SAP BusinessObjects

SAP BusinessObjects Enterprise Web Application Deployment Guide

- SAP BusinessObjects Enterprise XI 4.0

2010-11-26

SAP

2010-11-26

Contents

Getting Started

1.1 What is SAP BusinessObjects Enterprise?

SAP BusinessObjects Enterprise is a flexible, scalable, and reliable business intelligence reporting system that can be tightly integrated into your information technology infrastructure. Support for many industry-standard database systems makes it easier to access your organization's data for analysis. The use of common industry standards for security allow you to use your existing authentication systems to control access to SAP BusinessObjects Enterprise. And broad platform support allows you to implement the operating systems and hardware architecture that you prefer.

As a system administrator, you will be faced with many choices when installing SAP BusinessObjects Enterprise. This documentation helps you to make the right decisions and create a reliable and powerful business intelligence reporting system for your organization.

1.2 About this document

This document provides information on how to configure and deploy SAP BusinessObjects Enterprise web applications to a supported Java web application server. In particular, this guide contains detailed information for users of the WDeploy web application deployment tool that ships with SAP BusinessObjects Enterprise.

For information related to the installation of SAP BusinessObjects Enterprise, see the *SAP BusinessObjects Enterprise Installation Guide*.

For information related to the administration of an SAP BusinessObjects Enterprise server, see the *SAP BusinessObjects Enterprise Administrator Guide*.

1.3 What's new in the SAP BusinessObjects Enterprise Web Application Deployment Guide?

This document is the first to include information on the following features in this version of SAP BusinessObjects Enterprise:

Table 1-1: What's new with the WDeploy deployment tool

Feature	Description
GUI interface	New GUI interface for `deployall` and `undeployall` actions.
Two levels of log files	One log summary to help administrators understand the deployment status; one detailed log to help developers troubleshoot deployment issues.
Easier configuration	Centralized configuration file for WDeploy global parameters; access server and WDeploy parameters from WDeploy GUI.
Localization support	WDeploy GUI localization support.
No intermediate WAR files	Intermediate WAR files are no longer created as a part of the deployment process. If you need to create a WAR file that is not tailored to a specific web application server, use the `wdeploy buildwarall` command. WAR files created with `wdeploy buildwarall` can be found in: `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/java/applications`.
Simplified deployment	The number of WAR files used for SAP BusinessObjects Enterprise web applications has been reduced. This helps reduce duplicate resource consumption and number of queries made from the web application server to the CMS.

Table 1-2: What's new in this document

Feature	Description
Content reorganization	Rather than list each WDeploy command for every web application server, now a WDeploy command is now shown only once, with examples for each supported web application server.
New features documented	New WDeploy features have been documented.

SAP BusinessObjects Enterprise product documentation is available in supported languages from the support web site, and is refreshed with up-to-date content as it becomes available between releases. For the most recent product documentation, visit help.sap.com.

1.4 Who should read this documentation

This documentation is intended for the system administrator or IT professional working to support an installation of SAP BusinessObjects Enterprise. Familiarity with your overall network environment, port usage conventions, database environment, and web server software is essential.

1.5 SAP BusinessObjects Enterprise guides

The following table provides a list of SAP BusinessObjects Enterprise guides and their contents.

Guide	Description
SAP BusinessObjects Enterprise Planning Guide	Covers the key concepts you should consider before you begin deploying SAP BusinessObjects Enterprise. This guide includes an overview of the architecture, tips for assessing your existing environment, determining your organization's needs, and preparing for the installation.
SAP BusinessObjects Enterprise Installation Guide	Leads you through the steps required to run the installation program and complete your installation of SAP BusinessObjects Enterprise. There are UNIX and Windows versions of this guide available.
SAP BusinessObjects Enterprise Web Application Deployment Guide	Covers topics related to the deployment of web applications to web application servers with SAP BusinessObjects Enterprise. There are UNIX and Windows versions of this guide available.
SAP BusinessObjects Enterprise Administrator Guide	Provides content for server and content administration. The server administration topics includes server configuration, managing authentication, configuring firewalls, and measuring system performance. The content administration topics include working with the CMC, configuring rights and access levels, managing users, and working with SAP Business Objects applications and objects.

Guide	Description
SAP BusinessObjects Enterprise Publisher's Guide	Provides an overview of the publishing process, working with publications, publishing Crystal reports, publishing SAP BusinessObjects Business Intelligence Solution documents, publishing documents, and publishing security.
SAP BusinessObjects Enterprise Upgrade Guide	Information for upgrades from previous versions of SAP BusinessObjects Enterprise to the current version of SAP BusinessObjects Enterprise.
SAP BusinessObjects Enterprise BI launch pad User's Guide	Provides an overview of BI launch pad and working with SAP Crystal Reports, SAP BusinessObjects Interactive Analysis, objects, discussions, encyclopedia, and SAP BusinessObjects Advanced Analysis (formerly Voyager) workspaces.

For a complete list of all of our product documentation please the SAP BusinessObjects support site service.sap.com/bosap-support.

Overview of web application deployment

The SAP BusinessObjects Enterprise installation program can deploy web applications only to the bundled Tomcat 6.0 web application server. All other supported web application servers require that web applications be deployed after the installation is complete. It is recommended that you use the WDeploy web application deployment tool. However, you can deploy web applications with the server's administrative console if you prefer.

Web applications deployed with the web application server's administrative console must first be modified to be deployable WAR or EAR files. The `wdeploy predeploy` and `wdeploy predeployall` commands automate this process. However, if you have extensive knowledge of your web application server and know how to customize web applications for deployment, this process can be done by hand. The manual tailoring of web applications for deployment to a web application server is not covered in this guide.

2.1 Overview of OSGi WAR files

The OSGi framework for Java web applications simplifies the deployment of the web applications bundled with SAP BusinessObjects Enterprise. It allows web applications, language packs, SDKs, plugins, and other resources to exist in a single bundle that can be deployed to a web application server in one step.

Deploying a single WAR file also means fewer web sessions are needed when a user accesses multiple web applications, which reduces the memory, disk, and processing load placed on a web application server.

2.2 Failover and load balancing

SAP BusinessObjects Enterprise supports clustered web application servers with load balancing. Hardware or software load balancers can be used as the entry-point for the web application servers to ensure that the processing is evenly distributed among the web application servers.

The following hardware load balancers are currently supported:

- Cisco Content Services Switches (CSS).
- Cisco Content Switching Modules (CSM).

- The F5 BIG-IP family of load balancers.

The following persistence types are currently supported:

- Source IP address persistence.
- Cookie persistence Insert mode (ArrowPoint Cookie).

Load balancing a cluster of SAP BusinessObjects Enterprise servers is not required, as the Central Management Server (CMS) already distributes work between cluster nodes.

Note:
The Central Management Console (CMC) web application does not support session fail-over. However, BI launch pad is fault-tolerant, and does support session fail-over, so users will not notice if a cluster node fails.

2.2.1 Web application clustering support

The Central Management Console (CMC) and BI launch pad web applications can be used in environments with a variety of different clustered, load balanced, or fault tolerant configurations. The table below lists configuration support for CMC and BI launch pad web applications.

Web application	Clustered web application servers	Load balancers with session affinity	Load balances without session affinity	Fault tolerant
BI launch pad (stateless)	Supported	Supported	Unsupported	Yes
CMC (stateful)	Supported	Supported	Unsupported	No

Note:
The WDeploy tool is not supported for deployment to a cluster.

2.3 SAP BusinessObjects Enterprise WAR and EAR files

The functionality that makes up SAP BusinessObjects Enterprise is divided between several web applications to make it easy to deploy only the components required by your organization. In SAP BusinessObjects Enterprise XI 4.0, many of the core web applications included with previous releases have been bundled into a single OSGi archive. This saves web application server memory and the reduces the number of web sessions needed for web applications that previously had multiple archives.

For example, SAP BusinessObjects Enterprise XI 3.x included the Central Management Console (CMC) and InfoView (now BI launch pad) web applications archived as `CMC.war` and `InfoView.war`. SAP

BusinessObjects Enterprise XI 4.0 has consolidated the CMC and InfoView (now BI launch pad) web applications, along with others, into a single archive named `BOE.war` or `BOE.ear`.

The WDeploy tool is used to automate the process of tailoring web applications to be deployable on a supported web application server. While it is possible for an experienced administrator to manually tailor a web application for a specific web application server, it is recommended that the WDeploy tool be used to automate the process.

The following table lists the web application archives, the web applications that require them. Web applications not automatically deployed by the installation program must be installed post-install.

Web application archive (may be WAR or EAR)	Deployed automatically?	Description
BOE	Yes	OSGi archive of core web applications, including: • Analytical Reporting • CMC • SAP Crystal Reports • BI launch pad (formerly InfoView) • Eclipse IDE support • Lifecycle Manager • Monitoring • OpenDocument • BI workspace (formerly Dashboard Builder) • Platform search • Platform services • Visual difference • SAP Crystal Dashboard Design (formerly Xcelsius)
BusinessProcessBI (deprecated)	Yes	This web application is deprecated. It provides support for legacy Crystal Reports web services and SDK components, including: • Crystal Enterprise • Crystal Reports Report Application Server (RAS) • SAP Crystal Dashboard Design (formerly Xcelsius) • SAP BusinessObjects Advanced Analysis, Web edition (formerly Voyager)
dswsbobje	Yes	Web Services components, including: • Session • BI platform • BI catalog • Federation Administration tool • Live Office • Web service query tool (formerly Query as a Web Service) • Publishing • Report Engine • SAP BusinessObjects Interactive Analysis (formerly Web Intelligence) • SAP Crystal Dashboard Design web services (formerly Xcelsius)
jsfplatform	No	Java Server Faces support and examples.
MobileOTA14	No	Web application for mobile client support.

Web application archive (may be WAR or EAR)	Deployed automatically?	Description
OpenSearch	No	OpenSearch support.
querybuilder	Yes	Query Builder support.
rebean3ws	Yes	Web Services support.

The following table compares the WAR files shipped in previous versions, and where to find the functionality in SAP BusinessObjects Enterprise XI 4.0.

Previous web application archive (may be WAR or EAR)	New web application archive (may be WAR or EAR)
AdminTools	AdminTools
AnalyticalReporting	BOE
BusinessProcessBI	BusinessProcessBI (deprecated)
CmcApp	BOE
CmcAppActions	BOE
CrystalReports	BOE
Dashboard Design	BOE
dswsbobje	dswsbobje
InfoViewApp	BOE
InfoViewAppActions	BOE
LCM	BOE
OpenDocument	BOE
PerformanceManagement	BOE
PlatformServices	BOE
PMC_Help	BOE
VoyagerClient	BOE
XCTemplateUploader	BOE

2.3.1 To deploy MobileOTA14.war for mobile application support

The SAP BusinessObjects Enterprise installation program does not deploy the MobileOTA14 web application that provides support for mobile applications.

To use mobile applications, you must deploy the MobileOTA14 archive (`MobileOTA14.war` or `MobileOTA14.ear`) manually once the installation process is complete.

Note:
The WDeploy GUI tool cannot be used to deploy individual web applications. To deploy an individual web application, such as MobileOTA14, use the WDeploy command-line tool.

1. Before deploying the MobileOTA14 web application, ensure that the WDeploy configuration file `config.<WEB_APPLICATION_SERVER>` has been configured appropriately for your web application server. See WDeploy configuration files.
2. Deploy the MobileOTA14 web application.

   To deploy the MobileOTA14 web application, use the following WDeploy command:

   ```
   wdeploy.sh <WEB_APPLICATION_SERVER>
     -Dwar_dir=<LOCATION_OF_MOBILEOTA14.WAR>
     -DAPP=MobileOTA14
     deploy
   ```

   For example, the following command deploys MobileOTA14 to a WebSphere 7 web application server:

   ```
   wdeploy.sh websphere7
     -Dwar=/sap/sap/businessobjects_xi40/mobile_14/Client
     -DAPP=MobileOTA14
     deploy
   ```

3. Restart the web application server.
4. Access the following URL to ensure that the MobileOTA14 web application is working:

   `http://<HOSTNAME>:<PORT>/MobileOTA14`

   Substitute *<HOSTNAME>* for the web application server hostname, and *<PORT>* for the web application server port number.

For more information on mobile products, refer to the *SAP BusinessObjects Mobile Installation and Deployment Guide*.

2.3.2 To deploy OpenSearch.war for OpenSearch support

The installation program does not deploy the OpenSearch web application that provides support for OpenSearch applications.

To use OpenSearch applications, you must deploy the `OpenSearch.war` archive manually once the installation process is complete.

Note:
The WDeploy GUI tool cannot be used to deploy individual web applications. To deploy an individual web application, such as OpenSearch, use the WDeploy command-line tool.

1. Ensure that web application server's connection details have been set in the WDeploy web application server configuration file. The file is located in:

   `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/config.<WEB_APP_SERVER>`

   For more information on the WDeploy web application server configuration file, see “Configuration files” in the *SAP BusinessObjects Enterprise Web Application Deployment Guide*.

2. Update the OpenSearch configuration in the OpenSearch web application's `config.properties` file. The file is located in:

   `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/warfiles/OpenSearch/WEB-INF`

   Ensure that the following parameters are configured for your server:

   - `CMS`: the CMS hostname and CMS port number. For example, use the format: `<CMS_HOSTNAME>:<PORT>`.
   - `OpenDocURL`: the URL of the OpenDocument web application. For example: `http://<WEB_APP_SERVER>:<PORT>/BOE/OpenDocument/opendoc/openDocument.jsp`.
   - `Proxy.rpurl`: reverse proxy URL, if your organization uses a reverse proxy server.
   - `Proxy.opendoc.rpurl`: the OpenDoc reverse proxy server URL, if your organization uses an OpenDoc reverse proxy server.

3. Deploy the OpenSearch web application.

   To deploy the OpenSearch web application, use the following WDeploy command:

   ```
   wdeploy.sh <WEB_APPLICATION_SERVER>
     -Dapp_source_dir=<LOCATION_OF_OPENSEARCH_WEB_APP_SOURCE_TREE>
     -DAPP=OpenSearch
     deploy
   ```

   For example, the following command deploys the OpenSearch web application to a WebSphere 7 web application server:

   ```
   wdeploy.sh websphere7
     -Dwar=/sap/sap_bobj/enterprise_xi40/warfiles/OpenSearch
     -DAPP=OpenSearch
     deploy
   ```

For more information on OpenSearch related products, refer to “OpenSearch” section of the *SAP BusinessObjects Enterprise Administrator Guide*.

2.4 Default context roots

All web applications can be deployed to a custom context root on your web application server. The following table lists the context roots for each web application.

Web application	Context path
Central Management Console (CMC)	/BOE/CMC
BI launch pad	/BOE/BI
Open Document	/BOE/OpenDocument
Crystal Reports BI Processes (deprecated)	/BusinessProcessBI
Web Services	/dswsbobje

2.5 Custom root contexts and context paths

SAP BusinessObjects Enterprise web applications can be deployed to a custom location on a web application server. This location is reflected in the URL used to access the web application from a web browser, and is often known as the “context”.

A root context is the top-level folder on a web application server in which web applications are located. The default root context for SAP BusinessObjects Enterprise web applications is /BOE. For example, on a web application server named www.mycompany.com, the URL prefix used to access web applications on the server would be http://www.mycompany.com/BOE/.

A context path (sometimes referred to as a virtual directory) is a folder within a root context, in which a web application is located. For example, the default context path for the BI launch pad application is /BI. The URL used to access the BI launch pad web application on a web application server named www.mycompany.com would be http://www.mycompany.com/BOE/BI.

Both the root context and the context path can be changed to suit the needs of your organization. The following table lists examples of deploying a web application named MyApp to different root and web application context paths. The following topics describe how to customize root and web application context paths.

2.5.1 To change the root context

You can change the root context used by SAP BusinessObjects Enterprise web applications. The default setting is to have an empty root context, so the web application context path is shown directly after the server address in a URL.

For example, an empty root context results in a URL such as http://localhost:8080/BOE/CMC, where http://localhost:8080/ is the server and port number, there is no root context, and BOE/CMC is the web application context path. Setting the root context to *MY_COMPANY* would change the URL example shown above to http://localhost:8080/*MY_COMPANY*/BOE/CMC.

When using the WDeploy tool, the root context can be set in the "Options" screen. When using the WDeploy command-line tools, the root context for SAP BusinessObjects Enterprise web applications is set in the following configuration file:

`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/wdeploy.conf`

Use a text editor to update the value for `root_context_path` given in `wdeploy.conf`.

For example, `wdeploy.conf` configuration file contains the following parameters by default:

```
as_lang=en
work_dir=
war_dir=
app_source_tree=
disable_CMC=false
disable_InfoView=false
JCoStandalone=
root_context_path=
recent_app_svr=<WEB_APP_SERVER>
```

2.5.2 To change a web application's context path

You can change the default context path (sometimes referred to as the virtual directory) of SAP BusinessObjects Enterprise web applications.

For example, the BOE web application's default context path is `BOE`, which can be seen in the sample URL `http://localhost:8080/BOE/CMC`. In this example, `http://localhost:8080/` is the server and port number, `BOE` is the web application, and `/CMC` is a component included within the BOE web application.

The context path for SAP BusinessObjects Enterprise web applications is set in the following configuration file:

`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/apps/<WEB_APP>.properties`

Use a text editor to update the value of the `appvdir` parameter found in configuration file `<WEB_APP>.properties`.

For example, the `BOE.properties` configuration file contains the following parameters by default:

```
warfile=BOE.war
appvdir=BOE
buildfile=BOE.xml
osgisupported=true
```

To deploy web applications with the WDeploy tool

3.1 Overview of WDeploy

The WDeploy tool is included with SAP BusinessObjects Enterprise to ease deployment of web applications to Java-based web application servers.

There are two different versions of WDeploy:

- A traditional, text-based interface that processes commands and parameters given on the command-line.
- A new console-based guided assistant similar to the SAP BusinessObjects Enterprise installation program, which prompts the user to enter deployment parameters.

While each supported web application server requires different commands and web application package updates, WDeploy provides a consistent interface for administrators, and automates the adjustments needed for deployment to a specific web application server.

For example, to deploy a web application to an SAP NetWeaver AS Java web application server, a single WDeploy command performs the following tasks:

1. Creates settings specific to SAP NetWeaver in the web application's `web.xml` file.
2. Bundles the web application content into a web archive.
3. Creates an SCA package containing the web application.
4. Copies the SCA package to the SAP NetWeaver deployment directory.
5. Calls SAP NetWeaver deployment tools to deploy the web application.

3.2 WDeploy prerequisites

This section details prerequisites for the deployment of SAP BusinessObjects Enterprise web applications to supported web application servers.

Note:
Before deploying web applications to WebSphere, see WASX7017E: Exception deploying in WebSphere.

For more information on WDeploy GUI prerequisites, see WDeploy GUI tool prerequisites.

3.2.1 Before you deploy web applications

Your web application server must be installed and working before you attempt to install SAP BusinessObjects Enterprise. Consult your web application server documentation for installation instructions.

Your web application server should have at least 5 GB of free disk space, in addition to any other requirements given by other software installed on the machine.

It is recommended that you change the heap size and maximum perm size settings of your JVM to -Xms128m -Xmx1024m -XX:MaxPermSize=512m. If using Tomcat for example, your modified settings would look like this:

```
JAVA_OPTS="-Xms128m -Xmx1024m -XX:MaxPermSize=512m"
```

Note:
For SAP NetWeaver AS Java 7.2, ensure that the maximum heap size is at least 2048 megabytes. For example: `JAVA_OPTS="-Xms128m -Xmx2048m -XX:MaxPermSize=512m"`

Consult your JVM documentation for information on changing your Java memory settings.

Before you begin the deployment process, ensure that the web application server is installed and verify that the application server is running correctly by launching its administrative console.

3.2.2 Hardware requirements

The deployment of web applications to a web application server with the WDeploy tool requires at least 4 GB of RAM (8 GB for SAP NetWeaver AS Java 7.2), and 15 GB of free disk space, plus a minimum 5 GB of free space on the partition that hosts the `/tmp` temporary folder, for the deployment of web applications. This is in addition to any other requirements of the web application server or any other servers or services installed on the host.

3.2.3 64-bit support

SAP BusinessObjects Enterprise is only supported on 64-bit operating systems and only supports 64-bit web application servers with a 64-bit JDK.

3.2.4 Minimum rights for deploying web applications

Tip:
We recommend that you use the same user account for installing BusinessObjects Enterprise and your web application server.

To deploy web applications using a user account different from the one used to install the web application server, you must ensure that the BusinessObjects Enterprise user account has the privileges listed in the following table.

