

 **Stuck Thread Count:**

You may increase the Complete Message Timeout and/or Idle Connection Timeout values under the Protocols tab and then the General tab if you need to print large files.

Another option to verify/update is the memory options of the Java Virtual Machine (JVM) that is used by the application server. Note the following recommended updates:

- It is recommended to set the minimum heap size to 512 MB (`-Xms512m`) and the maximum size to 1024 MB (`-Xmx1024m`) for managed servers which deploy DPS AutoVue Web Services. The file `setDomainEnv.cmd` (.sh for linux) located in the inside the `<WebLogic domain>\bin` directory holds the values for the JVM options. The default value in WebLogic is a minimum heap of 256MB and a maximum value of 512 MB.
- To enforce an incremental garbage collection mechanism for the JVM, you may add the `-Xincgc` parameter to the list of JVM parameters in the `setDomainEnv.cmd` (.sh for linux) file. This can be done by adding `-Xincgc` to the end of `JAVA_PROPERTIES` in the file.
- You might also want to experiment with JVM setting `-XX:NewRatio=1` in combination with `-Xms1024m` and `-Xmx1024m` in case you need to allocate more memory space to new objects created during printing, but it will trigger a bit more expensive full garbage collection. For example, VueBean methods `setfile()` and `print()` require ample memory, so a more frequent garbage collection is required.

You must restart the application server for JVM changes to take effect.



## DPS Client Code Settings

The client code that invokes DPS can also have some effect on the performance. A sample client code is included in the package. There are several print options that can be set by the client code as part of the print request. It is recommended to set the Java Printing option to TRUE (by default it is set to TRUE on the DPS side if the client does not change it). It is also recommended to enable the Send Page as Image option for 3D files and disable the option for the office documents.

It is recommended that the number of DPS clients should not exceed the size of the VueBean pool (default is 1). If more clients than the number of VueBeans are initiating requests at the same time, then the extra clients are blocked for a period of time until a VueBean becomes available (as described in the ["VueBean Availability Timeout"](#) section).

The sample code, SampleClient.java, provided in the <AutoVue DPS installation folder>\sample\_client directory demonstrates how to retry a request that is rejected due to the server is too busy or in a low memory state.

## General Recommendations

### Deploying DPS

The settings for AutoVue Web Services, AutoVue and Application Servers are to be used as self-tuning options regardless of the overall deployment topology. As for the overall deployment, it is highly recommended to have a dedicated AutoVue server for AutoVue Web Services and have it deployed on a separate machine than the one where the AutoVue Web Services is deployed. This is particularly important if you are printing large documents (for example, a file size larger than 10MB). If any VueLink is required as part of the deployment to access the documents it is recommended to deploy it on the same machine as the AutoVue Web Services rather than the AutoVue machine. The VueLinks can be deployed on a separate server or on the same server that VueServlet Web application is using.

### Installing New Printer for AutoVue Web Services

It is recommend to restart Web Logic in order to detect a newly installed printer for AutoVue Web Services.

# Deploying DPS

There are many possible deployment scenarios for DPS. For example, you may choose to deploy DPS on a single machine or on multiple machines (clustering). However, all deployment scenarios must follow these three steps:

- 1 Since AutoVue DPS provides a Web Services-based interface, you must deploy AutoVue Web Services on a J2EE5 compliant application server such as Oracle WebLogic.

**Important:** We recommend that you configure your application server to allow HTTP connections over Secure Socket Layer (SSL). For more information on how to set up AutoVue Web Services to run in a SSL environment, refer to section “HTTPS/SSL” in the *AutoVue Web Services Developer’s Guide*. If you choose not to run AutoVue Web Services over SSL, any data (including any user credentials) sent to AutoVue Web Services will be in clear text and not encrypted. It is recommended to use SSL for secure communication.

- 2 Once you have successfully installed AutoVue Server and deployed AutoVue Web Services, you may familiarize yourself with the available Web Services API.

**Note:** For information on features and functionalities provided by each Web Service, refer to the *Oracle AutoVue Web Services Developer’s Guide*. Note that only the print-related Web services are available if you purchase DPS.