Note:
In addition to the right to execute files or folders mentioned in the tables below, the right to execute is also required on the parent directory.

JBoss 4.2.3 and JBoss 5

Directory	Recursive	User	Group	Other	Example command
JBoss user account home and BusinessObjects Enterprise user folder subdirectory	No	rwx	r-x	r-x	chmod 755 ~<JBOSS_USER>
Web application deployment directory (`$as_dir/server/$as_instance/deploy`)	Yes	rwx	rwx	rwx	chmod -R 777 <JBOSS_HOME>/ server/default/ deploy
JBoss web application SAR (`$as_dir/server/$as_instance/deploy/jbossweb-tomcat55.sar`)	Yes	rwx	rwx	rwx	chmod -R 777 <JBOSS_HOME>/ server/default/ deploy/ jboss-tomcat55.sar
JBoss binaries (`$as_dir/bin`)	No	rwx	r-x	r-x	chmod 755 <JBOSS_HOME>/ bin
JBoss libraries (`$as_dir/lib`)	No	rwx	r-x	r-x	chmod 755 <JBOSS_HOME>/ lib

Oracle Application Server 10g R3

Directory	Recursive	User	Group	Other	Example command
Oracle application server user account home and BusinessObjects Enterprise user folder subdirectory	No	rwx	r-x	r-x	chmod 755 ~<ORA_USER>
Web application deployment directory (`$as_dir`)	No	rwx	r-x	r-x	chmod 755 <ORA_HOME>
Oracle AS J2EE directory (`$as_dir/j2ee`)	No	rwx	r-x	r-x	chmod 755 <ORA_HOME>/ j2ee
Oracle AS J2EE/home directory (`$as_dir/j2ee/home`)	Yes	rwx	r-x	r-x	chmod -R 755 <ORA_HOME>/ j2ee/home

Directory	Recursive	User	Group	Other	Example command
Oracle AS opmn directory(`$as_dir/opmn`)	No	rwx	r-x	r-x	chmod 755 <ORA_HOME>/opmn
Oracle AS opmn configuration directory (`$as_dir/opmn/conf`)	Yes	rwx	rwx	rwx	chmod -R 777 <ORA_HOME>/opmn/conf
Oracle AS opmn library directory (`$as_dir/opmn/lib`)	Yes	rwx	r-x	r-x	chmod -R 755 <ORA_HOME>/opmn/lib

SAP NetWeaver AS Java 7.2

Note:
If SAP is installed in the folder hierarchy like `/netweaver/sap`, then the rights outlined below must be given to both the SAP and NetWeaver directories.

Directory	Recursive	User	Group	Other	Example command
SAP application server account user account home and BusinessObjects Enterprise user folder subdirectory	No	rwx	r-x	r-x	chmod 755 ~<SAPAS_USER>
SAP AS installation directory	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>
SAP AS directory (`$as_dir`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap
SAP AS SID directory (`$as_dir/$as_sid`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap/CTB
SAP AS instance directory (`$as_dir/$as_sid/$as_instance`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap/CTB/JC00
SAP AS J2EE directory (`$as_dir/$as_sid/$as_instance/j2ee`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap/CTB/JC00/j2ee
J2EE deploying subdirectory (`$as_dir/$as_sid/$as_instance/j2ee/deploying`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap/CTB/JC00/j2ee/ deploying

Directory	Recursive	User	Group	Other	Example command
J2EE cluster subdirectory (`$as_dir/$as_sid/$as_instance/j2ee/cluster`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/sap/CTB/JC00/j2ee/ cluster

Directory	Recursive	User	Group	Other	Example command
J2EE cluster server subdirectory (`$as_dir/$as_sid/$as_instance/j2ee/cluster/server0`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap/CTB/JC00/j2ee/ cluster/server0
J2EE cluster server subdirectory (`$as_dir/$as_sid/$as_instance/j2ee/cluster/server0/apps`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap/CTB/JC00/j2ee/ cluster/server0/ apps
J2EE cluster domain subdirectory (`$as_dir/$as_sid/$as_instance/j2ee/cluster/server0/apps/sap.com`)	No	rwx	rwx	rwx	chmod 777 <SAPAS_HOME>/ sap/CTB/JC00/j2ee/ cluster/server0/ apps/sap.com

Tomcat 5.5 and Tomcat 6

Directory	Recursive	User	Group	Other	Example command
Tomcat user account home and BusinessObjects Enterprise user folder subdirectory	No	rwx	r-x	r-x	chmod 755 ~<TOMCAT_USER>
Web application deployment directory (`$as_dir/webapps`)	No	rwx	rwx	rwx	chmod 777 <TC_HOME>/ webapps
Tomcat binary directory (`$as_dir/bin`)	No	rwx	rwx	rwx	chmod 777 <TC_HOME>/ bin
Tomcat configuration directory (`$as_dir/conf`)	Yes	rwx	rwx	rwx	chmod -R 777 <TC_HOME>/ conf
Tomcat library directory (`$as_dir/lib`)	No	rwx	r-x	r-x	chmod 755 <TC_HOME>/ lib/*

WebLogic 9.2 MP2, WebLogic 10, and WebLogic 10.3

Directory	Recursive	User	Group	Other	Example command
	No	rwx	r-x	r-x	chmod 755 <WL_USER>

Directory	Recursive	User	Group	Other	Example command
WebLogic user account home and BusinessObjects Enterprise user folder subdirectory					
Web domain binaries (`$WL_HOME/<PROJECT>/<DOMAIN>/<BASE>/bin`)	No	rwx	rwx	rwx	chmod 777 <WL_HOME>/ user_projects/ domains/ base_domain/bin
WebLogic server libraries (`$WL_HOME/server/lib`)	No	rwx	r-x	r-x	chmod 755 <WL_SERVER>/ user_projects /lib
WebLogic server binaries (`$WL_HOME/server/domains/base_domain common/bin`)	No	rwx	rwx	r-x	chmod 775 <WL_SERVER>/ user_projects /bin
WebLogic server modules (`$WL_HOME/server/domains/base_domain modules`)	No	rwx	rwx	r-x	chmod 775 <WL_SERVER>/ modules
WebLogic Java (`$WL_HOME/jdk150_06`)	No	rwx	rwx	r-x	chmod 775 <WL_SERVER>/ jdk150_06

WebSphere 6.1 and WebSphere 7.0

Directory	Recursive	User	Group	Other	Example command
WebSphere user account home and BusinessObjects Enterprise user folder subdirectory	No	rwx	r-x	r-x	chmod 755 ~<WEBSPHERE_USER>
WebSphere profiles directory (`$WEBS_HOME/AppServer/profiles`)	Yes	rwx	rwx	rwx	chmod -R 777 <WEBSPHERE_HOME>/ AppServer/profiles
WebSphere admin script (`$as_dir/bin/wsadmin.sh`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ bin/wsadmin.sh
WebSphere plugin script (`$as_dir/bin/GenPluginCfg.sh`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ bin/GenPluginCfg.sh
WebSphere security (`$as_dir/bin/securityProcs.jacl`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ bin/securityProcs.jacl

Directory	Recursive	User	Group	Other	Example command
WebSphere security (`$as_dir/bin/LTPA_LDAPsecurityProcs.jacl`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ bin/LTPA_LDAPsecurityProcs.jacl
WebSphere plugins directory (`$as_dir/plugins`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ plugins
WebSphere Java directory (`$as_dir/java`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ java
WebSphere deployment tool directory (`$as_dir/deploytool`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ deploytool
WebSphere properties directory (`$as_dir/properties`)	No	rwx	r-x	r-x	chmod 755 <WEBSPHERE_HOME>/ properties

Note:

You must accept the WebSphere web application server SSL certificate before you can use WDeploy. To accept the certificate, use `wsadmin` command in the format <WEBSPHERE_HOME>/AppServer/bin/wsadmin -conntype SOAP -port <ADMIN_PORT> -user <AS_ADMIN_USERNAME> -password <AS_ADMIN_PASSWORD>. For example:

./wsadmin -conntype SOAP -port 8880 -user administrator -password websphere

Sun Java System Application Server

Directory	Recursive	User	Group	Other	Example command
Sun Java System Application Server user account home and BusinessObjects Enterprise user folder subdirectory	No	rwx	r-x	r-x	chmod 755 ~<SO_USER>
Sun Java System Application Server (`$as_dir/domains/$as_domain/config`)	Yes	rwx	rwx	rwx	chmod -R 777 <SO_HOME>/ domains/d1/config

Note:

You must accept the Sun One web application server SSL certificate before you can use WDeploy. To accept the certificate, use `wsadmin` command in the format <SO_HOME>/bin/asadmin get --port <ADMIN_PORT> --user admin --secure '*'. For example:

```
./asadmin get --port 4849
    --user admin --secure '*'
```

3.2.5 Minimum 4 GB in /tmp

Ensure that the host operating system has a minimum of 4 GB of disk space of temporary space for the deployment of web applications. For example, if temporary files are written to a `/tmp` partition, `/tmp` must be larger that 4 GB.

3.2.6 Set ulimit to unlimited

To successfully build and deploy SAP BusinessObjects Enterprise web applications, the host operating system or user account `ulimit` setting must be configured as *unlimited*.

Set the `ulimit` configuration with the `ulimit` command, or modify the system configuration file `/etc/security/limits.conf`. For more information about the `ulimit` setting, see the documentation included with your operating system.

3.2.7 To set up the Java environment

WDeploy requires a Java Virtual Machine to be available on the host system. Java Development Kit (JDK) 1.6 is installed automatically by the SAP BusinessObjects Enterprise installation program, but must be set up or copied from the SAP BusinessObjects Enterprise server when manually installed on a dedicated machine.

The JDK installed by the SAP BusinessObjects Enterprise installation program is used by default. If you are using a dedicated web application server, you must set up the JDK by performing either a Web Tier installation, or manually installing an appropriate JDK (1.5 or 1.6, as supported by the web application server). When setting up a JDK manually, ensure that the following environment settings have been configured:

- The *JAVA_HOME* environment variable has been set to the Java directory. If the *JAVA_HOME* variable is not set, or is invalid, WDeploy attempts to use the JVM installed with SAP BusinessObjects Enterprise. If no suitable JVM is found, WDeploy exits.
- The user account `PATH` environment variable includes:

  `<JAVA_HOME>/bin`
- To allow WDeploy to run from any directory, update the `PATH` environment variable to include:

  `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy`

3.2.8 Web application server parameter configuration files

Before using the WDeploy command, ensure that the correct parameters are set in the WDeploy web application deployment server parameter configuration file. Set parameters are used as default settings, and no longer need to be given on the command-line.

The WDeploy web application server parameter configuration file appropriate for your web application server in the folder `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf`. For example, if you are using SAP NetWeaver 7.2, select `/opt/sap/sap_bobj/enterprise_xi40/wdeploy/conf/config.sapappsvr72`.

Use a text editor to update values specific to your organization's web application server. For more information on how to configure WDeploy configuration files, see WDeploy configuration files.

Parameters set in the WDeploy web application server parameter configuration file can be overridden when calling WDeploy from the command-line, using switches and parameters to change the default behavior. However, it is highly recommended that you configure the parameters in the `config.<WEB_APP_SERVER>` configuration file appropriate for your deployment.

3.2.9 To install WDeploy on remote web application server

To deploy web applications to a dedicated web application server, perform a "Web Tier" or "Custom / Expand" installation with the SAP BusinessObjects Enterprise installation program. If you prefer to copy the WDeploy tool manually, copy the following folder to the web application server:

`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy`

Note:
When copying the WDeploy tool manually, ensure that the environment variable `JAVA_HOME` is configured appropriately on the web application server.

3.3 SAP System Landscape Directory (SLD) registration

Your organization may use the SAP System Landscape Directory (SLD) Data Supplier (DS) to maintain a central repository of information about your organization's SAP software. It provides administrators with detailed information about the system's topology and software components. For more information on SLD, see “Registration of SAP BusinessObjects Enterprise in the System Landscape” in the *SAP BusinessObjects Enterprise Administrator Guide*. For more general information, see service.sap.com/sld.

The SLD registration notifies the SLD when web applications are deployed or undeployed, keeping the SLD database current with the latest information about your organization's web application deployments.

Web application components can be registered with SLD on the following web application servers:

- SAP NetWeaver 7.2
- IBM WebSphere 6.1 and 7

3.3.1 To enable SLD registration for SAP NetWeaver

SAP System Landscape Directory Data Supplier (SLD-DS) integration is available for SAP NetWeaver. For information on SLD-DS integration and NetWeaver, see *Using the System Landscape Directory* at: http://www.sdn.sap.com/irj/sdn/nw-sld#section18.

For more information on SLD, see *Configuring, working with, and administering System Landscape in SAP NetWeaver 7.1* at: http://www.sdn.sap.com/irj/sdn/nw-sld?rid=/library/uuid/b00462bb-b7a0-2a10-8da6-f6b89834f8b4.

For more information on working with SLD, see *System Landscape Directory: SAP NetWeaver 7.0 SPS12* at: http://www.sdn.sap.com/irj/sdn/nw-sld?rid=/library/uuid/50924080-0627-2a10-2b89-8cb87b972fb1

3.3.2 To enable SLD registration for WebSphere

To use SAP System Landscape Directory Data Supplier (SLD-DS) with WebSphere 6.1 or 7.0, the SLDREG registration tool must be installed on each WebSphere web application server.

Note:
SLDREG is not installed as a part of SAP BusinessObjects Enterprise. For information on installing SLDREG, refer to SAP Note 1018839.

To configure SLDREG so that the SLD is updated whenever web applications are deployed or undeployed from a WebSphere web application server, refer to SAP Note 1482727.

3.4 Deployment modes

WDeploy supports two different deployment modes:

1. Standalone mode (a web application server)

The web application server serves both static content (HTML pages, images, documents, JavaScript, Cascading Style Sheets) and dynamic content (Java Server Pages, JAR files, XML files).

2. Split web tier mode (a web application server plus a dedicated web server)

   A dedicated web server receives requests from web browsers and serves all static content (HTML pages, images, documents, JavaScript, Cascading Style Sheets). Requests for dynamic content (Java Server Pages, JAR files, XML files) are forwarded to the dedicated application server and returned to the web browser when the content has been formed.

   This mode is suited to larger production deployments where scalability and performance are key.

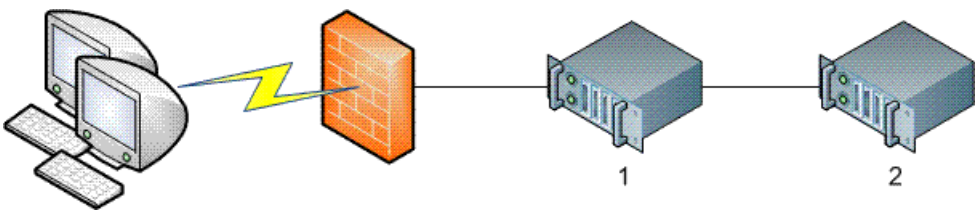
   Note:
   The following web application are not supported in split web tier mode deployments:
   - Web Services
   - MobileOTA14
   - OpenSearch
   - jsfplatform

It is also possible to use WDeploy to deploy web applications to a web application server installed on the same system as SAP BusinessObjects Enterprise. This configuration can be used for small development or test systems, and is not recommended for production systems.

3.4.1 Standalone deployments

Standalone mode refers to a web application server serving both static and dynamic content to web clients. The web application server could run on the same machine as SAP BusinessObjects Enterprise, or on a separate machine connected by network.

In the following diagram, web clients connect through a firewall to a web application server [1] that serves both static and dynamic content. Processing requests from the web application server are sent to the SAP BusinessObjects Enterprise Central Management Server (CMS) [2].



In the diagram above, the WDeploy is installed as a component of the CMS [2], and web applications are separated out into directories for static and dynamic content. The dynamic content can now be copied to the web application server [1].

The WDeploy command can also be installed on dedicated web application server [1], making it easy to deploy separated content received from a CMS [2].

This mode is best suited for small deployments with a limited number of users. The advantage of a standalone deployment is that it is easy to deploy and maintain, but it may not scale to a large number of users because the web application server delivers both static and dynamic content.

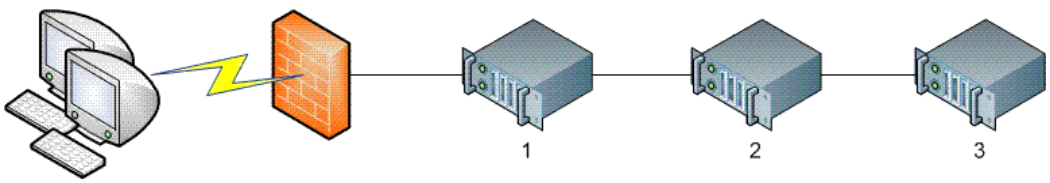
3.4.2 Split web tier deployments

A web tier deployment separates static and dynamic web application content so that static content is served by a web server, and dynamic content is served by a web application server. The web and web application servers could run on the same machine, or separate machines connected to a network.

Note:
The following web application are not supported in split web tier mode deployments

- Web Services
- MobileOTA14
- OpenSearch
- jsfplatform

In the following diagram, web clients connect through a firewall to a web server [1] that serves only static content (HTML pages, images, documents, JavaScript, Cascading Style Sheets). When dynamic content is required, the web server sends a request to the web application server [2]. Any requests that require further processing by SAP BusinessObjects Enterprise are sent to Central Management Server (CMS) for processing [3].



In the diagram above, the WDeploy is installed as a component of the CMS [3], and web applications are separated out into directories for static and dynamic content. The dynamic content can now be copied to the web application server [2], and the static content copied to the web server [1].

The WDeploy command can also be installed on dedicated web [1] and web application [2] servers, making it easy to deploy separated content received from a CMS [3].

If an organization implements security measures that restrict access to server machines, the separated static and dynamic content can be sent separately to those with the authority to deploy content.

Split deployments are best suited to mid or large-sized deployments with a large or increasing number of users. The advantage of a split deployment is that it is scalable and provides good performance.

Note:
Web, web application, and SAP BusinessObjects Enterprise servers can be clustered to provide an even greater degree of scalability, availability, and performance.

3.4.2.1 Supported dedicated web and web application server combinations

WDeploy supports the following web and web application server configurations for split deployments:

- Apache 2.2 web server with Tomcat 5.5 or 6 (6.0.20 or newer)
- Apache 2.2 web server with WebLogic 9.2 MP2 or 10.x
- IBM IHS web server 6 with WebSphere 6.1 (6.1.0.7 or newer)
- IBM IHS web server 7 with WebSphere 7.0 (6.0.0.11 or newer)
- Sun Java System Web server 7 with Sun Java System Application Server 9.1

Note:
Apache, and IBM IHS web servers are all given as `apache` in WDeploy configuration files.

3.5 WDeploy configuration files

The WDeploy configuration files contain settings saved in a key-value pair text format. Read by WDeploy when it starts, the options and parameters saved in the configuration files are used as default settings and no longer need to be given on the command-line. However, all options and parameters can still be given on the command-line, which overrides the configuration file. Options and parameters given on the command-line do not change the options stored in configuration files.

Note:
It is recommended that web application server administrator account passwords are not stored in the WDeploy configuration file, but rather passed to WDeploy from the command-line with the parameter `-Das_admin_password=<PASSWORD>`.

There are three configuration files used by WDeploy:

- WDeploy configuration file (`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/wdeploy.conf`): stores general WDeploy settings that apply to all web applications severs.
- Web application server configuration files (`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/config.<WEB_APP_SERVER>`): stores settings for a specific web application server.
- Web application configuration file (`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/apps/<WEB_APP>.properties`): stores individual deployment settings for each web application.

3.5.1 WDeploy configuration file

The <*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/conf/wdeploy.conf configuration file stores settings likely to be shared among any web application servers in your deployment.

Properties stored in the WDeploy configuration file can be overridden from the command-line using the -D<*PROPERTY*> switch parameter. However, it is highly recommended that you configure the parameters in the config.<*WEB_APP_SERVER*> configuration file appropriate for your deployment.

For a complete list of WDeploy properties, see WDeploy property parameters.

3.5.2 Web application server configuration files

As each web server and web application server requires different WDeploy settings, the WDeploy tool references a configuration file for each supported server. The configuration files are stored in the WDeploy conf folder as follows:

<*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/conf/config.<*WEB_APP_SERVER*>

Substitute <*WEB_APP_SERVER*> for the name of the web application server to which the WDeploy tool deploys.