- 3 Build a print server that acts as a client to the DPS by creating a Web Services proxy client that consumes DPS.

For technical details and instructions for developing your own proxy client, refer to the *Oracle AutoVue Web Service’s Developer’s Guide*.

Refer to "[Appendix A: Configuring AutoVue Web Services Clustering on WebLogic](#)" for a sample horizontal cluster deployment.

The following sections provide information on multiple machine deployment scenarios.

**Note:** A sample code for packet printing functionality, `SamplePacketPrintClient.java`, is provided in this release.

## Multiple Machine Deployment (Clustering)

Clustering of the AutoVue Web Services may be accomplished at the Application Server level. Oracle WebLogic is a certified Web application server for clustering AutoVue Web Services. Note that you must use the exact same version of Oracle WebLogic for all the servers in the cluster.

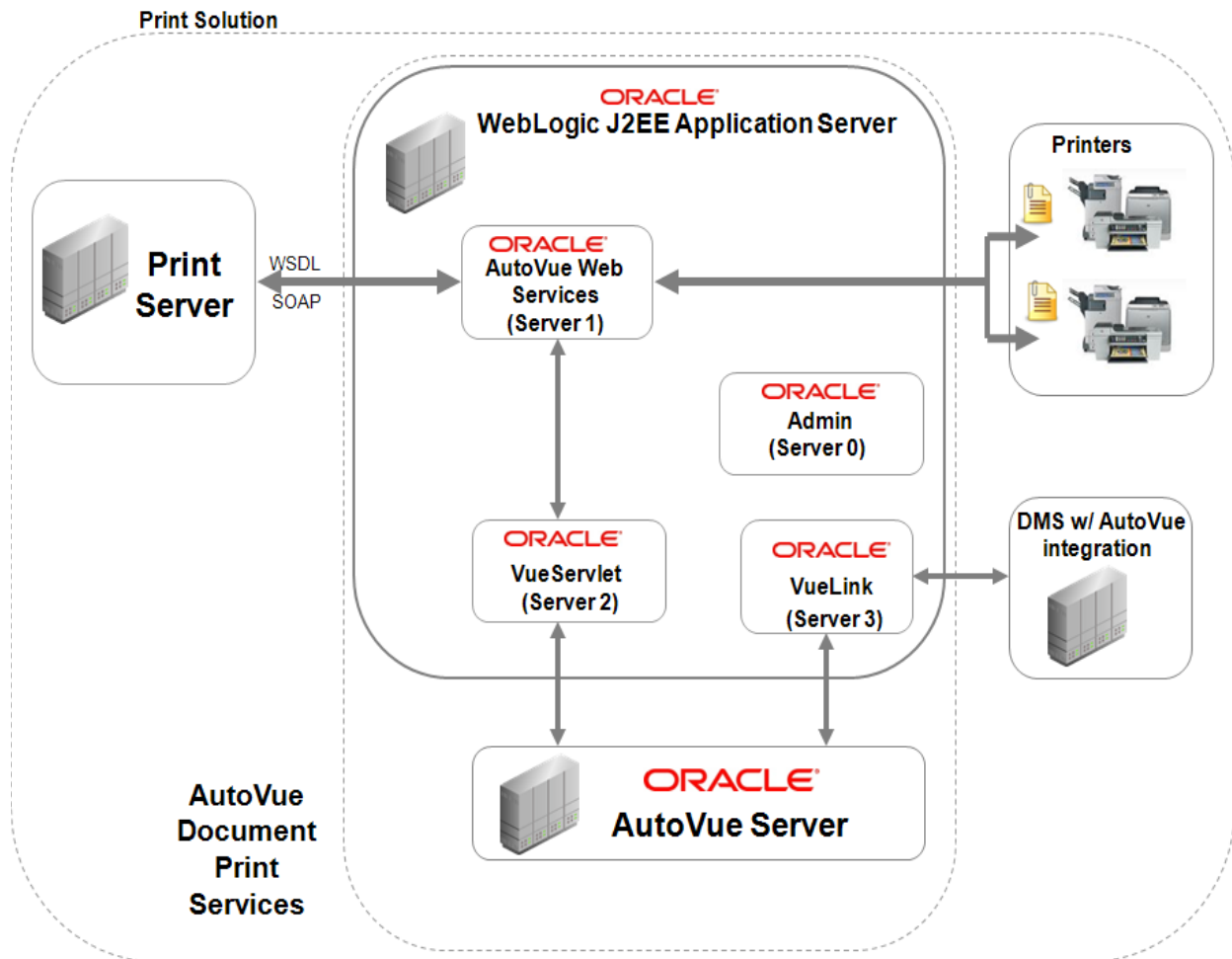
For more information on clustering, refer to Oracle WebLogic documentation.