The following list shows all configuration files for currently supported web application servers:

- config.jboss4 (JBoss 4.2.3)
- config.jboss5
- config.oas1013 (Oracle Application Server 10g R3)
- config.sapappsvr72 (SAP NetWeaver AS Java 7.2)
- config.sunone9 (Sun Java System Application Server 9.1)
- config.tomcat55 (Tomcat 5.5.20)
- config.tomcat6 (Tomcat 6.0.20)
- config.weblogic9 (WebLogic 9.2 MP2)
- config.weblogic10 (WebLogic 10 and WebLogic 10.3)
- config.websphere6 (WebSphere 6.1.0.7)
- config.websphere7 (WebSphere 7.0.0.11)

The following sections list the configuration options available for each supported web and web application server.

3.5.2.1 JBoss 4.2.3 or 5.0 configuration file

Set default values for the following parameters in `config.jboss4` (JBoss 4.2) or `config.jboss5` (JBoss 5.0) to avoid having to give them on the command-line every time.

Table 3-2: Mandatory WDeploy parameters for JBoss 4.2.3 or 5.0

Parameter name	Description	Example
as_dir	Installed location of the JBoss web application server (*JBOSS_HOME*).	/opt/jboss-4 or /opt/jboss-5
as_instance	Name of the JBoss application server instance.	default

3.5.2.2 Oracle Application Server 10g R3 configuration file

Set default values for the following parameters in `config.oas1013` to avoid having to give them on the command-line every time.

Table 3-3: Mandatory WDeploy parameters for Oracle Application Server 10g R3

Parameter name	Description	Example
as_admin_port	Oracle Application Server administration request port (notification server port, as found in the `<notification-server>` section of `opmn.xml`.	6003
as_admin_username	Oracle Application Server administrative account username.	oc4jadmin
as_admin_password	Oracle Application Server administrative account password.	password
as_instance	Oracle Application Server application server instance name.	home

Parameter name	Description	Example
as_admin_is_secure	Switch to tell WDeploy that Oracle Application Server SSL encryption is enabled. **Note:** If as_admin_is_secure is not enabled, the username and password must still be supplied.	false
as_dir	Installed location of Oracle Application Server.	/opt/product/10.1.3.1/OracleAS_1
as_appserver_name	Target application server hostname.	oracle10g.mycompany.com
as_group_id	Server group to which the target web application server belongs.	Default_group

3.5.2.3 SAP NetWeaver AS Java 7.2 configuration file

Set default values for the following parameters in `config.sapappsvr72` to avoid having to give them on the command-line every time.

Table 3-4: Mandatory WDeploy parameters for SAP NetWeaver AS Java 7.2

Parameter name	Description	Example
as_dir	Installed location of SAP NetWeaver AS Java.	/opt/sap
as_sid	System ID of the target instance.	AS1
as_instance	Application server instance name.	JC01
as_admin_port	SAP NetWeaver AS Java administration request port.	50004
as_admin_username	SAP NetWeaver AS Java administrative account username.	administrator

Parameter name	Description	Example
as_admin_password	SAP NetWeaver AS Java administrative account password.	password
clear.temp.dirs	Passed to SAP NetWeaver AS Java during deployment: toggles the automatic removal of temporary files. Should be set to "true".	true
use.deploy.controller	Passed to SAP NetWeaver AS Java during deployment: toggles use of the SAP Deploy Controller tool. Should be set to "true".	true
time.to.wait.factor	Passed to SAP NetWeaver AS Java during deployment: wait time in seconds for the WDeploy script to wait for the web application server to complete deployment (for example "1600").	1600

3.5.2.4 Sun Java System Application Server 9.1 configuration file

Set default values for the following parameters in `config.sunone9` to avoid having to give them on the command-line every time.

Table 3-5: Mandatory WDeploy parameters for Sun Java System Application Server 9.1

Parameter name	Description	Example
as_dir	Installed location of Sun Java System Application Server.	/opt/SUNWappserver82e
as_instance	Name of the Sun Java System Application server instance.	server
as_domain	Domain to which the application server is attached.	domain1
as_admin_port	Java System Application Server administration request port.	4849

Parameter name	Description	Example
as_admin_username	Sun Java System Application Server administrative account username.	admin
as_admin_password	Sun Java System Application Server administrative account username.	password
as_admin_is_secure	Switch to tell WDeploy that Sun Java System Application Server SSL encryption is enabled. **Note:** If as_admin_is_secure is not enabled, the username and password must still be supplied.	true

3.5.2.5 Tomcat 5.5 or 6 configuration file

Set default values for the following parameters in `config.tomcat55` or `config.tomcat6` to avoid having to give them on the command-line every time.

When deploying to a split environment, where a Tomcat web application server is paired with a dedicated Apache web server, see To deploy to separate Apache web and Tomcat web application servers.

Table 3-6: Mandatory WDeploy parameters for Tomcat 5.5 or 6

Parameter name	Description	Example
as_instance	Application server instance name.	localhost
as_service_name	Name of the Tomcat service when the application server is installed as a Windows service (only applicable for installations on Microsoft Windows).	Tomcat55 or Tomcat6
as_dir	Installed location of Tomcat 5.5 or 6.	/opt/tomcat

Parameter name	Description	Example
as_service_key	Registry key used by Tomcat to give Java parameters (only applicable for installations on Microsoft Windows).	HKLM\SOFTWARE\Wow6432Node\Apache Software Foundation\Procrun 2.0\$<as_service_name>\Parameters\Java
as_service_key_value	Tomcat's Java parameters: the value of the registry key as_server_key.	Options.

3.5.2.6 WebLogic 9, 10, or 10.3 configuration file

Set default values for the following parameters in `config.weblogic9` or `config.weblogic10` to avoid having to give them on the command-line every time.

Table 3-7: Mandatory WDeploy parameters for WebLogic 9, 10, and 10.3

Parameter name	Description	Example
as_admin_port	WebLogic administration request port.	7001
as_admin_username	WebLogic administrative account username.	weblogic
as_admin_password	WebLogic administrative account password.	password
as_instance	Name of the WebLogic application server instance.	AdminServer
as_domain_dir	WebLogic domain directory.	/opt/bea/weblogic/user_projects/domains/base_domain

3.5.2.7 WebSphere 6 or 7 configuration file

Set default values for the following parameters in `config.websphere6` or `config.websphere7` to avoid having to give them on the command-line every time.

Table 3-8: Mandatory WDeploy parameters for WebSphere 6 or 7

Parameter name	Description	Example value
as_soap_port	Port number for SOAP application server administration. If not set, the default SOAP port number will be used.	8880
as_instance	The name of the WebSphere application server instance.	server1
as_admin_password	WebSphere administrative account password.	password
as_admin_username	WebSphere administrative account username.	administrator
as_profile_name	Name of the profile created for Websphere Application Server. Give this parameter when a non-default profile is used for the deployment. **Note:** This parameter is currently only available when using the WDeploy command-line tool. It cannot be used with the WDeploy GUI tool.	AppSrv01
as_virtual_host	Virtual host to which the application must be bound.	default_host

Parameter name	Description	Example value
as_admin_is_secure	Instructs WDeploy that WebSphere security is enabled. **Note:** Values for as_admin_username and as_admin_password must be set when as_admin_is_secure is true").	false
as_dir	Installed location of WebSphere.	/opt/IBM/WebSphere/AppServer
ws_instance	Web server instance when deployed to a split environment (dedicated web server).	webserver1
enforce_file_limit	Indicates to WDeploy whether or not the web application server may encounter issues loading applications that contain more than 65,535 files (false by default).	false
as_plugin_cfg_dir	Location of the WebSphere `plugin-cfg.xml` file. This parameter is only required for split web tier deployments.	/opt/AppServers/IBMWebSphere7/profiles/AppSrv01/config/cells/Wdeploy2k364BNode01Cell/nodes/Wdeploy2k364BNode01/servers/webserver1

3.5.2.8 Dedicated web servers in split deployments

When deploying web applications to a dedicated web server in split deployments, use the name of the supported web server:

- `config.apache` (Apache Web Server or IBM IHS)
- `config.sunwebsvr7` (Sun Java System Web Server 7)

Note:
It is recommended that web application server administrator account passwords are not stored in the WDeploy configuration file, but rather passed to WDeploy from the command-line with the parameter `-Das_admin_password=<PASSWORD>`.

Split deployments employ a dedicated web application server to serve dynamic content, and a dedicated web server to serve static content. The following table lists the web application servers that can be configured for a split deployment of SAP BusinessObjects Enterprise.

Application Server	Apache web server	Sun ONE web server
SAP NetWeaver 7.2	N/A	N/A
Tomcat 6.0	Yes	No
WebLogic 10	Yes	No
WebLogic 10.3	Yes	No
WebSphere 6.1	Yes (IHS 6)	No
WebSphere 7	Yes (IHS 7)	No
JBoss 4.2	No	No
JBoss 5.0	No	No
Sun Java Application Server 9.1	No	Yes
Oracle Application Server 10g R3	No	No
Tomcat 5.5	Yes	No
WebLogic 9.2 MP2	Yes	No

3.5.2.8.1 Apache or IBM IHS (split deployment) configuration file

Set default values for the following parameters in `config.apache` to avoid having to give them on the command-line every time.

Note:
The same configuration file (config.apache) is used for Apache 2.2 or IBM IHS.

Table 3-9: Mandatory WDeploy parameters for Apache 2.2 or IBM IHS (split deployment)

Parameter name	Description	Example
connector_host	The hostname of the web application server.	localhost
connector_port	The port number of the web application server.	8080

Parameter name	Description	Example
connector_type	The Apache connector type used to configure split mode	tomcat6
deployment_dir	Directory under which `WDeploy` creates a subdirectory for static content served by the web server. This name must match the application name, and the name of the zip archive that contains the application's static resources. `WDeploy` creates a virtual directory on the web server, mapping the subdirectory to the URL	/opt/apache2/htdocs
ws_dir	The Apache web server installation directory.	/opt/apache2

Note:
When using IBM IHS with a WebSphere 6 or WebSphere 7 web application server, ensure that as_plugin_cfg_dir is correctly configured in `config.websphere6` or `config.websphere7`.

3.5.2.8.2 Sun Java System Web Server 7 (sunwebsvr7) configuration file

Set default values for the following parameters in `config.sunwebsvr7` to avoid having to give them on the command-line every time.

Table 3-10: Mandatory WDeploy parameters for Mandatory WDeploy parameters for Sun Java System Web Server 7 (sunwebsvr7)

Parameter name	Description	Example
ws_dir	The web server installation directory that contains the config and doc subdirectories.	/opt/sunwebsvr7
ws_config_dir	The sunwebsvr7 config directory. By default: `$WS_DIR/config`	/opt/sunwebsvr7/config
connector_name	The name of the object declaring the connector service in `obj.conf`	passthrough

Parameter name	Description	Example
connector_type	The connector type used to configure split mode.	sunone9
deployment_dir	The document root. By default: `$WS_DIR/docs`. The path must be an absolute path	/opt/sunwebsvr7/htdocs

3.5.3 Web application configuration property file

Each web application can be deployed to a different location on the web application server, and can be configured with different packaging options.

Each deployable web application is configured in the configuration file `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/app/<WEB_APP>.properties`.

The following settings are available in a `<WEB_APP>.properties` configuration file:

Variable	Description	Example
`warfile`	Name of the WAR file to create for this web application.	`dswsbobje.war`
`appvdir`	Name of the context path (also known as a virtual directory) to which the web application is deployed.	`dswsbobje`
`buildfile`	Ant build file used to build the web application.	`dswsbobje.xml`
`deploy_as_a_file tree`	Toggle to enable the deployment of an exploded WAR file when true.	`true`
`classloading_mode`	Setting to determine the Classloader order.	`PARENT_LAST`
`classloader_package_filtering`	Filter for classloader.	`com.crystalx.*,javax.xml.*,org.apache.*`

When deploying a web application to a specific location on a web application server, use the `appvdir` variable to set the location of the web application within the server's root context. For example, if a web application server's root context was `http://www.mycompany.com/BOE`, setting `appvdir` to `mywebapp/` would result in the web application being accessible from `http://www.mycompany.com/BOE/mywebapp/`.

3.6 Using the WDeploy command-line tool

Before using the WDeploy command-line tool, ensure that the WDeploy configuration files have been configured appropriately for your web application server. See WDeploy configuration files.

For information on WDeploy prerequisites, see:

- WDeploy prerequisites
- WDeploy GUI tool prerequisites

The WDeploy command-line tool is installed as a part of SAP BusinessObjects Enterprise:

`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/wdeploy.sh`

There are two types of commands:

- General commands

  Used to provide high-level information about the tool itself and the available web application server deployment. They are not used together with server names, properties, or actions. For example:

  - `wdeploy.sh help`
  - `wdeploy.sh listappservers`

- Deployment commands

  Used to deploy specific web applications to a specific web application server. These commands always follow the format: server, properties, action. For example:

  ```
  wdeploy.sh <WEB_APP_SERVER> [-D<PROPERTY>=<value>] <ACTION>
  ```

  Where:

  - `<WEB_APP_SERVER>` is the name of the web or web application server and must match the name of the WDeploy configuration file `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/config.<WEB_APP_SERVER>`.

    For example, for SAP NetWeaver AS Java 7.2, the configuration file is named `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/config.sapappsvr72`, so the name given for the `<WEB_APP_SERVER>` parameter is `sapappsvr72`.

  - `-D<PROPERTY>=<value>` is at least one key-value pair. Parameters given on the command-line override those in the configuration file.

    For example, to deploy the BOE web application, use `-DAPP=BOE`. This overrides any value for `-DAPP=` stored in the web application server configuration file `config.<WEB_APP_SERVER>`.

  - `<ACTION>` is the name of the operation to perform.

    For example, the `deployall` action will deploy all web applications to the web application server given as `<WEB_APP_SERVER>` in the first parameter.

3.6.1 Syntax

3.6.1.1 WDeploy server names

WDeploy uses a server name on the command-line and as part of the filename for server configuration files. The server name is the first parameter given on the command-line when running the WDeploy tool:

```
wdeploy.sh <WEB_APP_SERVER> [-D<PROPERTY>=<value>] <ACTION>
```

<WEB_APP_SERVER> is the name of the web or web application server. The name given must match the name of the configuration file in *<BOE_INSTALL_DIR>*`/sap_bobj/enterprise_xi40/wdeploy/conf`.

For a complete list of valid names for *<WEB_APP_SERVER>*, see Values for WEB_APP_SERVER.

3.6.1.2 WDeploy property parameters

WDeploy properties can be set on the command-line with the `-D`*<PROPERTY>* switch parameter. Repeat the `-D`*<PROPERTY>* switch parameter for each property to be set.

For example, in the following command, multiple invocations of the `-D`*<PROPERTY>* switch parameter are used to set multiple properties for a Tomcat 6 web application server (`as_dir`, `as_instance`, `as_service_name`, and `APP`:

```
wdeploy.sh Tomcat6
  -Das_dir=/opt/tomcat6
  -Das_instance=localhost
  -Das_service_name=Tomcat6
  -DAPP=BOE
  deployonly
```

Properties can also be configured in the `wdeploy.config` configuration file. This allows properties to be set by default, and over-ridden from the command-line when required. For more information on the `wdeploy.conf` configuration file, see WDeploy configuration file.

The following table lists all properties for WDeploy.

Table 3-11: Properties for WDeploy

Property	Description	Example value
APP	Name of the web application to deploy, as found in the WAR or EAR file name and the web application properties file: <*BOE_INSTALL_DIR*/sap_bobj/enterprise_xi40/wdeploy/conf/apps/<*WEB_APP*>.properties	BOE
app_source_dir	Location of an individual web application's source files. This property is used when working with individual web applications, as with the wdeploy predeploy or wdeploy deploy commands.	<*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/warfiles/webapps/<*WEB_APP*>
app_source_tree	Location of the source files for all available web applications (the parent folder of app_source_dir). This property is used when working with all available web applications simultaneously, as with the wdeploy predeployall or wdeploy deployall commands.	<*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/warfiles/webapps
as_admin_is_secure	For web application servers that use SSL encryption during web application deployment, such as Oracle Application Server, WebSphere, or Sun Java Application Server. **Note:** Setting this value to true indicates requires that you also give an administrator account name and password.	false (default) true
as_admin_password	Mandatory for NetWeaver, Oracle Application Server, and Sun Java Application Server. Web application server administrative account password.	password

Property	Description	Example value
as_admin_port	Mandatory for NetWeaver, Oracle Application Server, and Sun Java Application Server. Port number for web application server administrative access. • For WebSphere, the SOAP port. If not set, the default SOAP port is used. • For Oracle Application Server (oas1013): WDeploy expects the request port of the <notification-server> element in the file opmn.xml.	8080
as_admin_username	Mandatory for NetWeaver, Oracle Application Server, and Sun Java Application Server. Web application server administrative account username.	administrator
as_appserver_name	For Oracle Application Server. The name of the target application server.	oracleas.mycompany.com
as_dir	Installation directory of the web application server.	/opt/tomcat55
as_domain_dir	Installation directory of the web application server. For WebLogic application servers, as_domain_dir is the domain root.	/opt/bea/weblogic
as_group_id	For Oracle Application Server. The server group to which the target web application server belongs.	Default_group
as_instance	Mandatory for all the application servers. Name of the web application server instance.	localhost

Property	Description	Example value
as_lang	Preferred language for the WDeploy user interface.	• Czech: CS • Danish: DA • Dutch: NL • English: EN • Finnish: FI • French: FR • German: DE • Hungarian: HU • Italian: IT • Japanese: JA • Korean: KO • Norwegian Bokmal: NB • Polish: PL • Portuguese: PT • Russian: RU • Simplified Chinese: zh_CN • Slovak: SK • Spanish: ES • Swedish: SV • Thai: TH • Traditional Chinese: zh_TW • Turkish: TR
as_mode	Type of web application deployment. A standalone is a web application server that serves both static and dynamic web content. A split deployment uses a dedicated web server to serve static content, and a web application server to serve dynamic content.	standalone split
as_service_key	For Tomcat running on Windows. When installed as a service, the name of the Windows registry key where the JVM startup parameters are stored.	HKLM\SOFTWARE\Apache Software Foundation\Procrun 2.0\<*AS_SERVICE_NAME*>\Parameters\Java

Property	Description	Example value
as_service_key_value	For Tomcat running on Windows. When installed as a service, the value of Windows registry key where the JVM startup parameters are stored.	
as_service_name	For Tomcat running on Windows. The name of the Tomcat service.	Tomcat55
as_sid	Mandatory for NetWeaver. The system ID of the target instance.	AS1
as_soap_port	Mandatory for WebSphere. Port number for SOAP application server administration. If not set, the default SOAP port number is used.	8880
as_virtual_host	For WebSphere only. Virtual host to which the application must be bound.	default_host
classloader_package_filtering	For WebLogic 9.2 MP2 and 10. When the property is set, the application is turned into an EAR, and a filtering Classloader is set-up with the package list. This parameter is a comma-separated list of packages to filter from the classloader.	classloader_package_filter [illegible]
connector_conf_file	For Sun Java System Web Server only. Name of the configuration file holding the application server connector configuration.	<*WS_DIR*>/opii.conf
connector_host	For split deployments using Oracle Application Server. The host name of the web application server to contact.	www.mycompany.com
connector_name	For Sun Java System Web Server only. The name of the object that describes the connector service to the application server, in the configuration fileobj.conf.	passthrough

Property	Description	Example value
connector_port	For split deployments using Oracle Application Server. The ajp13 port number.	7002
connector_type	For web servers running in split mode, such as Apache. Set connector_type to the name of the dedicated web application server used by the web server.	tomcat6
deploy_as_a_filetree	For WebLogic web application servers. Indicates whether the application must be deployed as a file tree (expanded format) or as a packaged WAR or EAR file.	false (default) true
deployment_dir	Directory under which WDeploy creates a subdirectory for static content served by a dedicated web server. This name must match the web application name, and the name of the .zip archive that contains the application's static resources. WDeploy creates a virtual directory on the web server, mapping the subdirectory to the URL.	/opt/apache2
disable_CmcApp	Disables the CMC web application when set to true.	false
disable_InfoView	Disables the BI launch pad web application when set to true.	false
enforce_file_limit	Tells WDeploy whether or not the web application contains more than 65,535 files. Set to false by default, except for Sun Java System Application Server and WebSphere 6.	false (default) true
JCoStandalone	For non-SAP web application servers, use com.businessobjects.swd.jco.conn.jar and com.businessobjects.swd.jco.conn_native.jar. For SAP web application servers, use com.businessobjects.swd.jco.extensionbundle.jar.	(blank)

Property	Description	Example value
recent_app_svr	The most recent web application server to which web applications were deployed.	Tomcat6
root_context_path	Web application root context path to which all web applications are deployed. To deploy a web application to a folder within the root context, see the appvdir setting in the web application .properties configuration file.	/BOE
work_dir	Folder in which WDeploy manipulates the web applications WAR or EAR archives (for example, to split static and dynamic content in a web application). This folder stores the results of the wdeploy predeploy action, and stores data required to undeploy web applications.	<*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/workdir
ws_dir	For web servers running in split mode, such as Apache or Sun Java System Web Server. The web server home directory.	/opt/apache2
ws_instance	For WebSphere 6 only. The name of the web server where the application is installed, in split mode.	webserver1
ws_type	For web servers running in split mode, such as Apache or Sun Java System Web Server. The name of the web server, as found to the <*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/conf/config.<*WEB_SERVER*> file. **Note:** Apache, IBM HTTP Server, and Sun Java System Web Server both have a ws_type of apache.	apache
war_dir	Location of source content used to create WAR or EAR archives.	<*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/java/applications

3.6.1.2.1 Mandatory property parameters for split wet tier deployments

Some properties are required for split web tier deployments (separate web and web application servers) . When deploying to a split web tier deployment, ensure that the following property parameters are either given on the command-line, or are configured in the web or web application server configuration file (`config.<WEB_APP_SERVER>`).

- Set the `-Das_mode=split` property to separate static content for the web server and dynamic content for the web application server.
- Set the `-Dconnector_type=<CONNECTOR_TYPE>` property to the appropriate connector type. Set `connector_type` to the name of the dedicated web application server used by the web server. For example, when using an Apache web server and a Tomcat 6 web application server, `connector_type` should be set to `tomcat6`.
- On deployments where the web server runs on the same host as the web application server, you must give the `-Dws_type=<WEB_SERVER_TYPE>` and `-Dws_dir=<WEB_SERVER_DIR>` properties..

3.6.1.3 WDeploy actions

The last parameter of a WDeploy command is the action to be performed. The following section defines each valid action, and what each does.

Table 3-12: WDeploy Actions

Action	Description
`predeploy`	The `wdeploy predeploy` command prepares a web application for deployment to the target web application server. The web application's `web.xml` configuration file is updated, along with any other changes required to make the web application deployable to the target web application server. Once configured, the web application is packaged into a WAR or EAR file and saved to `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/<WEB_APP_SERVER>`. This file can be deployed manually through the web application server's administrative console, or with the `wdeploy deploy` command. Use `wdeploy predeploy` to prepare the BOE web application so that it is ready for deployment to a specific web application server. For example: `wdeploy.sh <WEB_APP_SERVER> -DAPP=BOE predeploy`

Action	Description

`predeployall`	The `wdeploy predeployall` command performs the `wdeploy predeploy` command for all web applications located in the SAP BusinessObjects Enterprise web application source directory:
	
	`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/warfiles/webapps`
	
	Use `wdeploy predeployall` to apply the `wdeploy predeploy` command to all web applications. For example:
	
	`wdeploy.sh <WEB_APP_SERVER> predeployall`
	
	Note:
	Predeployment does not require access to the web application server, with the exception of WebSphere. When deploying to WebSphere, the `default_virtual_host` parameter is mandatory.

`deploy`	The `wdeploy deploy` command is a concatenation of the `wdeploy predeploy` and `wdeploy deployonly` commands, which prepares and deploys a web application to the target web application server with just one command.
	
	Use `wdeploy deploy` to prepare and deploy the BOE web application to the target web application server.
	
	`wdeploy.sh <WEB_APP_SERVER>`
	`  -DAPP=BOE`
	`  deploy`

`deployall`	The `wdeploy deployall` command performs the `wdeploy deploy` command for all web applications located in the SAP BusinessObjects Enterprise web application source directory:
	
	`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/warfiles/webapps`
	
	Use `wdeploy deployall` to apply the `wdeploy deploy` command to all web applications. For example:
	
	`wdeploy.sh <WEB_APP_SERVER> deployall`

Action	Description
`deployonly`	The `wdeploy deployonly` command deploys a prepared web application to the target web application server. Deployable WAR or EAR files located in `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/<WEB_APP_SERVER>` are deployed to the web application server by the `wdeploy predeploy` command invoking the web application server's command-line interface. If a web application has not already been prepared for deployment, the `wdeploy predeploy` command is called automatically. Use `wdeploy deployonly` to deploy a prepared BOE web application to a web application server. For example: `wdeploy.sh <WEB_APP_SERVER> -DAPP=BOE deployonly`
`deployonlyall`	The `wdeploy deployonlyall` command performs the `wdeploy deployonly` command for all prepared web applications. Use `wdeploy deployonlyall` to apply the `wdeploy deployonly` command to all prepared web applications. For example: `wdeploy.sh <WEB_APP_SERVER> deployonlyall`
`buildwarall`	Builds a generic WAR file from the web application source tree. For example: `wdeploy.sh buildwarall -Dapp_source_tree=<LOCATION_OF_APP_SOURCE_TREE> -Dwar_dir=<TARGET_LOCATION_OF_GENERIC_WAR_FILE>`
`listapps`	Lists the web applications available for deployment to a web application server. For example: `wdeploy.sh <WEB_APP_SERVER> listapps`
`listdeployedapps`	Lists all web applications currently deployed to a web application server. This action can only be run on a web server or web application server system. For example: `wdeploy.sh <WEB_APP_SERVER> listdeployedapps`

Action	Description

`undeploy`	The `wdeploy undeploy` command invokes the target web application server's command-line interface to remove a deployed SAP BusinessObjects Enterprise web application from the server.
	
	Use the `wdeploy undeploy` command to undeploy the BOE web application from the target web application server.
	
	`wdeploy.sh <WEB_APP_SERVER>`
	`  -DAPP=BOE`
	`  undeploy`

`undeployall`	The `wdeploy undeployall` command performs the `wdeploy undeploy` command for all SAP BusinessObjects Enterprise web applications deployed to the target web application server. For example: `wdeploy.sh <WEB_APP_SERVER> undeployall`
`validateconfig`	Validates the WDeploy configuration for the supported web application servers to ensure that the deployment can be successful. For example: `wdeploy.sh <WEB_APP_SERVER> validateconfig`

Substitute *<WEB_APP_SERVER>* for the name of the web application server.

3.6.1.4 WDeploy general commands

WDeploy general commands are used to provide high-level information about the tool itself and the available web application server deployment. They are not used together with server names, properties, or actions. The following general commands are available:

- `wdeploy help`: displays a summary of available WDeploy command-line usage.
- `wdeploy listappservers`: lists all Java web application servers supported by this version of WDeploy.
- `wdeploy version`: displays the version number of the WDeploy tool itself.
- `wdeploy buildwarall`: generates a generic WAR file by using the web application source tree.

  Usage:

```
wdeploy.sh
   buildwarall
   -Dapp_source_tree=<LOCATION_OF_WEB_APP_SOURCE>
   -Dwar_dir=<TARGET_LOCATION_OF_GENERIC_WAR_FILES>
```

For example:

```
wdeploy.sh
   buildwarall
   -Dapp_source_tree=/sap/sap_bobj/enterprise_xi40/warfiles/webapps
   -Dwar_dir=/sap/sap_bobj/enterprise_xi40/java/applications
```

3.6.2 Examples of using WDeploy

This section contains examples of using WDeploy commands for supported web application servers.

Remember:
Before using WDeploy, ensure that the WDeploy configuration files have been configured appropriately for your web application server. See WDeploy configuration files. Read Special considerations for particular web application servers for a list of important information specific to your web application server.

3.6.2.1 Values for WEB_APP_SERVER

In the following examples, substitute the variable *<WEB_APP_SERVER>* for the name of your web application server, as shown in the table below.

Web application server	*<WEB_APP_SERVER>* name
JBoss 4.2.3	jboss4
JBoss 5.0	jboss5
Oracle Application Server 10g R3	oas1013
SAP NetWeaver AS Java 7.2	sapappsvr72
Sun Java System Application Server 9.1	sunone9
Tomcat 5.5.20	tomcat55
Tomcat 6.0.20	tomcat6
WebLogic 9.2 MP2	weblogic9
WebLogic 10 or 10.3	weblogic10
WebSphere 6.1.0.7	websphere6

Web application server	*<WEB_APP_SERVER>* name
WebSphere 7.0.0.11	websphere7

3.6.2.2 wdeploy predeploy

The `wdeploy predeploy` command prepares a web application for deployment to the target web application server.

The web application's internal `web.xml` configuration file is set by WDeploy, along with any other changes required to make the web application deployable to the target web application server.

Once configured, the web application is packaged into a WAR or EAR file and saved to *<BOE_INSTALL_DIR>*`/sap_bobj/enterprise_xi40/wdeploy/`*<WEB_APP_SERVER>*.

The resulting WAR or EAR file can be deployed manually through the web application server's administrative console, or with the `wdeploy deploy` command.

Example:

For example, use `wdeploy predeploy` to prepare the BOE web application so that it is ready for deployment to a specific web application server.

```
wdeploy.sh <WEB_APP_SERVER>
  -DAPP=BOE
  predeploy
```

Substitute *<WEB_APP_SERVER>* for the name of the web application server, as shown in the following table.

The table below lists example parameters for predeploying the BOE web application for specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.`*<WEB_APP_SERVER>* configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 -Das_instance=default -DAPP=BOE predeploy`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 -Das_instance=default -DAPP=BOE predeploy`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 -DAPP=BOE predeploy` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`)	`wdeploy.sh sapappsvr72 -DAPP=BOE predeploy`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -DAPP=BOE predeploy`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 -DAPP=BOE predeploy`
Tomcat 6.0 (`tomcat6`)	`wdeploy.sh Tomcat6 -DAPP=BOE predeploy`
WebLogic 9.2 MP2(`weblogic9`)	`wdeploy.sh weblogic9 -DAPP=BOE predeploy`
WebLogic 10 and 10.3(`weblogic10`)	`wdeploy.sh weblogic10 -DAPP=BOE predeploy`
WebSphere 6.1 (`websphere6`)	

Server	Parameters

	`wdeploy.sh websphere6`
	`  -Das_virtual_host=default_host`
	`  -DAPP=BOE`
	`  predeploy`

WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7`
	`  -Das_virtual_host=default_host`
	`  -DAPP=BOE`
	`  predeploy`

3.6.2.3 wdeploy predeployall

The `wdeploy predeployall` command performs the `wdeploy predeploy` command for all web applications located in the SAP BusinessObjects Enterprise web application source directory:

`<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/warfiles/webapps`

Example:

For example, use `wdeploy predeployall` to apply the `wdeploy predeploy` command to all web applications.

```
wdeploy.sh <WEB_APP_SERVER> predeployall
```

Substitute *<WEB_APP_SERVER>* for the name of the web application server, as shown in the following table.

The table below lists example parameters for predeploying all web applications for specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.`*<WEB_APP_SERVER>* configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 predeployall`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 predeployall`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 predeployall` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`	`wdeploy.sh sapappsvr72 predeployall`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -Das_admin_is_secure=true predeployall`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 predeployall`
Tomcat 6.0 (`tomcat6`)	`wdeploy.sh Tomcat6 predeployall`
WebLogic 9.2 MP2 (`weblogic9`)	`wdeploy.sh weblogic9 predeployall`
WebLogic 10 and 10.3 (`weblogic10`)	`wdeploy.sh weblogic10 predeployall`
WebSphere 6.1 (`websphere6`)	`wdeploy.sh websphere6 -Das_virtual_host=default_host predeployall`
WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7 -Das_virtual_host=default_host predeployall`

3.6.2.4 wdeploy deployonly

The `wdeploy deployonly` command deploys a prepared web application to the target web application server.

Deployable WAR or EAR files located in `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/<WEB_APP_SERVER>` are deployed to the web application server by the `wdeploy predeploy` command invoking the web application server's command-line interface.

If a web application has not already been prepared for deployment, the `wdeploy predeploy` command is called automatically.

Example:

For example, use `wdeploy deployonly` to deploy a prepared BOE web application to a web application server.

```
wdeploy.sh <WEB_APP_SERVER>
  -DAPP=BOE
  deployonly
```

Substitute `<WEB_APP_SERVER>` for the name of the web application server, as shown in the following table.

The table below lists example parameters for deploying the BOE web application to specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.<WEB_APP_SERVER>` configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 -Das_dir=/opt/jboss-4 -Das_instance=default -DAPP=BOE deployonly`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 -Das_dir=/opt/jboss-5 -Das_instance=default -DAPP=BOE deployonly`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 -Das_dir=/opt/product/10.1.3.1/OracleAS_1 -Das_appserver_name=oracle10g.myhost.mycompany.com -Das_admin_port=6003 -Das_admin_username=oc4jadmin -Das_admin_password=password1 -DAPP=BOE deployonly` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`	`wdeploy.sh sapappsvr72 -Das_dir=/usr/sap -Das_instance=JC01 -Das_sid=AS2 -Das_admin_username=Administrator -Das_admin_password=password1 -Das_admin_port=50004 -DAPP=BOE deployonly`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -Das_dir=/opt/SUNWappserver91 -Das_instance=server -Das_domain=domain1 -Das_admin_port=4849 -Das_admin_username=admin -Das_admin_password=password -Das_admin_is_secure=true -DAPP=BOE deployonly`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 -Das_dir=/opt/tomcat5 -Das_instance=localhost -Das_service_name=Tomcat5 -DAPP=BOE deployonly`
Tomcat 6.0 (`tomcat6`)	

Server	Parameters

	`wdeploy.sh Tomcat6`
	`  -Das_dir=/opt/tomcat6`
	`  -Das_instance=localhost`
	`  -Das_service_name=Tomcat6`
	`  -DAPP=BOE`
	`  deployonly`

WebLogic 9.2 MP2 (`weblogic9`)	`wdeploy.sh weblogic9`
	`  -Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`  -Das_admin_port=7001`
	`  -Das_instance=AdminServer`
	`  -Das_admin_username=weblogic`
	`  -Das_admin_password=weblogic`
	`  -DAPP=BOE`
	`  deployonly`

WebLogic 10 and 10.3 (`weblogic10`)	`wdeploy.sh weblogic10`
	`  -Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`  -Das_admin_port=7001`
	`  -Das_instance=AdminServer`
	`  -Das_admin_username=weblogic`
	`  -Das_admin_password=weblogic`
	`  -DAPP=BOE`
	`  deployonly`

WebSphere 6.1 (`websphere6`)	`wdeploy.sh websphere6`
	`  -Das_dir=/opt/ibm/WebSphere/AppServer`
	`  -Das_instance=server1`
	`  -Das_virtual_host=default_host`
	`  -Das_profile_name=AppSrv01`
	`  -Das_soap_port=8880`
	`  -DAPP=BOE`
	`  deployonly`

WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7`
	`  -Das_dir=/opt/IBM/WebSphere/AppServer`
	`  -Das_instance=server1`
	`  -Das_virtual_host=default_host`
	`  -Das_profile_name=AppSrv01`
	`  -Das_soap_port=8880`
	`  -DAPP=BOE`
	`  deployonly`

3.6.2.5 wdeploy deployonlyall

The `wdeploy deployonlyall` command performs the `wdeploy deployonly` command for all prepared web applications.

Example:

For example, use `wdeploy deployonlyall` to apply the `wdeploy deployonly` command to all prepared web applications.

```
wdeploy.sh <WEB_APP_SERVER> deployonlyall
```

Substitute <*WEB_APP_SERVER*> for the name of the web application server, as shown in the following table.

The table below lists example parameters for deploying all web applications to specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.`<*WEB_APP_SERVER*> configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 -Das_dir=/opt/jboss-4 -Das_instance=default deployonlyall`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 -Das_dir=/opt/jboss-5 -Das_instance=default deployonlyall`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 -Das_dir=/opt/product/10.1.3.1/OracleAS_1 -Das_appserver_name=oracle10g.myhost.mycompany.com -Das_admin_port=6003 -Das_admin_username=oc4jadmin -Das_admin_password=password1 deployonlyall` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`)	`wdeploy.sh sapappsvr72 -Das_dir=/usr/sap -Das_instance=JC01 -Das_sid=AS2 -Das_admin_username=Administrator -Das_admin_password=password1 -Das_admin_port=50004 deployonlyall`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -Das_dir=/opt/SUNWappserver91 -Das_instance=server -Das_domain=domain1 -Das_admin_port=4849 -Das_admin_username=admin -Das_admin_password=password -Das_admin_is_secure=true deployonlyall`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 -Das_dir=/opt/tomcat5 -Das_instance=localhost -Das_service_name=Tomcat5 deployonlyall`
Tomcat 6.0 (`tomcat6`)	`wdeploy.sh Tomcat6 -Das_dir=/opt/tomcat6 -Das_instance=localhost -Das_service_name=Tomcat6 deployonlyall`

Server	Parameters

WebLogic 9.2 MP2 (`weblogic9`)	`wdeploy.sh weblogic9`
	`-Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`deployonlyall`

WebLogic 10 and 10.3 (`weblogic10`)	`wdeploy.sh weblogic10`
	`-Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`deployonlyall`

WebSphere 6.1 (`websphere6`)	`wdeploy.sh websphere6`
	`-Das_dir=/opt/ibm/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`deployonlyall`

WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7`
	`-Das_dir=/opt/ibm/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`deployonlyall`

3.6.2.6 wdeploy deploy

The `wdeploy deploy` command is a concatenation of the `wdeploy predeploy` and `wdeploy deployonly` commands, which prepares and deploys a web application to the target web application server with just one command.

Example:

For example, use `wdeploy deploy` to prepare and deploy the BOE web application to the target web application server.

```
wdeploy.sh <WEB_APP_SERVER>
  -DAPP=BOE
  deploy
```

Substitute <*WEB_APP_SERVER*> for the name of the web application server, as shown in the following table.

The table below lists example parameters for deploying the BOE web application to specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.`<*WEB_APP_SERVER*> configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 -Das_dir=/opt/jboss-4 -Das_instance=default -DAPP=BOE deploy`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 -Das_dir=/opt/jboss-5 -Das_instance=default -DAPP=BOE deploy`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 -Das_dir=/opt/product/10.1.3.1/OracleAS_1 -Das_appserver_name=oracle10g.myhost.mycompany.com -Das_admin_port=6003 -Das_admin_username=oc4jadmin -Das_admin_password=password1 -DAPP=BOE deploy` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`	`wdeploy.sh sapappsvr72 -Das_dir=/usr/sap -Das_instance=JC01 -Das_sid=AS2 -Das_admin_username=Administrator -Das_admin_password=password1 -Das_admin_port=50004 -DAPP=BOE deploy`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -Das_dir=/opt/SUNWappserver91 -Das_instance=server -Das_domain=domain1 -Das_admin_port=4849 -Das_admin_username=admin -Das_admin_password=password -Das_admin_is_secure=true -DAPP=BOE deploy`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 -Das_dir=/opt/tomcat5 -Das_instance=localhost -Das_service_name=Tomcat5 -DAPP=BOE deploy`
Tomcat 6.0 (`tomcat6`)	

Server	Parameters

	`wdeploy.sh Tomcat6`
	`-Das_dir=/opt/tomcat6`
	`-Das_instance=localhost`
	`-Das_service_name=Tomcat6`
	`-DAPP=BOE`
	`deploy`

WebLogic 9.2 MP2 (`weblogic9`)	`wdeploy.sh weblogic9`
	`-Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`-DAPP=BOE`
	`deploy`

WebLogic 10 and 10.3 (`weblogic10`)	`wdeploy.sh weblogic10`
	`-Das_domain_dir=/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`-DAPP=BOE`
	`deploy`

WebSphere 6.1 (`websphere6`)	`wdeploy.sh websphere6`
	`-Das_dir=/opt/ibm/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`-DAPP=BOE`
	`deploy`

WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7`
	`-Das_dir=/opt/ibm/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`-DAPP=BOE`
	`deploy`

3.6.2.7 wdeploy deployall

The `wdeploy deployall` command performs the `wdeploy deploy` command for all web applications located in the SAP BusinessObjects Enterprise web application source directory:

<*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/warfiles/webapps

Example:

For example, use `wdeploy deployall` to apply the `wdeploy deploy` command to all web applications.

```
wdeploy.sh <WEB_APP_SERVER> deployall
```

Substitute <*WEB_APP_SERVER*> for the name of the web application server, as shown in the following table.