Clustering of AutoVue Server can be done through its RMI settings. For more information on how to setup the cluster, refer to *Oracle AutoVue Installation Guide*.

The following are two recommended deployment scenarios for DPS:

## Two Machine Deployment (Non-Clustered)

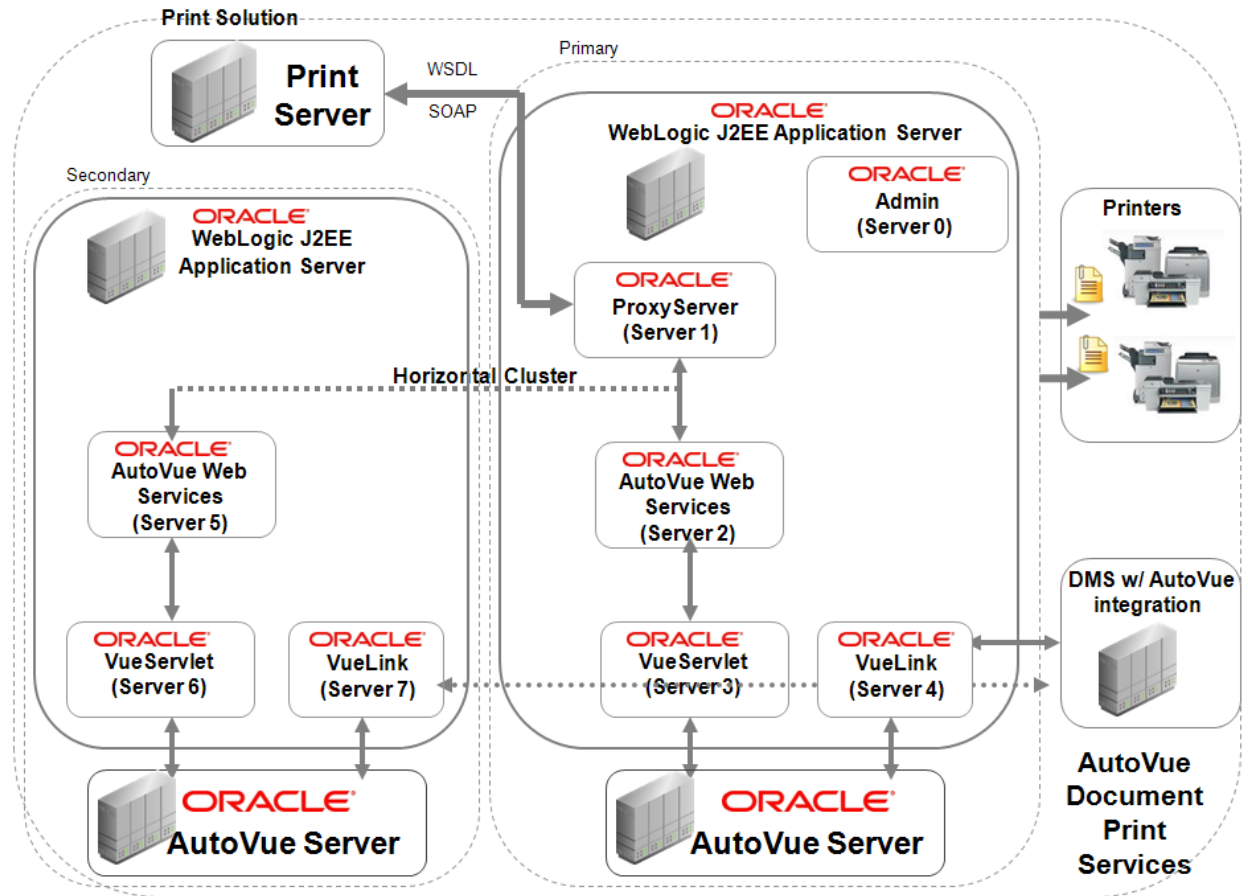
Two machine deployment with AutoVue Web Services and AutoVue Server running on different machines.