The table below lists example parameters for deploying the all web applications to specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.`<*WEB_APP_SERVER*> configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 -Das_dir=/opt/jboss-4 -Das_instance=default deployall`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 -Das_dir=/opt/jboss-5 -Das_instance=default deployall`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 -Das_dir=/opt/product/10.1.3.1/OracleAS_1 -Das_appserver_name=oracle10g.myhost.mycompany.com -Das_admin_port=6003 -Das_admin_username=oc4jadmin -Das_admin_password=password1 deployall` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`)	`wdeploy.sh sapappsvr72 -Das_dir=/usr/sap -Das_instance=JC01 -Das_sid=AS2 -Das_admin_username=Administrator -Das_admin_password=password1 -Das_admin_port=50004 deployall`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -Das_dir=/opt/SUNWappserver91 -Das_instance=server -Das_domain=domain1 -Das_admin_port=4849 -Das_admin_username=admin -Das_admin_password=password -Das_admin_is_secure=true deployall`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 -Das_dir=/opt/tomcat5 -Das_instance=localhost -Das_service_name=Tomcat5 deployall`
Tomcat 6.0 (`tomcat6`)	`wdeploy.sh Tomcat6 -Das_dir=/opt/tomcat6 -Das_instance=localhost -Das_service_name=Tomcat6 deployall`

Server	Parameters

WebLogic 9.2 MP2 (`weblogic9`)	`wdeploy.sh weblogic9`
	`-Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`deployall`

WebLogic 10 and 10.3 (`weblogic10`)	`wdeploy.sh weblogic10`
	`-Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`deployall`

WebSphere 6.1 (`websphere6`)	`wdeploy.sh websphere6`
	`-Das_dir=/opt/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`deployall`

WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7`
	`-Das_dir=/opt/ibm/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`deployall`

3.6.2.8 wdeploy undeploy

The `wdeploy undeploy` command invokes the target web application server's command-line interface to remove a deployed SAP BusinessObjects Enterprise web application from the server.

Example:

For example, use the `wdeploy undeploy` command to undeploy the BOE web application from the target web application server.

```
wdeploy.sh <WEB_APP_SERVER>
  -DAPP=BOE
  undeploy
```

Substitute <*WEB_APP_SERVER*> for the name of the web application server, as shown in the following table.

The table below lists example parameters for undeploying the BOE web application from specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.`<*WEB_APP_SERVER*> configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 -Das_dir=/opt/jboss-4 -Das_instance=default -DAPP=BOE undeploy`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 -Das_dir=/opt/jboss-5 -Das_instance=default -DAPP=BOE undeploy`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 -Das_dir=/opt/product/10.1.3.1/OracleAS_1 -Das_appserver_name=oracle10g.myhost.mycompany.com -Das_admin_port=6003 -Das_admin_username=oc4jadmin -Das_admin_password=password1 -DAPP=BOE undeploy` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`	`wdeploy.sh sapappsvr72 -Das_dir=/usr/sap -Das_instance=JC01 -Das_sid=AS2 -Das_admin_username=Administrator -Das_admin_password=password1 -Das_admin_port=50004 -DAPP=BOE undeploy`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -Das_dir=/opt/SUNWappserver91 -Das_instance=server -Das_domain=domain1 -Das_admin_port=4849 -Das_admin_username=admin -Das_admin_password=password -Das_admin_is_secure=true -DAPP=BOE undeploy`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 -Das_dir=/opt/tomcat5 -Das_instance=localhost -DAPP=BOE -Das_service_name=Tomcat5 undeploy`
Tomcat 6.0 (`tomcat6`)	

Server	Parameters

	`wdeploy.sh Tomcat6`
	`-Das_dir=/opt/tomcat6`
	`-Das_instance=localhost`
	`-DAPP=BOE`
	`-Das_service_name=Tomcat6`
	`undeploy`

WebLogic 9.2 MP2 (`weblogic9`)	`wdeploy.sh weblogic9`
	`-Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`-DAPP=BOE`
	`undeploy`

WebLogic 10 and 10.3 (`weblogic10`)	`wdeploy.sh weblogic10`
	`-Das_domain_dir=/opt/bea/user_projects/domains/base_domain`
	`-Das_admin_port=7001`
	`-Das_instance=AdminServer`
	`-Das_admin_username=weblogic`
	`-Das_admin_password=weblogic`
	`-DAPP=BOE`
	`undeploy`

WebSphere 6.1 (`websphere6`)	`wdeploy.sh websphere6`
	`-Das_dir=/opt/ibm/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`-DAPP=BOE`
	`undeploy`

WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7`
	`-Das_dir=/opt/ibm/WebSphere/AppServer`
	`-Das_instance=server1`
	`-Das_virtual_host=default_host`
	`-Das_profile_name=AppSrv01`
	`-Das_soap_port=8880`
	`-DAPP=BOE`
	`undeploy`

3.6.2.9 wdeploy undeployall

The `wdeploy undeployall` command performs the `wdeploy undeploy` command for all SAP BusinessObjects Enterprise web applications deployed to the target web application server.

Example:

```
wdeploy.sh <WEB_APP_SERVER> undeployall
```

Substitute <*WEB_APP_SERVER*> for the name of the web application server, as shown in the following table.

The table below lists example parameters for undeploying all web applications from specific web application servers. All parameters are required, unless they are marked as optional in the configuration files. Parameters given on the command-line over-ride those stored in configuration files. However, it is recommended that you configure the parameters in the `config.`<*WEB_APP_SERVER*> configuration file appropriate for your deployment.

Server	Parameters
JBoss 4.2.3 (`jboss4`)	`wdeploy.sh jboss4 -Das_dir=/opt/jboss-4 -Das_instance=default undeployall`
JBoss 5 (`jboss5`)	`wdeploy.sh jboss5 -Das_dir=/opt/jboss-5 -Das_instance=default undeployall`
Oracle Application Server 10g R3 (`oas1013`)	`wdeploy.sh oas1013 -Das_dir=/opt/product/10.1.3.1/OracleAS_1 -Das_appserver_name=oracle10g.myhost.mycompany.com -Das_admin_port=6003 -Das_admin_username=oc4jadmin -Das_admin_password=password1 undeployall` **Note:** Before deploying web applications to Oracle 10g R3, see Before the deployment of web applications to Oracle 10g R3.
SAP NetWeaver AS Java 7.2 (`sapappsvr72`)	`wdeploy.sh sapappsvr72 -Das_dir=/usr/sap -Das_instance=JC01 -Das_sid=AS2 -Das_admin_username=Administrator -Das_admin_password=password1 -Das_admin_port=50004 undeployall`
Sun Java System Application Server 9.1 (`sunone9`)	`wdeploy.sh sunone9 -Das_dir=/opt/SUNWappserver91 -Das_instance=server -Das_domain=domain1 -Das_admin_port=4849 -Das_admin_username=admin -Das_admin_password=password -Das_admin_is_secure=true undeployall`
Tomcat 5.5 (`tomcat55`)	`wdeploy.sh tomcat55 -Das_dir=/opt/tomcat5 -Das_instance=localhost -Das_service_name=Tomcat5 undeployall`
Tomcat 6.0 (`tomcat6`)	`wdeploy.sh Tomcat6 -Das_dir=/opt/tomcat6 -Das_instance=localhost -Das_service_name=Tomcat6 undeployall`

Server	Parameters
WebLogic 9.2 MP2 (`weblogic9`)	`wdeploy.sh weblogic9 -Das_domain_dir=/opt/bea/user_projects/domains/base_domain -Das_admin_port=7001 -Das_instance=AdminServer -Das_admin_username=weblogic -Das_admin_password=weblogic undeployall`
WebLogic 10 and 10.3 (`weblogic10`)	`wdeploy.sh weblogic10 -Das_domain_dir=/opt/bea/user_projects/domains/base_domain -Das_admin_port=7001 -Das_instance=AdminServer -Das_admin_username=weblogic -Das_admin_password=weblogic undeployall`
WebSphere 6.1 (`websphere6`)	`wdeploy.sh websphere6 -Das_dir=/opt/ibm/WebSphere/AppServer -Das_instance=server1 -Das_virtual_host=default_host -Das_profile_name=AppSrv01 -Das_soap_port=8880 undeployall`
WebSphere 7.0 (`websphere7`)	`wdeploy.sh websphere7 -Das_dir=/opt/ibm/WebSphere/AppServer -Das_instance=server1 -Das_virtual_host=default_host -Das_profile_name=AppSrv01 -Das_soap_port=8880 undeployall`

3.6.3 Special considerations for particular web application servers

The following section contains important information related to the deployment of web applications on your web application server.

3.6.3.1 WebSphere

3.6.3.1.1 To deploy to separate IHS web and WebSphere web application servers

To reduce the load on a web application server, you can set up a separate, dedicated, web server to serve static content. All static content will be served by the web server, while dynamic content will be served by the web application server.

Note:
This configuration requires that you use the WDeploy command to split web application resources into static and dynamic content that can be deployed to separate web and web application servers.

1. Set up an IBM HTTP Server (IHS) web server and ensure that it is working correctly. Load a web page, such as the IHS default test page, to verify that the web server is serving content correctly.

   Open a web browser and enter the IP address or hostname of the web server, and a port number if the server is not listening on port 80. For example: `http://ihs.mycompany.com`.

2. Ensure that your WebSphere web application server is working correctly.

   Open a web browser and enter the IP address or hostname of the web application server, and a port number. For example: `http://websphere.mycompany.com:9080`.

   Note:
   If you have any existing SAP BusinessObjects Enterprise web applications running on the server, they must be undeployed before continuing.

3. Run the web server plug-in installation wizard to install the plug-in that bridges WebSphere with IHS, and follow the directions to enter information about your IHS web server.
4. Follow the plug-in configuration instructions on the WebSphere web site for configuring the bridge between IHS and WebSphere.

   Requests for dynamic resources are now forwarded to WebSphere when received by IHS.

5. Ensure that the bridge between the web server and web application server is working by pointing a browser to the web server and verifying that dynamic content from the web application server is served correctly.

   For example, visit the URL: `http://ihs.mycompany.com/snoop/`.

6. If the web application server is installed on the same machine as SAP BusinessObjects Enterprise, run WDeploy locally on that machine. If the web application server runs on a different machine copy the WDeploy command and environment to the web application server. See *To deploy web applications on a remote machine*.
7. Configure WDeploy environment to separate content between the web server and the web application server. This is known as "split" mode.

   The WDeploy configuration files for Apache and WebSphere are located in `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf`.

   - Edit `config.apache`. For example:

     ```
     ws_dir=/opt/apache
     connector_type=websphere6
     deployment_dir=/opt/apache/htdocs
     plugin_install_dir=${ws_dir}/Plugins
     ```

- Edit `config.websphere6`. For example:

```
as_soap_port=8880
#as_admin_username=admin
#as_admin_password=password
as_dir=/opt/websphere/appserver
as_instance=server1
as_plugin_cfg_dir=/opt/ibm/WebSphere/AppSever/profiles/AppSrv01/config/cells/<CELLNAME>/nodes/<NODE
NAME>/servers/${ws_instance}
as_profile_name=AppSrv01
as_virtual_host=default_host
as_admin_is_secure=false
enforce_file_limit=true
ws_instance=webserver1
```

8. Use `wdeploy predeploy` in split mode to split source web applications into separate static and dynamic resources.

   Note:
   Before running `wdeploy predeploy`, ensure that parameter `as_plugin_cfg_dir` in `config.websphereX` has been set to the folder that contains the WebSphere file `plugin-cfg.xml`.

   For example:

   - Run the following command to extract static content for the IHS web server.

```
wdeploy.sh websphere6
   -Das_mode=split
   -Dws_type=apache
   predeployall
```

   The dynamic content of web applications is located in: `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere6/application`. The static content is located in: `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere6/resources`.

9. Run `wdeploy deployonlyall` command to deploy the dynamic content to WebSphere application server and static content to IHS.

   If IHS and WebSphere are on the same machine, static and dynamic content will be automatically deployed to servers by the following command:

```
wdeploy.sh websphere6 -Das_mode=split
    -Dws_type=apache deployonlyall
```

   Note:
   If your dynamic and static content are in a custom location, use the -Dwork_dir parameter.

   If IHS and WebSphere are on different machines, dynamic content will be automatically deployed to WebSphere by the following command. Static content must be manually deployed to the remote IHS machine afterwards.

```
./wdeploy.sh websphere6 -Das_mode=split
    deployonlyall
```

   Note:
   If your dynamic and static content are in a custom location, use the -Dwork_dir parameter.

   Copy static content to the `htdocs` directory on the web server:

- Extract the zip files on the web application server under <*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere6/resources.
- Copy these folders from the WebSphere web application server to the IHS <*WS_DIR*>/htdocs folder on the IHS server.
- Copy the bobj.<application>.conf files from the WebSphere web application server to the IHS <*WS_DIR*>/conf folder on the IHS server.
- Update IHS httpd.conf under <*WS_DIR*>/conf with the application configuration files. For each web application include an entry in httpd.conf. For example, to include BOE, you would enter:

```
Include conf/bobj.BOE.conf
```

10. Copy the plugin-cfg.xml file to the IHS and WebSphere work folders. If you are using the WebSphere administrative console to deploy, and IHS and WebSphere are installed on the same host system, skip to the next step. The WDeploy command will automatically copy plugin-cfg.xml when IHS and WebSphere are installed on the same host system.

    For example, copy the following file:

    <*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/work_dir/websphere6/application/PluginSplit/plugin-cfg.xml

    To the WebSphere work folder:

    <*WS_HOME*>/AppServer/profiles/AppSrv01/config/cells/<*CELL_NAME*>/nodes/<*NODE_NAME*>/servers/<*SERVER_NAME*>

    And to the IHS work folder:

    <*WS_DIR*>/Plugins/config/<*WEB_SERVER_NAME*>

11. Ensure that both static and dynamic content are correctly configured by trying to access a web application through the web server. For example, create a URL that includes the address of the web server with the root context of a web application deployed to the web application server: http://ihs.mycompany.com/BOE/CMC/. In this example, ihs.mycompany.com is the web server, and /BOE/CMC/ is a deployed web application.

Static content is now served by a dedicated web server, and dynamic content is served by a dedicated web application server.

3.6.3.1.2 Disable WebSphere JSF implementation

If pages rendered by WebSphere do not show cached values correctly, it may be using its own version of the JavaServer Faces (JSF) library, rather than the JSF implementation supplied with SAP BusinessObjects Enterprise. To correct this problem, manually reconfigure WebSphere for any web application that uses JSF. For example, the BOE web application uses JSF.

To disable the JSF implementation supplied with WebSphere, follow the instructions below.

1. Log in to the "WebSphere Application Server Administration" console using the following URL: http://*WAS_HOSTNAME*:*PORT*/admin. The WebSphere administrative console's default port number is 9060.
2. Browse to **YOUR_WEB_APPLICATION > Manage Modules > YOUR_WEB_APPLICATION.ear**

Replace **YOUR_WEB_APPLICATION** in the example above with the name of your web application.

3. Stop the web application, so its settings can be modified.
4. Select **Classes loaded with application class loader first (parent last)** from the combo box.
   Restart the web application.
5. Repeat these steps for all other web applications that use JSF.

Reload pages that display cached values. Cached values are now displayed correctly.

3.6.3.1.3 Classes loaded with application class loader first

Log in to the WebSphere "Integrated Solutions Console" with the Administrator account. You can run the WebSphere "Administrative console" program, or use a web browser to open `http://`*`WAS_HOSTNAME`*`:`*`PORT`*`/ibm/console` where *`WAS_HOSTNAME`* is the name of your WebSphere server, and *`PORT`* is the port number on which the server listens for login requests. The default port number is 9060.

1. Select **Applications > Application Type > WebSphere enterprise applications** in the menu.
   The "Enterprise Applications" screen appears.
2. Choose the web application deployed by WDeploy from the list of administered resources.
   The "Enterprise Applications" configuration screen appears.
3. Click **Manage Modules**.
   The "Manage Modules" screen appears.
4. Select the web application module (in the "Module" column).
   The "Manage Modules General Properties" screen appears.
5. Select **Classes loaded with application class loader first (parent last)** from the "Class loader order" property.
   A confirmation message appears.
6. Click **Save directly to the master configuration**.
   The web application configuration is saved and you are returned to the "Manage Modules" screen.
7. Click **OK**.
   A master configuration change confirmation message appears.
8. Click **Save directly to the master configuration**.
   The web application configuration is saved and you are returned to the "Enterprise Applications" screen.
9. Select the web application checkbox and click **Start**.
   A message appears to confirm that the web application started successfully.

When deploying more than one web application, repeat steps 2 to 9 for each web application.

3.6.3.1.4 WebSphere 7.0 deployment time

The deployment of the BOE web application archive (`BOE.ear`) to a WebSphere 7.0 web application server may take a long time. This is due to the way that WebSphere 7.0 handles the deployment of large web applications.

Note:
The following work-around applies only to the deployment of the `BOE.ear` archive on WebSphere 7.0. To deploy other web applications on WebSphere 7.0, see wdeploy deploy. WebSphere 6.x is not affected.

To work around this problem, the `BOE.ear` archive must be updated, and the `BOE.ear/BOE.war/WEB-INF/eclipse` folder must moved to the WebSphere web application server.

To perform this update, follow the steps below.

1. WDeploy requires Java. If the `JAVA_HOME` environment variable is not defined in your environment, you can use the JDK that comes with SAP BusinessObjects Enterprise.

   The JDK bundled with SAP BusinessObjects Enterprise is located in `<BOE_INSTALL_DIR>/sap_bobj/<PLATFORM>/enterprise_xi40/jdk`. For example:

   ```
   export JAVA_HOME=/opt/sap/sap_bobj/hpux_i64/enterprise_xi40/jdk
   ```

2. Ensure that the Java binaries are included in the `PATH` environment variable. If `JAVA_HOME` is defined correctly, you can use the Java `bin` folder.

   For example:

   ```
   export PATH=$PATH:$JAVA_HOME/bin
   ```

3. Create the `BOE` web application archive with the `wdeploy predeploy` command.

   The WDeploy command is located in: `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy`.

   For example, use the following WDeploy tool command to generate a deployable `BOE.ear` archive for WebSphere 7.0:

   ```
   ./wdeploy.sh websphere7
     -Das_virtual_host=default_host
     -DAPP=BOE
     predeploy
   ```

   Note:
   Generating `BOE.ear` may take a long time (approximately 45 minutes).

   The WDeploy tool creates `BOE.ear` in:

   `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere7/applications`.

4. Create temporary folders and extract the `BOE.ear` archive.
   a. Create temporary folders named `temp1`, `temp2`, and `temp3`.

For example:

```
mkdir temp1 temp2 temp3
```

b. Change directory to the `temp1` folder.
For example:

```
cd temp1
```

c. From inside the `temp1` folder, extract the `BOE.ear` archive.
For example:

```
jar xvf <BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere7/applications/BOE.ear
```

d. Remove the original `BOE.ear` archive. For example:

```
rm <BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere7/applications/BOE.ear
```

5. From inside the `temp2` folder, extract the `BOE.war` archive.
   a. Change directory from the `temp1` folder to the `temp2` folder.
   For example:

```
cd ../temp2
```

   b. Extract `BOE.war`.
   For example:

```
jar xvf ../temp1/BOE.war
```

   Extracting `BOE.war` inside the `temp2` temporary folder results in the following structure:

```
temp1/
   BOE.war           (extracted from BOE.ear)
   META-INF/         (extracted from BOE.ear)
temp2/               (current folder)
   META-INF/         (extracted from BOE.war)
   WEB-INF/          (extracted from BOE.war)
      classes/
      config/
      eclipse/
      ibm-web-bnd.xmi
      ibm-web-ext.xmi
      internal/
      jars/
      lib/
      TraceLog/
      web.xml
temp3/
```

6. From inside the `temp2` folder, move the `eclipse` sub-folder in `WEB-INF` to the `temp3` folder.
   For example:

```
mv WEB-INF/eclipse ../temp3
```

   Moving the `temp2/WEB-INF/eclipse` folder to the `temp3` folder results in the following structure:

```
temp1/
   BOE.war           (extracted from BOE.ear)
   META-INF/         (extracted from BOE.ear)
temp2/
   META-INF/         (extracted from BOE.war)
```

```
   WEB-INF/          (current folder)
      classes/
      config/
      ibm-web-bnd.xmi
      ibm-web-ext.xmi
      internal/
      jars/
      lib/
      TraceLog/
      web.xml
temp3/
   eclipse/           (from temp2/WEB-INF)
```

7. From inside the `temp2` folder, remove and recreate the `BOE.war` archive.
   a. Remove the original `BOE.war` archive.
      For example:

      ```
      rm ../temp1/BOE.war
      ```

   b. Create a new `BOE.war` archive, containing the original `META-INF` folder, and the updated `WEB-INF` folder.
      For example:

      ```
      jar cvf ../temp1/BOE.war META-INF WEB-INF
      ```

8. From inside the `temp1` folder, remove and recreate the `BOE.ear` archive.
   a. Change directory from the `temp2` folder to the `temp1` folder.
      For example:

      ```
      cd ../temp1
      ```

   b. Remove the original `BOE.ear` archive from the WDeploy `applications` folder.
      For example:

      ```
      rm <BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere7/applications/BOE.ear
      ```

      The original `BOE.ear` archive can always be recreated with the `wdeploy predeploy` command in step 3, if required.
   c. Create an updated `BOE.ear` archive in the WDeploy `applications` folder. The updated `BOE.ear` archive will contain the updated `BOE.war` archive, and the original `META-INF` folder.
      For example:

      ```
      jar cvf <BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere7/applications/BOE.ear
      BOE.war META-INF
      ```

   The `BOE.ear` archive has been extracted from `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere7/applications` and updated with a new `BOE.war` archive and the original `META-INF` folder.

9. The updated `BOE.ear` archive now is much smaller, and can now be deployed to the WebSphere web application server. The updated BOE.ear file is located in:

   `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/websphere7/applications`

To deploy `BOE.ear`, use the WebSphere administrative console.

Note:
Do not use the WDeploy command to deploy the updated `BOE.ear`.

10. The `eclipse` folder that was copied to the `temp3` folder in step 6 must be loaded by WebSphere.

    To get WebSphere to load the contents of the `eclipse` folder, move the `temp3/eclipse` folder to the WebSphere 7 deployment destination folder (for example, `<WS7_HOME>/AppServer/profiles/<PROFILE>/installedapps/<NODE>/BOE.ear/BOE.war/WEB-INF/`) and restart WebSphere.

    a. Change directory from the `temp1` folder to its parent folder.

       For example:

       ```
       cd ..
       ```

    b. From the temporary folders' parent folder, move the `temp3` subfolder in the `eclipse` folder to the expanded WebSphere deployment folder:

       `<WS_HOME>/profiles/<SERVER_NAME>/installedApps/<CELL_NAME>/BOE.ear/BOE.war/WEB-INF/eclipse`

       For example:

       ```
       mv temp3/eclipse <WS_HOME>/profiles/<SERVER_NAME>/installedApps/<CELL_NAME>/BOE.ear/BOE.war/WEB-INF
       ```

    c. Restart the WebSphere web application server.

11. From the temporary folders' parent folder, remove the temporary folders `temp1`, `temp2`, and `temp3`.

    For example:

    ```
    rm -R temp1 temp2 temp3
    ```

The `BOE.ear` web application has been updated and redeployed to WebSphere 7.0.

The `eclipse` folder, which was originally located in `BOE.ear/BOE.war/WEB-INF/eclipse`, has been moved to the WebSphere deployment folder (`<WS_HOME>/profiles/<SERVER_NAME>/installedApps/<CELL_NAME>/BOE.ear/BOE.war/WEB-INF/eclipse`).

3.6.3.2 WebLogic

3.6.3.2.1 To deploy to separate Apache web and WebLogic web application servers

To reduce the load on a web application server, you can set up a separate, dedicated, web server to serve static content. All static content will be served by the web server, while dynamic content will be served by the web application server.

Note:
This configuration requires that you use the WDeploy command to split web application resources into static and dynamic content that can be deployed to separate web and web application servers.

1. Set up the Apache web server and ensure that it is working correctly. Load a web page, such as Apache's default test page, to verify that the web server is serving content correctly.

   Open a web browser and enter the IP address or hostname of the web server, and a port number if the server is not listening on port 80. For example: `http://apache.mycompany.com`.

2. Ensure that your WebLogic web application server is working correctly.

   Open a web browser and enter the IP address or hostname of the web application server, and a port number. For example: `http://weblogic.mycompany.com:7001`.

   Note:
   If you have any existing SAP BusinessObjects Enterprise web applications running on the server, they must be undeployed before continuing.

3. Download the WebLogic Apache HTTP Server Plug-In from the BEA web site. The plug-in allows you to connect an Apache web server with a WebLogic web application server, so Apache can forward requests for dynamic resources to WebLogic.
4. Follow the plug-in configuration instructions on the BEA web site for configuring the bridge between Apache web server and WebLogic.

   Requests for dynamic resources are now forwarded to WebLogic when received by Apache.

5. Ensure that the bridge between the web server and web application server is working by pointing a browser to the web server and verifying that dynamic content from the web application server is served correctly.

   For example, visit the URL: `http://apache.mycompany.com/jsp-examples/`.

   Note:
   This example URL will only work if you have manually deployed the `jsp-examples` web application.

6. If the web application server is installed on the same machine as SAP BusinessObjects Enterprise, run WDeploy locally on that machine. If the web application server runs on a different machine copy the WDeploy command and environment to the web application server. See *To deploy web applications on a remote machine*.
7. Configure the WDeploy environment to separate content between the web server and the web application server. This is known as "split" mode.

   The WDeploy configuration files for Apache and WebLogic are located in `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf`.

   - Edit `config.weblogic9` or `config.weblogic10`. For example:

     ```
     as_admin_port=7001
     as_admin_username=weblogic
     as_admin_password=weblogic
     as_instance=AdminServer
     as_domain_dir=/usr/bea/user_projects/domains/base_domain
     ```

- Edit `config.apache`. For example:

```
ws_dir=/opt/apache224/
connector_type=weblogic9
deployment_dir=/opt/apache224/htdocs
```

8. Use `wdeploy predeploy` in split mode to split source web applications into separate static and dynamic resources.

   For example:

   - Run the following command to extract static content for the Apache web server and dynamic content for WebLogic application server.

```
wdeploy.sh weblogic9 -Das_mode=split
    -Dws_type=apache predeployall
```

   Dynamic content is located in: `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/weblogic9/application`. Static content is located in: `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/weblogic9/resources`.

9. Run `wdeploy deployonlyall` to deploy the dynamic content to WebLogic application server and static content to Apache.

   If Apache and WebLogic are on the same machine, static and dynamic content will be automatically deployed to the servers with this command:

```
wdeploy.sh weblogic9 -Das_mode=split
    -Dws_type=apache deployonlyall
```

Note:

If your dynamic and static content are in a custom location, use the -Dwork_dir parameter.

If Apache and WebLogic are on different machines, dynamic content will be automatically deployed to WebLogic by the following command. Static content must then be manually copied over and deployed to the remote Apache machine.

```
./wdeploy.sh weblogic9 -Das_mode=split
    deployonlyall
```

Note:

If your dynamic and static content resides in a custom location, use the -Dwork_dir parameter.

Copy static content to the `htdocs` directory on the web server:

- Extract the zip files on the web application server under `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/WebLogic9/resources`.
- Copy these folders from the WebLogic web application server to the Apache `<WS_DIR>/htdocs` folder on the Apache server.
- Copy the `bobj.<application>.conf` files from the WebLogic web application server to the Apache `<WS_DIR>/conf` folder on the Apache server.
- Update Apache `httpd.conf` under `<WS_DIR>/conf` with the application configuration files. For each web application include an entry in `httpd.conf`. For example, to include BOE, you would enter:

```
Include conf/bobj.BOE.conf
```

Static content is now served by a dedicated web server, and dynamic content is served by a dedicated web application server.

3.6.3.2.2 Changes to installed languages

To add language support for web applications, you must re-install SAP BusinessObjects Enterprise and select the required language(s) during the installation process. Once the second installation is complete, any deployed web applications that require the new language(s) must be re-deployed to the web application server.

This is because the installation program does not redeploy web applications automatically. The web applications must be re-packaged and re-deployed to the web application server.

3.6.3.3 Tomcat

3.6.3.3.1 To deploy to separate Apache web and Tomcat web application servers

To reduce the load on a web application server, you can set up a separate, dedicated, web server to serve static content. All static content will be served by the web server, while dynamic content will be served by the web application server.

Note:
This configuration requires that you use the WDeploy command to split web application resources into static and dynamic content that can be deployed to separate web and web application servers.

1. Set up the Apache web server and ensure that it is working correctly. Load a web page, such as Apache's default test page, to verify that the web server is serving content correctly.

   Open a web browser and enter the web server URL. For example: `http://apache.mycompany.com`.

2. Ensure that your Tomcat web application server is working correctly.

   Open a web browser and enter the IP address or hostname of the web application server, and a port number. For example: `http://tomcat.mycompany.com:8080`.

   Note:
   If you have any existing SAP BusinessObjects Enterprise web applications running on the server, they must be undeployed before continuing.

3. Download the Apache Tomcat connector from the Tomcat web site. The Apache Tomcat connector allows you to connect an Apache web server with a Tomcat web application server, so Apache can forward requests for dynamic resources to Tomcat.
4. Follow the plug-in configuration instructions on the Apache web site for configuring the bridge between Apache web server and Tomcat web application server.

   Requests for dynamic resources are now forwarded to Tomcat when received by Apache.

5. Ensure that the bridge between the web server and web application server is working by pointing a browser to the web server and verifying that dynamic content from the web application server is served correctly.

   For example, visit the URL: `http://apache.mycompany.com/jsp-examples/`.

   Note:
   This example URL will only work if you have manually deployed the `jsp-examples` web application.

6. If the web application server is installed on the same machine as SAP BusinessObjects Enterprise, run WDeploy locally on that machine. If the web application server runs on a different machine copy the WDeploy command and environment to the web application server. See *To deploy web applications on a remote machine*.
7. Configure WDeploy environment to separate content between the web server and the web application server. This is known as "split" mode.

   The WDeploy configuration files for Apache and Tomcat are located in `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf`.

   - Edit the WDeploy configuration file for Tomcat. For example, for Tomcat 5.5, the file `config.Tomcat` will look similar to this:

     ```
     as_dir=/opt/Tomcat
     as_instance=localhost
     as_service_name=Tomcat5
     ```

   - Edit `config.apache`. For example:

     ```
     ws_dir=/opt/apache224
     connector_type=Tomcat
     deployment_dir=/opt/apache224/htdocs
     ```

8. Use `wdeploy predeploy` in split mode to separate source web applications into static and dynamic files.

   For example:

   - Run the following command to extract static content for the Apache web server.

     ```
     wdeploy.sh Tomcat -Das_mode=split
         -Dws_type=apache predeployall
     ```

9. Run `wdeploy deployonlyall` command to deploy the dynamic content to Tomcat application server and static content to Apache.

   If Apache and Tomcat are on the same machine, static and dynamic content will be automatically deployed to servers by the following command:

   ```
   wdeploy.sh Tomcat -Das_mode=split
       -Dws_type=apache deployonlyall
   ```

   Note:
   If your dynamic and static content are in a custom location, use the -Dwork_dir parameter.

If Apache and Tomcat are on different machines, dynamic content will be automatically deployed to Tomcat by the following command. Static content must be manually deployed to the remote Apache machine afterwards.

```
wdeploy.sh Tomcat -Das_mode=split
    deployonlyall
```

Note:
If your dynamic and static content are in a custom location, use the -Dwork_dir parameter.

Copy static content to the `htdocs` directory on the web server:

- Extract the zip files on the web application server under `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/Tomcat/resources`.
- Copy these folders from the Tomcat web application server to the Apache `<WS_DIR>/htdocs` folder on the Apache server.
- Copy the `bobj.<application>.conf` files from the Tomcat web application server to the Apache `<WS_DIR>/conf` folder on the Apache server.
- Update Apache `httpd.conf` under `<WS_DIR>/conf` with the application configuration files. For each web application include an entry in `httpd.conf`. For example, to include BOE, you would enter:

  ```
  Include conf/bobj.BOE.conf
  ```

Static content is now served by a dedicated web server, and dynamic content is served by a dedicated web application server.

3.6.3.3.2 Tomcat gzip compression

If you install a new installation of SAP BusinessObjects Enterprise and choose to use the bundled Tomcat web application server, Tomcat's HTTP gzip compression is enabled automatically.

The gzip compression improves web application server response time and throughput. However, if you have plan to deploy web applications to the Tomcat web application server, note that the deployment of web applications to a Tomcat server with gzip compression enabled may differ from the process used to deploy web applications to a version of Tomcat bundled with an earlier release of SAP BusinessObjects Enterprise.

3.6.3.3.3 Changes to installed languages

To add language support for web applications, you must re-install SAP BusinessObjects Enterprise and select the required language(s) during the installation process. Once the second installation is complete, any deployed web applications that require the new language(s) must be re-deployed to the web application server.

This is because the installation program does not redeploy web applications automatically. The web applications must be re-packaged and re-deployed to the web application server.

3.6.4 Split web tier pre-deployment without access to a web application server

The WDeploy command can separate static and dynamic content for deployment to a web server and web application server. Access to a web application server's deployment folder is not required: the separated content can be copied from the server hosting WDeploy to the web and web application servers manually.

When WDeploy is installed on a dedicated web server, the `wdeploy predeploy` or `wdeploy predeployall` commands are used to create static content from web applications directly to the web server's content directory. Some parameters are mandatory for particular web or web application servers:

Web or web application server	Parameters required for predeployment
WebSphere Application Server	as_virtual_host
Apache and IBM IHS	ws_dir, deployment_dir, connector_type

3.6.5 To disable the CMC or BI launch pad web applications

The Central Management Console (CMC) and BI launch pad (previously InfoView) web applications are now included within the `BOE` archive (`BOE.war` or `BOE.ear`). To disable either the CMC or BI launch pad web application, so either or both cannot be accessed when the `BOE` archive is deployed to a web application server, use the WDeploy tool.

To disable the CMC web application, use the `-Ddisable_CmcApp=true` switch when deploying `BOE.war` to the web applications server. For example, the following command will deploy BOE.war to a Tomcat 6 web application server but disable the CMC:

```
wdeploy.sh tomcat6 -DAPP=BOE -Ddisable_CmcApp=true deploy
```

To disable the BI launch pad web application, included the `-Ddisable_InfoView=true` switch when deploying `BOE.war` to the web applications server. For example, the following command will deploy BOE.war to a Tomcat 6 web application server but disable BI launch pad:

```
wdeploy.sh tomcat6 -DAPP=BOE -Ddisable_InfoView=true deploy
```

3.7 Using the WDeploy GUI tool

The WDeploy GUI tool is installed as a part of SAP BusinessObjects Enterprise and provides an alternative, graphical, method of running the `wdeploy deployall` or `wdeploy undeployall` commands.

For information on WDeploy requirements, see:

- WDeploy prerequisites
- WDeploy GUI tool prerequisites

To perform other deployment operations, use the command-line version of WDeploy.

The WDeploy GUI tool defaults to an English interface, but prompts the user to choose a language if any language packs are installed and the tool is run for the first time. The following language packs are supported by SAP BusinessObjects Enterprise:

- Czech
- Simplified Chinese
- Traditional Chinese
- Danish
- Dutch
- English
- Finnish
- French
- German
- Italian
- Japanese
- Korean
- Norwegian Bokmal
- Polish
- Portuguese
- Russian
- Spanish
- Swedish
- Thai

3.7.1 WDeploy GUI tool prerequisites

To use the WDeploy GUI tool, ensure that the following requirements are met:

- When using WebSphere 6 or WebSphere 7 with a non-default profile name (a profile name that isn't set to `AppSrv01`), manually update the `as_profile_name` parameter in the `config.websphereX` configuration file before launching the WDeploy GUI tool.

  For example:

  1. Update `as_profile_name` in `config.websphereX`
  2. Launch WDeploy GUI tool
  3. Select "WebSphere 6" or "WebSphere 7" and provide all parameters
  4. Perform the deployment

  For information on giving the profile name with the WDeploy command-line tool, see WebSphere 6 or 7 configuration file.

- Set the `JAVA_HOME` environment variable to the desired JDK folder. For example, `JAVA_HOME` may be set to:

  `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/java`

  Or:

  `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/<PLATFORM>64_x64/jdk`

- Ensure that the host system has at least 5 GB of free space for the deployment of web applications.

  For example, ensure that there is at least 5 GB available in `/tmp`.

- Ensure that the web application server host system has at least 15 GB of free hard disk space before attempting to deploy web applications.
- Ensure that the host system has at least 4 GB of RAM. When using SAP NetWeaver AS Java 7.2, ensure that the host system has at least 8 GB of RAM.
- Ensure that the host web application server minimum heap size (`-Xms`), maximum heap size (`-Xmx`), and Permanent Generation (`-XX:MaxPermSize`) settings are configured. For example:

  ```
  JAVA_OPTS=-Xms128m -Xmx1024m -XX:MaxPermSize=512
  ```

  Note:
  For SAP NetWeaver AS 7.2, ensure that the maximum heap size is at least 2048 megabytes:

  ```
  JAVA_OPTS=-Xms128m -Xmx2048m -XX:MaxPermSize=512
  ```

- Set the `PERL_HOME` environment variable.

  If you do not already have Perl 5.8.9 installed, set `PERL_HOME` to `<BOE_INSTALL_DIR>/InstallData/InstallCache/tp.perl-5.8.9-core-32/0/perl`.

  It is recommended that you use the version of Perl installed with SAP BusinessObjects Enterprise. To use your own version of Perl 5.8.9, ensure that the following modules are also installed:

  - `Text::CharWidth`

    Gets the number of occupied columns of a string on terminal. For more information, see the `CharWidth.pm` article at http://search.cpan.org.

  - `Text::WrapI18N`

Line wrapping module with support for multibyte, fullwidth, and combining characters and languages without whitespaces between words. For more information, see the `WrapI18N.pm` article at http://search.cpan.org.

3.7.2 Starting WDeploy GUI tool

The WDeploy GUI tool is located in the folder `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy`. The syntax of the WDeploy command is:

```
./wdeployGUI.sh
```

When the WDeploy GUI tool is run for the first time, it prompts for which language to use for displaying information in the user interface. Select a language and continue.

3.7.3 WDeploy GUI tool window

The WDeploy GUI tool allows users to deploy and undeploy web applications to a web application server. Any web applications not deployed by the SAP BusinessObjects Enterprise installation program or WDeploy tool will be unaffected.

Select a supported web application server from the **Select Web Application Server Type** drop-down. Before deploying or undeploying web applications to a web application server, ensure that the WDeploy web application server configuration file (`config.<WEB_APP_SERVER>`) is correctly configured. For more information, see WDeploy configuration files.

When a web application server is selected, review the information in the "Web Application Server Information" section and the "Application Server Domain Root Directory" section to ensure that it is correct for your web application server.

For more advanced options, select **Options**.

Note:
Not all web applications are deployed automatically. Web applications that are not deployed automatically must be deployed with the WDeploy command-line tool or with the web application server administrative console.

Web application archive (may be WAR or EAR)	Deployed automatically?
BOE	Yes
BusinessProcessBI (deprecated)	Yes
rebean3ws	Yes
dswsbobje	Yes
MobileOTA14	No
jsfplatform	No
OpenSearch	No
AdminTools	Yes

3.7.4 WDeploy GUI tool options

The WDeploy GUI tool Options screen allows you to select custom folders to use for the deployment of web applications. This is equivalent to setting properties when using the WDeploy command-line. For more information on WDeploy parameters, see WDeploy property parameters.

- "WDeploy work directory" (equivalent to the `work_dir` parameter).
- "Web application source tree location" (equivalent to the `app_source_tree` parameter) or "Generic WAR files location" (equivalent to the `war_dir` parameter).
- "Web applications root context" (equivalent to the `root_context_path` parameter).

Note:
Non-English characters in paths are currently not accepted as valid by the WDeploy GUI tool.

3.8 After deploying web applications

After deploying or undeploying web applications, restart the web application server.

To verify the web application server configuration, make sure that BI launch pad and the Central Management Console (CMC) can be launched in a web browser. For example:

- `http[s]://<WEB_APP_SERVER>:<PORT>/<BI_LAUNCHPAD_CONTEXT>`
- `http[s]://<WEB_APP_SERVER>:<PORT>/<CMC_CONTEXT>`

Replace *<WEB_APP_SERVER>* with the hostname or IP address of the web or web application server, and *<PORT>* with the port number used for either HTTP or HTTPS communication. The default root context used for BI launch pad is `/BOE/BI`; the default context used for the CMC is `/BOE/CMC`.

3.9 Log files

The WDeploy tool creates two log files with detailed information that may be useful for diagnosing problems or monitoring activity.