This setup is recommended if large files are being printed. In this case, if a VueLink exists, it should be deployed on the AutoVue Web Services machine. The VueServlet may be shared between VueLink and AutoVue Web Services.

## Four Machine Deployment (Horizontal Cluster)

Four-machine deployment - Horizontal cluster of AutoVue DPS Servers with AutoVue Servers on two separate machines.



- The Web Services is clustered through WebLogic clustering.
- The AutoVue servers can be clustered to achieve failover and load balancing.
- The VueLinks (the connector to your DMS), if present need to be clustered as well in order to keep the load balanced on each application server.

# Appendix A: Configuring AutoVue Web Services Clustering on WebLogic

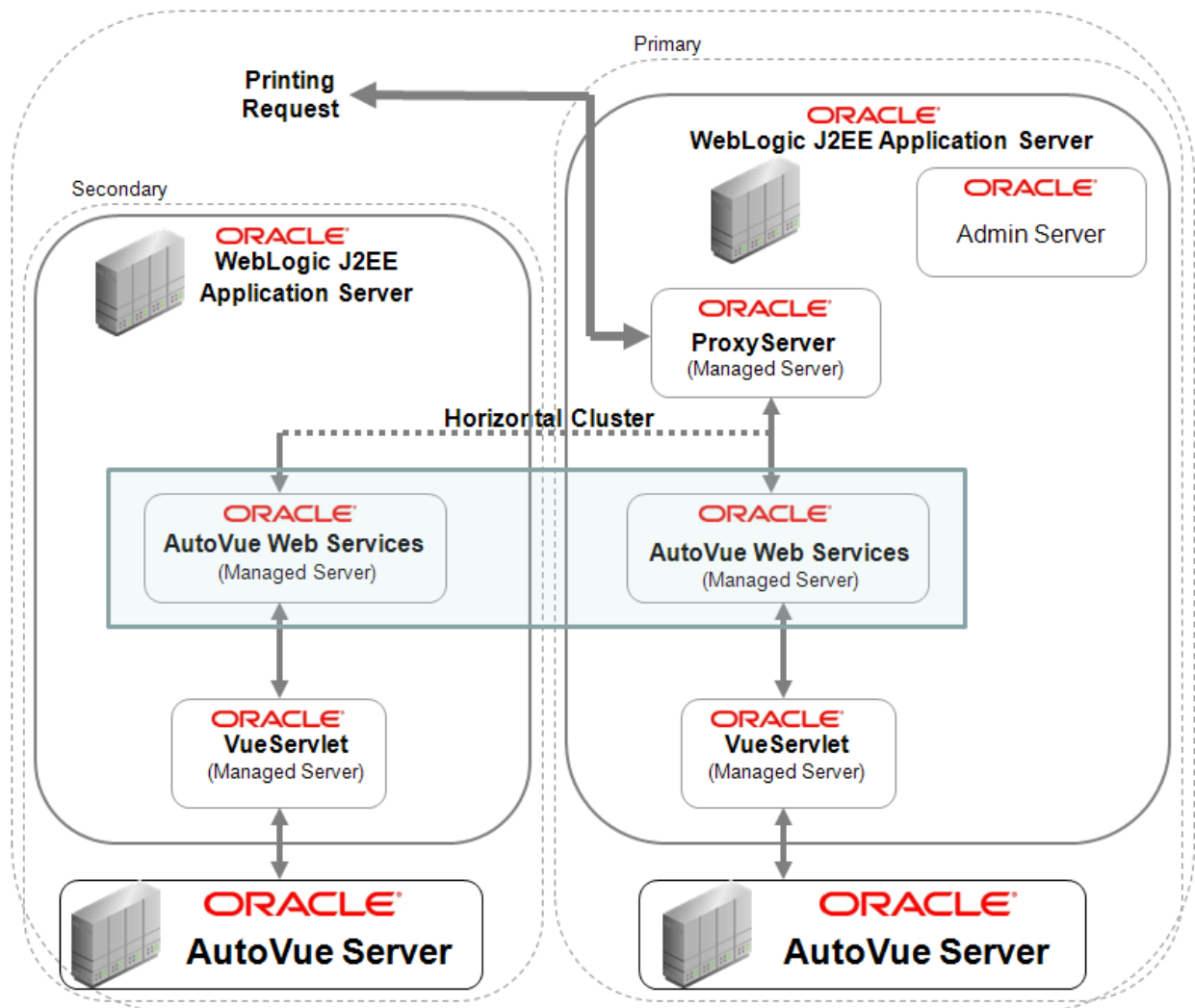
AutoVue Web Services may be deployed on multiple managed WebLogic Servers cluster. The WebLogic Server cluster provides a more scalable and reliable platform as it improves the response time to clients as long as the AutoVue server is able to handle the work load.