1. <*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/logs/WDeploy_summary.log: a high-level record of WDeploy activity that includes information on the version of WDeploy, the JDK being used, the command issued, configuration, WAR file location, actions or events and their outcomes, and any error messages or warnings displayed to the user at runtime.
2. <*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/logs/WDeploy.log: includes everything in WDeploy_summary.log, but also includes timestamps and any warning or error messages.

When a WDeploy command is issued, WDeploy_summary.log.01 is overwritten with the current command, and WDeploy.log is appended.

As the WDeploy.log file grows to more than 512 KB, the file is renamed with a unique numbered suffix. For example, WDeploy.log will be renamed to WDeploy.log.01. If a file named WDeploy.log.01 already exists, WDeploy.log.02 will be created.

3.10 Web application updates made by WDeploy

Table 3-23: Changes made to web application configuration by WDeploy

Server	Changes made
Apache web server	For each application, ${ws_dir}/conf/bobj.${APP}.conf file is created, containing connector configuration, directory and servlet mapping information. File ${ws_dir}/conf/httpd.conf is modified to include that file.

Server	Changes made
Tomcat	Files added to the `classpath` are dropped in `${as_dir}/shared/lib` `${as_dir}/bin/bobjeEnv.${APP}.[sh\|bat]` file is created, containing library path modifications, and java properties modifications to add the application's properties. `${as_dir}/bin/catalina.[sh|bat]` file is modified to include this file. On Windows, additional java properties are added to the Tomcat service startup parameters. Library path modifications must manually be made to the system-wide PATH environment variable. Restart Tomcat to take changes into account.
JBoss	The JSF package shipped with JBoss is deleted if it is detected (directory `${as_dir}/server/${as_instance}/deploy/jbossweb-tomcat55.sar/jsf-libs` is deleted). Files added to the `classpath` are dropped in `${as_dir}/server/${as_instance}/lib`. File `${as_dir}/bin/bobjeEnv.${APP}.[sh|bat]` is created, containing library path modifications, and java properties modifications to add the application's properties. File `${as_dir}/bin/run.[sh|bat]` is modified to include this file. Restart JBoss to take changes into account.
NetWeaver	Files to add to the `classpath` are bundled in the application (added to `WEB-INF/lib`). Additional library path and application properties modifications are not supported yet. Such modifications must be done manually using NetWeaver's configuration interface.
Oracle	Files to add to the `classpath` are bundled in the application (added to `WEB-INF/lib`). Library path and application specific properties are set into `${as_dir}/opmn/conf/opmn.xml`. Restart your application server to take changes into account.

Server	Changes made
WebLogic	Files to add to the `classpath` are bundled in the application (added to `WEB-INF/lib`). `${as_domain_dir}/bin/bobjeEnv.${APP}.[sh|cmd]` file is created, containing library path and java properties modifications. `${as_domain_dir}/bin/startWebLogic.[sh|cmd]` file is modified to include this file.
WebSphere	Files to add to the `classpath` are bundled in the application (added to `WEB-INF/lib`). Properties are added as JVM custom properties. Library path modifications are done modifying the environment of the JVM. Web Server plugin gets automatically regenerated.

Table 3-24: Changes made to Sun Java configuration by WDeploy

Server	Changes made

Sun Java System Application Server	Properties are added as JVM options. Library path modifications are done modifying JVM's native library path suffix. The `java.policy` file is modified to add permissions as follows:
	`"grant {`
	`permission java.util.PropertyPermission "*", "read,write";`
	`permission java.lang.RuntimePermission "createClassLoader";`
	`permission java.lang.RuntimePermission "getClassLoader";`
	`permission java.lang.RuntimePermission "shutdownHooks";`
	`permission java.io.FilePermission "<business objects installation directory>", "read,write,execute";`
	`permission java.net.SocketPermission "*", "connect,listen,accept,resolve"; };"`

Sun Java System Web Server	For each application, `${ws_dir}/obj.conf` file is modified, to add connector configuration, directory and servlet mapping information. `NameTrans fn=pfx2dir from="${appvdir}" dir="${deployment_dir}/${APP}"`

To deploy web applications with the administrative console

4.1 To manually deploy web applications

The following steps will deploy a web application using the application server's administrative console.

The WDeploy tool must be used to generate deployable WAR or EAR files before they can be deployed to your web application server.

1. Run the `wdeploy predeploy` command to prepare a single web application, or `wdeploy predeployall` to prepare all web applications.

   Example 1: using `wdeploy predeploy` to prepare the BOE.war web application WAR file for deployment to Tomcat.

   ```
   wdeploy.sh tomcat55
     -Das_dir=/opt/usr/tomcat
     -Das_instance=localhost
     -Das_service_name=Tomcat55
     -DAPP=BOE
     predeploy
   ```

   Example 2: using `wdeploy predeployall` to prepare all web applications for deployment to Tomcat.

   ```
   wdeploy.sh tomcat55
     -Das_dir=/opt/usr/tomcat
     -Das_instance=localhost
     -Das_service_name=Tomcat55
     predeployall
   ```

   Note:
   The `predeploy` and `predeployall` commands can be used on machines that do not host a web application server, as long as the parameters required by the `predeploy` or `predeployall` commands are given.

   The WDeploy command will prepare web applications for deployment on a web application server, and creates WAR or EAR files in `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/<APP_SERVER>/application` unless you provide the -Dwork_dir parameter.

2. Follow the manual deployment procedure specific to your web application server.

Your web application can now be manually deployed on your web application server.

Repeat these steps for any other web applications that need to be installed.

4.1.1 JBoss 4.2.3 manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

The JMX administrative console does not deploy web applications to JBoss automatically. To deploy web applications to JBoss, you must manually copy the web application file into the `deploy` directory.

The JavaServer Faces (JSF) bundled with the JBoss web application servers must be disabled for the Central Management Console (CMC) and BI launch pad web clients to work correctly.

To disable JSF, the following directories must be removed and JBoss restarted:

- *JBOSS_HOME*`/server/default/deploy/jbossweb-tomcat55.sar/jsf-libs`
- *JBOSS_HOME*`/server/default/tmp`
- *JBOSS_HOME*`/server/default/work`

Note:
To use the Java `log4j` logging included with SAP BusinessObjects Enterprise, refer to section 10.3.7 *Using your own log4j.properties file - class loader scoping* in the *JBoss Development Process Guide*: http://docs.jboss.org/process-guide/en/html/logging.html#d0e3341.

The following deployment paths are based upon the different levels of service:

- *JBOSS_HOME*`/server/all/deploy`
- *JBOSS_HOME*`/server/default/deploy`
- *JBOSS_HOME*`/server/minimal/deploy`

1. Copy your WAR files to the appropriate context root sub-directory under one of the above paths, based upon the level of service provided by JBoss for your deployment.

   The web application is automatically deployed by JBoss when the file is copied to the appropriate directory.

2. Check the JBoss server log, and you should see a message similar to the one shown below to confirm that the WAR deployment succeeded.

```
09:54:28,703 INFO [TomcatDeployer] deploy, ctxPath=/BOE,
warUrl=.../tmp/deploy/tmp43109BOE-exp.war/
```

4.1.2 JBoss 5 manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

The JMX administrative console does not deploy web applications to JBoss automatically. To deploy web applications to JBoss, you must manually copy the web application file into the `deploy` directory.

The JavaServer Faces (JSF) bundled with the JBoss web application servers must be disabled for the Central Management Console (CMC) and BI launch pad web clients to work correctly.

To disable JSF, the following directories must be removed and JBoss restarted:

- *JBOSS_HOME*/server/default/deploy/jbossweb-tomcat55.sar/jsf-libs
- *JBOSS_HOME*/server/default/tmp
- *JBOSS_HOME*/server/default/work

Note:
To use the Java `log4j` logging included with SAP BusinessObjects Enterprise, refer to section 10.3.7 *Using your own log4j.properties file - class loader scoping* in the *JBoss Development Process Guide*: http://docs.jboss.org/process-guide/en/html/logging.html#d0e3341.

The following deployment paths are based upon the different levels of service:

- *JBOSS_HOME*/server/all/deploy
- *JBOSS_HOME*/server/default/deploy
- *JBOSS_HOME*/server/minimal/deploy

1. Copy your WAR files to the appropriate context root sub-directory under one of the above paths, based upon the level of service provided by JBoss for your deployment.

   The web application is automatically deployed by JBoss when the file is copied to the appropriate directory.

2. Check the JBoss server log, and you should see a message similar to the one shown below to confirm that the WAR deployment succeeded.

```
09:54:28,703 INFO [TomcatDeployer] deploy, ctxPath=/BOE,
warUrl=.../tmp/deploy/tmp43109BOE-exp.war/
```

4.1.2.1 To deploy the dswsbobje web application to JBoss 5

To deploy the `dswsbobje` web application to JBoss 5, follow the steps below.

1. Generate the `dswsbobje` web application with the `wdeploy predeploy` command.
2. Locate the `dswsbobje` web application in exploded form in the following folder:

   <*BOE_INSTALL_DIR*>/sap_bobj/enterprise_xi40/wdeploy/workdir/jboss5/application

3. Copy the dswsbobje folder to the JBoss deployment folder:

   <JBOSS_HOME>/server/default/deploy

The web application server should automatically deploy `dswsbobje`. If not, restart JBoss.

4.1.2.2 To deploy the BusinessProcessBI web application to JBoss 5

To deploy the `BusinessProcessBI` web application to JBoss 5, follow the steps below.

1. Generate the `BusinessProcessBI` web application with the `wdeploy predeploy` command.
2. Locate the `BusinessProcessBI` web application in exploded form in the following folder:

   `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/jboss5/application`
3. Copy the `BusinessProcessBI` folder to the JBoss deployment folder:

   `<JBOSS_HOME>/server/default/deploy`

The web application server should automatically deploy `BusinessProcessBI`. If not, restart JBoss.

4.1.3 Oracle Application Server 10g R3 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

Create an Oracle container for Java (OC4J) container in which to run SAP BusinessObjects Enterprise web applications.

1. Open the Oracle Application Server Enterprise Manager server page at `http://WAS_HOSTNAME:PORT/em`. Replace *WAS_HOSTNAME* with the hostname or IP address of your web application server, and *PORT* with the port number used for HTTP. The default port for Oracle Application Server is 7777.

   The Oracle Application Server Control page is displayed.
2. click the **home** OC4J container.

   The **home** group is displayed under the **Groups** heading.
3. In the **home** OC4J container, click the **Applications** tab.

   Currently deployed web applications are displayed.
4. click the **Deploy** button.

   The **Deploy: Select Archive** window is displayed.
5. Ensure that the **Archive is present on local host option** is selected, unless the web application is already running on the server, in which case select **Archive is already present on the server where Application Server Control is running**.
6. In the "Deployment Plan" section, ensure that **Automatically create a new deployment plan** is selected.

7. Enter the full path, or browse, to the WAR file in the **Archive Location** field.
8. Press the **Next** button to advance to the next step in the WAR deployment wizard.
   Step 2 of 3, **Deploy: Application Attributes** is displayed.
9. Enter the application name and context root of the file you want to deploy. When ready, press **Next** to proceed.
   The Step 3 of 3, **Deploy: Deployment Settings** page is displayed.
10. If your web application requires any class libraries, click **Configure Class Loading**, navigate to **Configure Web Mobule Class Loaders**, and provide the library class path(s).
    If your web application does not require class libraries, proceed to the next step.
11. Click **Deploy**.
    Check the output of the "Progress Messages" log field to ensure that the web application deployed.

4.1.4 SAP NetWeaver AS Java 7.2 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

SAP NetWeaver Developer Studio 7.1 is required to deploy web applications with the SAP NetWeaver AS Java 7.2 administrative console. SAP NetWeaver Developer Studio can be installed on the same system that hosts SAP BusinessObjects Enterprise, or on a separate system with network access to both the web application server, and the EAR files to be deployed.

1. Start the "SAP NetWeaver Developer Studio" application.
2. If this is your first time using SAP NetWeaver Developer Studio with this instance of SAP NetWeaver AS Java 7.2 register the web application server in the Developer Studio preferences.
   a. Select **Window > Preferences > SAP AS Java**.
   b. Enter the hostname or IP address of the SAP NetWeaver Java AS 7.1 server in the "Instance host" field.
   c. Enter the SAP NetWeaver AS Java 7.1 server instance number in the "Instance number" field.
      For example, if the server instance ID is `J00`, the instance number is `00`.
   d. Click **Register SAP Instance**.
      The "SAP system" selection field is populated, and server instances are displayed in the "SAP System instances" list.
3. Open the "Deploy View" tab. Select **Window > Show View > Other... > Deploy View > Deploy View**.
   Preferences for the Deploy View tab can be set in **Window > Preferences > SAP AS Java > Deploy View**.
4. Select "External Deployable Archives" and review the available options.
5. Click **Import Java EE Archives** in the toolbar.
6. Select the web application EAR file.

Packaged SAP BusinessObjects Enterprise web applications are located in `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/sapappsvr72`.

The EAR appears in the list of external deployable archives.

7. Click **Deploy** in the toolbar.

   If this is the first time you are deploying a web application to this server, you are prompted to log on to the SAP J2EE Engine. Enter a valid username and password (such as the SAP NetWeaver Administrator account).

   The web application is deployed to SAP NetWeaver AS Java 7.2, and a confirmation dialog appears. The web application's icon in the "External Deployable Archives" tree displays a green checkmark.

8. Confirm that the web application has started by selecting the Repository View tab with **Window > Show View > Other... > Deploy View > Repository View**.

4.1.4.1 To update web.xml for SAP NetWeaver AS Java 7.2

In addition to using `wdeploy predeploy` ensure that the web application's `web.xml` file is updated from servlet specification 2.3 to servlet specification 2.4. This update is currently not performed by `WDeploy` for SAP NetWeaver AS Java 7.2 deployments.

Caution:
This process requires a significant technical understanding of XML code and the differences between servlet specifications 2.3 and 2.4. Each web application's `web.xml` file is unique and requires different changes.

Follow these steps to update the `web.xml` file for each web application before it is deployed.

1. Use the `wdeploy predeploy` command to create an EAR file that can be deployed to SAP NetWeaver AS Java 7.2.

   For example, the command `wdeploy.sh sapappsvr72 -DAPP=BOE predeploy` creates the file `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/workdir/sapappsvr72/application/BOE.ear`.

2. Extract the EAR file to a temporary folder.

   For example, use the command `jar xvf BOE.ear -C ivatmp1` to extract the EAR file to a directory named `ivatmp1`.

   Note:
   The Java `bin` directory must be included in your `PATH` environment variable to use the `jar` command.

   The `jar` command is located in `<BOE_INSTALL_DIR>/javasdk/bin/jar`.

3. Extract the WAR file resulting from the previous step to a different temporary folder.

   For example, use the command `jar xvf ivatmp1/BOE.war -C ivatmp2` to extract the WAR file to a directory named `ivatmp2`.

4. Use a text, XML, or IDE editor to open the web application's `web.xml` file.

   The file is located in `ivatmp2/WEB-INF/web.xml`.

5. Ensure that XML elements within `web.xml` conform to the Servlet 2.4 specification.

   For more information on the Servlet 2.4 specification, and how it compares to the 2.3 specification, visit http://java.sun.com/products/servlet/download.html.

   For example, the 2.4 specification requires that the `<taglib>` tag comes under the `<jsp-config>` tag, whereas the Servlet 2.3 specification puts the `<taglib>` tag under the `<webapp>`.

   For illustration purposes only, below is a diff listing for the differences between a `web.xml` servlet specification 2.3 and one that has been updated to servlet specification 2.4. For information on how to read diff listings, visit http://www.gnu.org/software/diffutils/manual/.

```
$ diff PlatformServices_web.xml_2-3 PlatformServices_web.xml_2-4
1c1,4
< <?xml version="1.0"?>
---
> <?xml version="1.0" encoding="UTF-8"?>
> <web-app id="WebApp_9" version="2.4" xmlns="http://java.sun.com/xml/ns/j2ee"
>       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
>       xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-
app_2_4.xsd">
3,8d5
< <!DOCTYPE web-app PUBLIC
<   "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
<   "http://java.sun.com/dtd/web-app_2_3.dtd">
<
<
< <web-app>
484,493d480
<     <servlet>
<         <servlet-name>Not Found Servlet</servlet-name>
<         <servlet-class>com.businessobjects.webutil.ForwardServlet</servlet-class>
<         <init-param>
<             <param-name>url</param-name>
<             <param-value>/httperror_404.htm</param-value>
<         </init-param>
<         <load-on-startup>4</load-on-startup>
<     </servlet>
<
527,536d513
<
<     <servlet-mapping>
<         <servlet-name>Not Found Servlet</servlet-name>
<         <url-pattern>*.jspf</url-pattern>
<     </servlet-mapping>
<
<     <servlet-mapping>
<         <servlet-name>Not Found Servlet</servlet-name>
<         <url-pattern>*.inc</url-pattern>
<     </servlet-mapping>
560a538
>     <jsp-config>
592a571,572
>     </jsp-config>
>
```

6. Create a new WAR file from the updated contents. For example, the command `jar cvf BOE.war .` run within the `ivatmp2` directory from step 3 above creates a new WAR file `BOE.war` in the `tmp` directory.

   A new WAR file containing an updated `web.xml` is created.

7. Copy the WAR file to the temporary directory used to extract the EAR file. Overwrite the previous WAR file.

   The directory that contains the extracted EAR file contents now contains the WAR file with the updated `web.xml`.

8. Create a new EAR file from the updated contents. For example, the command `jar cvf BOE.war .` run within the `ivatmp1` directory from step 2 above creates a new WAR file `BOE.war` in the `tmp` directory.

   A new EAR file is created.

4.1.5 Sun Java System Application Server 9.1 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

Ensure that your Sun Java System Application Server web application server is installed, configured, and running before deploying WAR files.

Note:
Sun Java System Application Server imposes a strict default security policy. You must modify the security policy before the BI launch pad and Central Management Console (CMC) can operate normally:

1. Edit `<SUNONE_INSTALL_DIR>/domains/domain1/config/server.policy`
2. Assign write permission to `java.util.PropertyPermission`.
3. Assign execute and delete permission to `java.io.FilePermission`.
4. Assign listen and accept permissions to `java.net.SocketPermission`.

For example:

```
// Basic set of required permissions granted to all remaining code

grant {
        permission java.lang.RuntimePermission  "loadLibrary.*";
        permission java.lang.RuntimePermission  "queuePrintJob";
        permission java.net.SocketPermission    "*", "connect, listen, accept";
        permission java.io.FilePermission       "<<ALL FILES>>", "read, write, execute, delete";

        // work-around for pointbase bug 4864405
        permission java.io.FilePermission "${com.sun.aas.instanceRoot}${/}lib${/
}databases${/}-", "delete";
        permission java.io.FilePermission "${java.io.tmpdir}${/}-", "delete";

        permission java.util.PropertyPermission "*", "read, write";

        permission java.lang.RuntimePermission  "modifyThreadGroup";
};
```

If you are deploying Performance Management applications to Sun Java System Application Server, you must first edit Sun Java's startup configuration. For more information, see “To deploy Performance Management to Sun Java System Server”.

Follow this procedure for each WAR file that you want to deploy.

1. Log on to the Sun Java System Server administrative console.

   For example: `https://<YourServer>:4849/asadmin`, where YourServer is the name of the system that the Domain Admin Server is running on. Port 4849 is the default administrative console's default port number. If you changed the port number when you installed the Sun Java System Server administrative console, use that port number instead.

Note:
Because the administrative console is a secure web application, you must use HTTPS instead of HTTP.

2. Under Common Tasks, select **Applications > Web Applications**.
3. On the right-hand pane, click **Deploy**.
4. On the "Deploy Web Module" page:
   a. If you are accessing the administrative console locally, select **Specify a package file or a directory path that must be accessible from the server** and browse to the WAR file to deploy.
   b. If you are accessing the administrative console remotely, select **Specify a package file or a directory path that must be accessible from the server** and enter the full path to the WAR file.
5. On the next "Deploy Web Module" page, type the application name and context root of the web application that you want to deploy. In the "Targets" area, select the instance to deploy under "Available" and add it to the "Selected" area.

   Note:
   It is recommended that you use the application name as the context root.

6. Click **OK**.

   The WAR file that you chose is deployed. Repeat steps 4 to 7 for any additional WAR files that you want to deploy.

7. To grant the web application access to web services, add the following lines at the end of the `server.policy` file, located in the `Sun\AppServer\domains\domain1\config` directory:

```
grant codeBase "file:${com.sun.aas.instanceRoot}/domains/domain1/applications/j2ee-modules/BusinessPro
cessBI/-"
{
      permission java.security.AllPermission;
};

grant codeBase "file:${com.sun.aas.instanceRoot}/domains/domain1/applications/j2ee-modules/dswsbobje/-"
{
      permission java.security.AllPermission;
};
```

4.1.6 Tomcat 5.5 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

Log on to the "Tomcat Manager Console".