Horizontal clustering on different machines is recommended for AutoVue Web Services clustering. In this case, you must ensure that the WebLogic servers are of the same version.

A WebLogic Server proxy plug-in provides round robin load balancing and allows server cluster instances to appear as a single server to clients.

## Sample Horizontal Clustering Architecture

The following diagram displays a cluster where AutoVue Web Services is clustered on two different machines and two AutoVue servers are on two other machines.



The following table describes which server is installed on which machine.

Machine 1	Machine 2	Machine 3	Machine 4
Admin Server	AutoVue Web Services Server 2	AutoVue Server	AutoVue Server
Proxy Server	VueServlet Server 2		
AutoVue Web Services Server 1			
VueServlet Server 1			

Note the following when using horizontal clustering:

- Two managed servers (AutoVue Web Services Server 1 and 2) are deployed with AutoVue Web Services. The servers are running on two different machines.
- The proxy server is responsible for load balancing client requests between the two AutoVue Web Services servers. The proxy server is a managed server that does not belong to a cluster.
- Two AutoVue Servers are running on two different machines. The two AutoVue Servers may be clustered to achieve failover and load balancing.
- Two managed servers (VueServlet Server 1 and 2) are deployed with the VueServlet Web application and are running on two different machines. On each machine, AutoVue Web Services communicates with the VueServlet on the same machine and each VueServlet communicates with its own AutoVue Server.

## Configuring WebLogic for Horizontal Clustering

The section describes how to configure WebLogic for horizontal clustering.

**Note:** Steps 1 through 6 are conducted on Machine-1. Steps 7 through 10 are performed on Machine-2.

### Step 1

Install WebLogic. Run the Configuration Wizard to create a domain.

**For example:** base\_domain

### Step 2

Create managed servers, cluster and machines. When creating a domain you may also create all the managed servers, cluster and machines. Alternately, you can create a single domain with the Administration Server and then create the remaining using the Admin Console.

To create managed servers, cluster, proxy server and machines using the WebLogic Admin Server Console:

- 1 Create two machines:
  - Machine-1
  - Machine-2
- 2 Create one cluster:
  - Cluster1
- 3 Create five managed servers:
  - AVWSServer-1
  - AVWSServer-2
  - VueServletServer-1
  - VueServletServer-2

- ProxyServer

**Note:** Make sure to provide correct listen ports and listen addresses.

- 4 Assign AVWSServer-1 and AVWSServer-2 to Cluster1.

Assign AVWSServer-2 and VueServletServer-2 to Machine-2.

Assign all other servers to Machine-1.

To create managed servers, cluster, proxy server and machines using the WebLogic Configuration Wizard:

- 1 Follow the onscreen instructions and accept the default settings until the Select Optional Configuration page.
- 2 At the Select Optional Configuration page, select **Administration Server** and **Managed Servers, Clusters and Machines**.
- 3 Accept the default settings for Administration Server configuration on the next page.
- 4 On the Configure Managed Servers page, click Add to add the following managed servers:
  - AVWSServer-1
  - AVWSServer-2
  - VueServletServer-1
  - VueServletServer-2
  - ProxyServer

**Note:** Make sure to provide the correct listen address and listen port when creating the servers. Servers that listen on a different address may have the same port number.

- 5 On the Configure Clusters page, add a cluster and use the default **unicast** cluster messaging mode.
- 6 On the Assign Servers to Clusters page, add AVWSServer-1 and AVWSServer-2 to Cluster1.
- 7 On the Create HTTP Proxy Applications page, select the **Create HTTP Proxy** checkbox and then select **ProxyServer** from the Proxy Server list.
- 8 On the Configure Machines page, add two machines in correspondence to the two physical machines.
- 9 On the Assign Server to Machines page, add servers to the two machines.
  - AVWSServer-2 and VueServletServer-2 are added to Machine-2.
  - AVWSServer-1, VueServletServer-1, and ProxyServer are added to Machine-1.
- 10 On the Configuration Summary page, take note of the default application that is going to be deployed on ProxyServer.
- 11 Click **Create** to create the domain.

### Step 3

Update server configurations.

- 1 From the General menu select Advanced.
- 2 Select **WebLogic Plug-in Enabled** for AVWSServer-1 and AVWSServer-2.
- 3 Update the configuration for the servers.

### Step 4

Create a domain template on Machine-1 where AdminServer is installed. Pack up the domain configuration on Machine-1 using the following DOS command. Note that this configuration is to be used on Machine-2 to create the same domain.

```
cd C:\Oracle\Middleware\user_projects\domains
C:\Oracle\Middleware\wlserver_10.3\common\bin\pack.cmd -domain=base_domain
-template=c:\temp\mytest.jar -template_name=mytest -managed=true
```

**Step 5**

Update WebLogic configuration for server memory. Provide the following arguments in WebLogic's Domain.Env.cmd file located in the base\_domain\bin\set directory:

- WLS\_MEM\_ARGS
- MEM\_ARGS

**Step 6**

Start servers that belong to Machine-1. AdminServer, AVWSServer-1, VueServletServer-1, and ProxyServer are started from Machine-1.

It is possible to start/stop the managed servers from the Admin Console by running the node manager (for example, located in the C:\Oracle\Middleware\wlserver\_10.3\server\bin\startNodeManager directory).

Alternately, you can also create batch files to start/stop the managed server directly. For example, create a startManagedWebLogic-AVWSServer-1.bat file in the base-domain\bin directory to start AVWSServer-1.

The server may be started using its own memory setting by providing USER\_MEM\_ARGS. For example:

```
title AVWSServer-1
set USER_MEM_ARGS=-Xms512m -Xmx1024m -XX:PermSize=64m -XX:MaxPermSize=128m
C:\Oracle\Middleware\user_projects\domains\base_domain\bin\startManagedWebL
ogic.cmd "AVWSServer-1" http://10.156.45.164:7001
```

**Note:** You must provide the managed server's name and the URL of AdminServer as parameters to startManagedWebLogic.cmd.

After starting a managed server, a separate folder is created for it. For example:

```
C:\Oracle\Middleware\user_projects\domains\base_domain\bin\servers\AVWSServ
er-1
```

To avoid having to entering the username and password information when starting a managed server, you may copy the **security** folder of the AdminServer to a managed server's folder or provide "WLS\_USER" and "WLS\_PW" in the startManagedWebLogic.cmd or startManagedWebLogic.sh.

**Step 7**

Install the same version of WebLogic on Machine-2. Do not run the Configuration Wizard to create a domain.

**Step 8**

Copy over the c:\temp\mytest.jar file from Machine-1 (as created in step 4 when configuring Machine-1).

**Step 9**

Enter a DOS command similar to the following:

```
C:\Oracle\Middleware\wlserver_10.3\common\bin\unpack.cmd
-domain=C:\Oracle\Middleware\user_projects\domains\base_domain
-template=c:\temp\mytest.jar
```

The defined domain is successfully created on this machine.

**Step 10**



Start the managed servers on Machine-2. You can start AVWSServer-2 and VueServletServer-2 from Admin Console using Node Manager or using batch scripts. For example, create a startManagedWebLogic-AVWSServer-2.bat file in the C:\Oracle\Middleware\user\_projects\domains\base\_domain\bin\ directory.

The content of the batch is as follows:

```
title AVWSServer-2
set USER_MEM_ARGS=-Xms512m -Xmx1024m -XX:PermSize=64m -XX:MaxPermSize=128m
C:\Oracle\Middleware\user_projects\domains\base_domain\bin\startManagedWebLogic.cmd "AVWSServer-2" http://10.156.45.164:7001
```

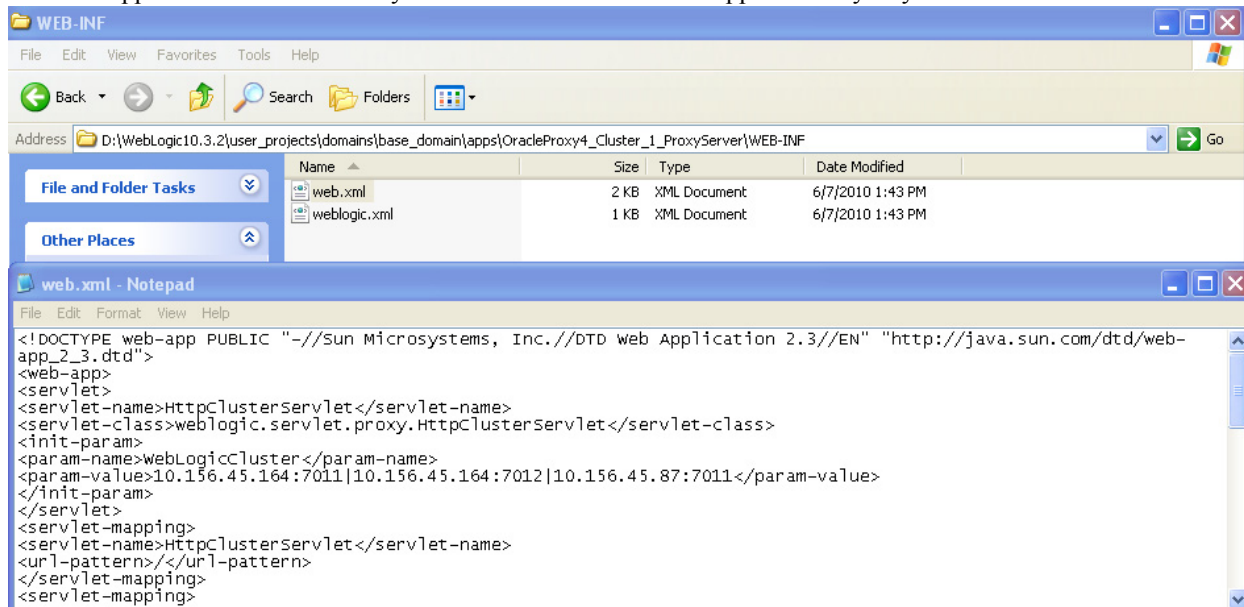
**Note:** Make sure the you provide the correct URL of AdminServer on Machine-1.

## Creating and Deploying Applications

The following steps describe how to create and deploy VueServlet, AutoVue Web Services and proxy Web applications.

- 1 Start AdminServer on Machine-1 and log into the Admin Console.
- 2 Generate two VueServlet Web applications: one for VueServletServer-1 and one for VueServletServer-2.  
**Note:** Make sure to update web.xml to provide correct values for JVueServer and VueServlet.
- 3 Deploy the two VueServlet Web applications to VueServletServer-1 and VueServletServer-2 separately.  
**Note:** You need to use two different deployment names but with the same context root (VueServlet).
- 4 Generate two AutoVue Web Services applications: one for AVWSServer-1 and one for AVWSServer-2.  
**Note:** To make AutoVue Web Service log available on Machine-2, make sure that Machine-2 has the same log4jPathFileName file (defined in the web.xml of AutoVueWS Web application) available at the same path as on Machine-1. By default, it is located under C:\Oracle\AutoVueWS\autovue\_webservices\sample\_config\log4j.properties.
- 5 Deploy the two AutoVue Web Services to AVWSServer-1 AVWSServer-2 separately.  
**Note:** Make sure to provide different deployment names but with the same context root (AutoVueWS) for the two AutoVue Web Services.
- 6 Generate a proxy Web application.

If a proxy server was created along with the domain on Machine-1 using the Configuration Wizard in Step 1, the Web application is automatically created and is located in the apps directory of your created domain.



However, if you created proxy server using the WebLogic Admin Console after you created a simple domain, you must copy over the sample HttpClusterServlet Web application to your desired location and deploy it to the proxy server (ProxyServer).

In both cases, you must customize the web.xml file of the proxy Web application. To do so, you must first update the value of WebLogicCluster. This parameter communicates to the proxy to load balance specified servers inside a cluster. In this case, you must add the Listen Address:Listen port of the two AutoVue Web Services servers to it. For example:

```
<init-param>
<param-name>WebLogicCluster</param-name>
<param-value>10.156.45.164:7011|10.156.45.87:7011</param-value>
</init-param>
```

For information on other parameters that must be set, refer to “Table 10-1” in *Oracle Fusion Middleware Using Clusters for Oracle WebLogic Server 11g Release 1 (10.3.1) E13709-02*. The following are some sample parameter that may be set in web.xml:

```
<init-param>
<param-name>ConnectTimeoutSecs</param-name>
<param-value>100</param-value>
</init-param>
<init-param>
<param-name>ConnectRetrySecs</param-name>
<param-value>10</param-value>
</init-param>
<init-param>
<param-name>Debug</param-name>
<param-value>ERR</param-value>
</init-param>
<init-param>
<param-name>DebugConfigInfo</param-name>
<param-value>OFF</param-value>
</init-param>
<init-param>
<param-name>WLSocketTimeoutSecs</param-name>
<param-value>1800</param-value>
</init-param>
<init-param>
<param-name>WLIOTimeoutSecs</param-name>
<param-value>3600</param-value>
</init-param>
```

**Note:** To avoid duplicate printing for very large files, the value for WLSocketTimeoutSecs might need to be increased, for example, to 3600 (seconds). Additionally, the value for WLSocketTimeoutSecs must be considerably higher when packet printing is being used since a packet print response time is equal to the print time of all documents in the packet. For example, if it takes more than 3 hours to print the whole packet, then this value must be greater than 3 hours. Otherwise, the proxy server resends the request to another server.

7 Deploy the proxy Web application to the proxy server (ProxyServer).

The configuration of WebLogic for horizontal clustering is now complete.

---

## **Appendix B: List of Services Available in AutoVue DPS**

The following is a list of APIs provided by AutoVue Document Print Service. For detailed information, refer to the AutoVue Web Services Developer's Guide.

---

<b>Print Services API</b>	<b>Description</b>
<code>print</code>	This printing Web Service method sends a given file to a printer for printing.
<code>packetPrint</code>	This Web Service method accepts a list of documents and then sends them to a printer one at a time.
<code>getPrinterNameList</code>	This utility Web Service method returns a list of available printers.
<code>getPaperList</code>	This utility Web Service method returns a list each printer's available paper sizes.
<code>ping</code>	This utility Web Service method echoes services response.

---

## **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 Web Services, 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](mailto: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](mailto:autovuesales_ww@oracle.com)

## **Customer Support**

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