- `http://`*`WAS_HOSTNAME`*`:`*`PORT`*`/manager/html`

1. Set the **Context Path** for the web application to be deployed. The context path must be the name of the WAR file, but without its extension. For example, to deploy the CMC web application (`BOE.war`), the context path must be `BOE`.

2. Set the **XML Configuration File** setting to be an XML file that contains the context path and document base. For example: `<Context docBase="<M>/WEB_APPLICATION.war" path="<context_path>" crossContext="false" debug="0" reloadable="false" trusted="false"/>`
3. Enter the full path to the WAR file and press the **Deploy** button
   The WAR file is deployed.

4.1.6.1 Special considerations for deploying on Tomcat clusters

To manually deploy web applications to a Tomcat web application cluster distributed over multiple machines, deploy the web applications to each Tomcat web application server. Use the administrative console to deploy web applications to the `webapps` subfolder in the Tomcat home directory for each server instance.

Note:
When using Tomcat, stop the web application server before issuing the `wdeploy undeployall` action command.

Tip:
It is recommended that you install and configure a hardware or software load balancer if running SAP BusinessObjects Enterprise web applications on a Tomcat web application cluster. Consult the Tomcat documentation for information on setting up load balancing on a Tomcat cluster.

4.1.6.2 Changes to installed languages

To add language support for web applications, you must re-install SAP BusinessObjects Enterprise and select the required language(s) during the installation process. Once the second installation is complete, any deployed web applications that require the new language(s) must be re-deployed to the web application server.

This is because the installation program does not redeploy web applications automatically. The web applications must be re-packaged and re-deployed to the web application server.

4.1.7 Tomcat 6 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

Note:
Tomcat 6 is supported only when using release 6.0.18 or newer.

Log on to the "Tomcat Manager Console".

- `http://WAS_HOSTNAME:PORT/manager/html`

1. Set the **Context Path** for the web application to be deployed. The context path must be the name of the WAR file, but without its extension. For example, to deploy a web application packaged as *YOUR_WEB_APPLICATION*`.war`, the context path must be *YOUR_WEB_APPLICATION*.
2. Set the **XML Configuration File** setting to be an XML file that contains the context path and document base. For example: `<Context docBase="<M>/YOUR_WEB_APPLICATION.war" path="<context_path>" crossContext="false" debug="0" reloadable="false" trusted="false"/>`
3. Enter the full path to the WAR file and press **Deploy**.
   The WAR file is deployed.

4.1.7.1 Special considerations for deploying on Tomcat clusters

To manually deploy web applications to a Tomcat web application cluster distributed over multiple machines, deploy the web applications to each Tomcat web application server. Use the administrative console to deploy web applications to the `webapps` subfolder in the Tomcat home directory for each server instance.

Note:
When using Tomcat, stop the web application server before issuing the `wdeploy undeployall` action command.

Tip:
It is recommended that you install and configure a hardware or software load balancer if running SAP BusinessObjects Enterprise web applications on a Tomcat web application cluster. Consult the Tomcat documentation for information on setting up load balancing on a Tomcat cluster.

4.1.7.2 Changes to installed languages

To add language support for web applications, you must re-install SAP BusinessObjects Enterprise and select the required language(s) during the installation process. Once the second installation is complete, any deployed web applications that require the new language(s) must be re-deployed to the web application server.

This is because the installation program does not redeploy web applications automatically. The web applications must be re-packaged and re-deployed to the web application server.

4.1.8 WebLogic 9.2 MP2 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

Create a WebLogic domain in which to run SAP BusinessObjects Enterprise web applications.

Note:

- Deployable web applications for WebLogic are in the following format in the WDeploy `workdir` folder:
  - Web Services web applications (`BusinessProcessBI` and `dswsbobje`) folders
  - All other web applications are WAR files.
- WebLogic deployments must use an extracted `dswsbobje.war`. See To deploy dswsbobje.war to WebLogic.

1. Open the "WebLogic Administrative Console" with a web browser at a `http://`*`WAS_HOSTNAME`*`:`*`PORT`*`/console` where *`WAS_HOSTNAME`* is the name of your WebLogic server and *`PORT_NUMBER`* is the port number on which the admin server listens.
   The "WebLogic Server Administration Console" web page is displayed.
2. In the left-hand navigation pane, click **Domain Structure** > **Deployments** .
   The "Summary of Deployments" page is displayed.
3. Click **Lock and Edit** in the left-hand pane.
   The domain is locked for editing.
4. Press the **Install** button and navigate to the WAR file you want to deploy. Select the **Install this deployment as an application** option. When you've selected the file and given any other parameters in the wizard, **Finish** to deploy the file.
   The WAR file is deployed to the WebLogic 9.2 MP2 server.
5. Press the **Activate Changes** button to apply the changes to the web application server.
   Your changes are saved.
6. When your WAR file has been deployed, press the **Start** button to initiate its execution.
   The web application you deployed will now start.

4.1.8.1 To deploy dswsbobje.war to WebLogic

Before deploying the web services WAR file on a WebLogic web application server, the user must extract `dswsbobje.war`. If the WAR file is not extracted and redeployed, web services applications may display the error message “An XSD Exception occurred”.

1. Extract `dswsbobje.war` to a directory on the WebLogic web application server.
2. Log on to the "WebLogic Administration Console".
3. Click **Lock & Edit**.
4. Go to **Domain Structure > Deployments** and click **Install**.
5. Browse to the extracted dswsbobje.war directory.
6. Select the directory and click **Next**.
7. Select **Install this deployment as an application** and click **Next**.
8. Verify the settings and click **Finish**.

   A message will be displayed when the dswsbobje.war web application has been successfully deployed. The list of web applications contains the dswsbobje web application.
9. Select the dswsbobje web application and click **Start**.
10. Click **Activate Changes**.
11. Select the dswsbobje web application and click **Start**.
12. Press **OK** to confirm the changes.

    The dswsbobje web application is started.
13. Open the web application in a web browser to confirm that it now runs.

4.1.8.2 Changes to installed languages

To add language support for web applications, you must re-install SAP BusinessObjects Enterprise and select the required language(s) during the installation process. Once the second installation is complete, any deployed web applications that require the new language(s) must be re-deployed to the web application server.

This is because the installation program does not redeploy web applications automatically. The web applications must be re-packaged and re-deployed to the web application server.

4.1.9 WebLogic 10 and 10.3 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create WAR files that can be deployed.

Created a WebLogic domain in which to run SAP BusinessObjects Enterprise web applications.

Note:

- Deployable web applications for WebLogic are in the following format in the WDeploy `workdir` folder:

  - Web Services web applications (`BusinessProcessBI` and `dswsbobje`) are folders.
  - The remaining web applications are in WAR file format.
- WebLogic deployments must use an extracted `dswsbobje.war`. See To deploy dswsbobje.war to WebLogic.

1. Open the "WebLogic Administrative Console" with a web browser at a `http://`*`WAS_HOSTNAME`*`:`*`PORT`*`/console` where *`WAS_HOSTNAME`* is the name of your WebLogic server and *`PORT`* is the port number on which the admin server listens.

   The "WebLogic Server Administration Console" web page is displayed.
2. In the left-hand navigation pane, click **Domain Structure** > **Deployments**.

   The "Summary of Deployments" page is displayed.
3. Press the **Install** button and navigate to the WAR file. When you've selected the file and given any other parameters in the wizard, click **Finish**.

   The WAR file is deployed.
4. Press the **Activate Changes** button to apply the changes to the web application server.

   Your changes are saved.
5. When your WAR file has been deployed, press **Start**.

   The web application starts.

4.1.9.1 To deploy dswsbobje.war to WebLogic

Before deploying the web services WAR file on a WebLogic web application server, the user must extract `dswsbobje.war`. If the WAR file is not extracted and redeployed, web services applications may display the error message “An XSD Exception occurred”.

1. Extract `dswsbobje.war` to a directory on the WebLogic web application server.
2. Log on to the "WebLogic Administration Console".
3. Click **Lock & Edit**.
4. Go to **Domain Structure** > **Deployments** and click **Install**.
5. Browse to the extracted dswsbobje.war directory.
6. Select the directory and click **Next**.
7. Select **Install this deployment as an application** and click **Next**.
8. Verify the settings and click **Finish**.

   A message will be displayed when the dswsbobje.war web application has been successfully deployed. The list of web applications contains the dswsbobje web application.
9. Select the dswsbobje web application and click **Start**.
10. Click **Activate Changes**.
11. Select the dswsbobje web application and click **Start**.
12. Press **OK** to confirm the changes.

The dswsbobje web application is started.

13. Open the web application in a web browser to confirm that it now runs.

4.1.9.2 Changes to installed languages

To add language support for web applications, you must re-install SAP BusinessObjects Enterprise and select the required language(s) during the installation process. Once the second installation is complete, any deployed web applications that require the new language(s) must be re-deployed to the web application server.

This is because the installation program does not redeploy web applications automatically. The web applications must be re-packaged and re-deployed to the web application server.

4.1.10 WebSphere 6.1 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create EAR files that can be deployed.

Log in to the "WebSphere Application Server Administration" console using the following URL: `http://`*`WAS_HOSTNAME`*`:`*`PORT`*`/admin`. The WebSphere administrative console's default port number is 9060.

1. Under the **Applications** heading of the console navigation menu, click **Enterprise Applications** on the left.
2. Click **Install** and navigate to the location of the EAR file to deploy. If deploying from a remote file system, select the Remote File System option.
3. Enter a context root for the EAR file (e.g. `/BOE` for `BOE.ear`) and press the **Next** button.
4. On the "Step 1" screen, give a unique name for your web application and proceed to the "Step 2" screen.
5. Under "Map Modules to Servers", highlight the server you created (or highlight **server1** if you didn't create your own) from the **Clusters and Servers** and enable the **Select** checkbox. Proceed to the "Step 3" screen.
6. Select the virtual host you created (or **default_host** if you didn't create your own) from the **Virtual Host** drop-down list. Proceed to the "Step 4" screen.
7. Review the summary page, and press **Finish** when done.
8. Click **Save to Master Configuration**.
9. Click **Save**, then the **Save** button.
10. Under the **Applications** heading of the console navigation menu, click **Enterprise Applications** on the left.
11. Verify that the EAR file was deployed, select **Start**.

Repeat steps 1 to 11 for each web application being deployed.

4.1.10.1 To change the WebSphere 6 deployment time-out setting

The deployment of SAP BusinessObjects Enterprise web applications to a WebSphere 6 web application server may take some time.

If you receive a time-out message while deploying web applications to a WebSphere 6, increase the deployment time-out setting.

1. Use a text editor to edit the configuration file `deployment.xml`. The location of `deployment.xml` will vary depending on how your server was installed and configured.

   The `deployment.xml` configuration file is typically located in `<WAS_HOME>/systemApps/adminconsole.ear/deployment.xml`.

2. Set the attribute `invalidationTimeout` to the desired value, in minutes, where the maximum value is `-1` (do not time out).
3. Save `deployment.xml`.
4. Restart the WebSphere service.

The WebSphere time-out setting has been changed. For more information, see: https://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.express.doc/info/exp/isc/cons_sessionto.html.

4.1.10.2 Disable WebSphere JSF implementation

If pages rendered by WebSphere do not show cached values correctly, it may be using its own version of the JavaServer Faces (JSF) library, rather than the JSF implementation supplied with SAP BusinessObjects Enterprise. To correct this problem, manually reconfigure WebSphere for any web application that uses JSF. For example, the BOE web application uses JSF.

To disable the JSF implementation supplied with WebSphere, follow the instructions below.

1. Log in to the "WebSphere Application Server Administration" console using the following URL: `http://WAS_HOSTNAME:PORT/admin`. The WebSphere administrative console's default port number is 9060.
2. Browse to **YOUR_WEB_APPLICATION > Manage Modules > YOUR_WEB_APPLICATION.ear**
   Replace **YOUR_WEB_APPLICATION** in the example above with the name of your web application.
3. Stop the web application, so its settings can be modified.
4. Select **Classes loaded with application class loader first (parent last)** from the combo box.
   Restart the web application.

5. Repeat these steps for all other web applications that use JSF.

Reload pages that display cached values. Cached values are now displayed correctly.

4.1.10.3 To deploy to a WebSphere cluster

To manually deploy SAP BusinessObjects Enterprise web applications to a WebSphere web application cluster distributed over multiple machines, you need to deploy the web applications to the machine hosting the WebSphere Deployment Manager. Once all the required web applications have been installed, you can use the WebSphere Integrated Solutions Console to separately deploy these applications to the cluster.

Tip:
It is recommended that you install and configure a hardware or software load balancer if running BusinessObjects Enterprise web applications on a WebSphere web application cluster. To configure the load balancer and cluster setup, consult your WebSphere documentation.

To deploy your web applications to a WebSphere cluster, use the following general workflow:

1. If BusinessObjects Enterprise is installed to the same machine hosting the WebSphere Deployment Manager, skip to step 3. If SAP BusinessObjects Enterprise and the WebSphere Deployment Manager runs on different machines first copy the WDeploy tool and environment to the machine hosting the WebSphere administration server.
2. Modify the `config.websphere`*X* file located in `<`*BOE_INSTALL_DIR*`>\SAP BusinessObjects Enterprise XI 4.0\wdeploy\conf`.

   You need to provide information for the administration server. For example:

   ```
   as_dir=C:\Program Files\IBM\WebSphere\AppServer
   as_soap_port=8779
   as_virtual_host=default_host
   as_admin_is_secure=false
   enforce_file_limit=true
   ```

3. Open a command-line console and run the following command:

   ```
   wdeploy.bat websphere6 predeployall
   ```

4. Open the WebSphere Integrated Solutions Console.
5. Go to **Applications > Enterprise Applications** to separately install each web application to deploy from the following directory: `<`*BOE_INSTALL_DIR*`>\SAP BusinessObjects Enterprise XI 4.0\wdeploy\workdir\websphere`*X*`\application\`

   Note:
   All web applications are stored as `EAR` modules.

6. Follow the Install new application steps in the console to install the web application

   Note:
   In "Map modules to servers", make sure that you highlight the target cluster and select the module to install before clicking Apply.

7. After installing the application, click **Save**.
8. Repeat steps 5 to 7 for each EAR file to install.
9. Go to **System administration > Nodes** and select the nodes in the cluster that are not indicated as synchronized
10. Click **Synchronize**.
11. Go to **Applications > Enterprise Applications** to start the application just installed.

4.1.11 WebSphere 7.0 administrative console manual deployment

Ensure that the web application server is installed, configured, and running. Use the `wdeploy predeploy` command to create EAR files that can be deployed.

Log in to the WebSphere "Integrated Solutions Console" with the Administrator account. You can run the WebSphere "Administrative console" program, or use a web browser to open `http://`*`WAS_HOSTNAME`*`:`*`PORT`*`/ibm/console` where *`WAS_HOSTNAME`* is the name of your WebSphere server, and *`PORT`* is the port number on which the server listens for login requests. The default port number is 9060.

1. Select **Applications > Application Types > WebSphere enterprise applications** in the menu.
   The "Enterprise Applications" screen appears.
2. Click **Install**.
   The "Preparing for the application installation" screen "Path to the new application" appears.
3. Click **Browse** and choose the web application's EAR file.
   The fully-qualified path to the web application's EAR file appears in the "Full Path" field.
4. Proceed to the next screen.
   The "How do to install the application?" screen appears.
5. Accept the default option of "Fast Path" and proceed to the next screen.
   The "Install New Application" screen "Step 1: Select install options" appears.
6. Accept the default options and proceed to the next screen.
   The "Step 2: Map modules to servers" screen appears.
7. Accept the default options and proceed to the next screen.
   The "Step 3: Map virtual hosts for Web modules" screen appears.
8. Accept the default options and proceed to the next screen.
   The "Step 4: Map context roots for Web modules" screen appears.
9. Enter a context root for the web module selected and proceed to the next screen.
   For example, when deploying `BOE.ear`, the context root could be `/BOE`.
   The "Step 5: Summary" screen appears.
10. Review the summary and click **Finish**.

The web application is installed and a master configuration change confirmation message appears.

11. Click **Save directly to the master configuration**.
    The web application configuration is saved and you are returned to the "Enterprise Applications" screen. Notice that the web application is not yet running.
12. Choose the web application from the list of administered resources.
    The "Enterprise Applications" configuration screen appears.
13. Click **Manage Modules**.
    The "Manage Modules" screen appears.
14. Select the web application module (in the "Module" column).
    The "Manage Modules General Properties" screen appears.
15. Select **Classes loaded with application class loader first (parent last)** from the "Class loader order" property.
    A confirmation message appears.
16. Click **Save directly to the master configuration**.
    The web application configuration is saved and you are returned to the "Manage Modules" screen.
17. Click **OK**.
    A master configuration change confirmation message appears.
18. Click **Save directly to the master configuration**.
    The web application configuration is saved and you are returned to the "Enterprise Applications" screen.
19. Select the web application checkbox and click **Start**.
    A message appears to confirm that the web application started successfully.

The web application has been deployed, configured, and started. Test the web application by using the root context you provided above. Repeat these steps for each web application.

4.1.11.1 To change the WebSphere 7.0 deployment time-out setting

The deployment of SAP BusinessObjects Enterprise web applications to a WebSphere 7 web application server may take some time.

If you receive a time-out message while deploying web applications to a WebSphere 7, increase the deployment time-out setting.

1. Use a text editor to edit the configuration file `deployment.xml`. The location of `deployment.xml` will vary depending on how your server was installed and configured.
   The `deployment.xml` configuration file is typically located in `<DEPLOYMENT_CELL>/applications/isclite.ear/deployments/isclite/deployment.xml`.
   Substitute `<DEPLOYMENT_CELL>` with the name of the deployment server cell name.

For example, if the cell is named `NetworkDeploymentCell01`, the full path to `deployment.xml` could be `/opt/ibm/WebSphere/AppServer/profiles/Dmgr01/config/cells/NetworkDeploymentCell01/applications/isclite.ear/deployments/isclite/deployment.xml`.

2. Set the attribute `invalidationTimeout` to the desired value, in minutes, where the maximum value is `-1` (do not time out).
3. Save `deployment.xml`.
4. Restart the WebSphere service.

The WebSphere time-out setting has been changed.

4.1.11.2 Disable WebSphere JSF implementation

If pages rendered by WebSphere do not show cached values correctly, it may be using its own version of the JavaServer Faces (JSF) library, rather than the JSF implementation supplied with SAP BusinessObjects Enterprise. To correct this problem, manually reconfigure WebSphere for any web application that uses JSF. For example, the BOE web application uses JSF.

To disable the JSF implementation supplied with WebSphere, follow the instructions below.

1. Log in to the "WebSphere Application Server Administration" console using the following URL: `http://`*`WAS_HOSTNAME`*`:`*`PORT`*`/admin`. The WebSphere administrative console's default port number is 9060.
2. Browse to **YOUR_WEB_APPLICATION > Manage Modules > YOUR_WEB_APPLICATION.ear**
   Replace **YOUR_WEB_APPLICATION** in the example above with the name of your web application.
3. Stop the web application, so its settings can be modified.
4. Select **Classes loaded with application class loader first (parent last)** from the combo box.
   Restart the web application.
5. Repeat these steps for all other web applications that use JSF.

Reload pages that display cached values. Cached values are now displayed correctly.

4.1.11.3 To deploy to a WebSphere cluster

To manually deploy SAP BusinessObjects Enterprise web applications to a WebSphere web application cluster distributed over multiple machines, you need to deploy the web applications to the machine hosting the WebSphere Deployment Manager. Once all the required web applications have been installed, you can use the WebSphere Integrated Solutions Console to separately deploy these applications to the cluster.

Tip:
It is recommended that you install and configure a hardware or software load balancer if running BusinessObjects Enterprise web applications on a WebSphere web application cluster. To configure the load balancer and cluster setup, consult your WebSphere documentation.

To deploy your web applications to a WebSphere cluster, use the following general workflow:

1. If BusinessObjects Enterprise is installed to the same machine hosting the WebSphere Deployment Manager, skip to step 3. If SAP BusinessObjects Enterprise and the WebSphere Deployment Manager runs on different machines first copy the WDeploy tool and environment to the machine hosting the WebSphere administration server.
2. Modify the `config.websphere`*X* file located in `<`*BOE_INSTALL_DIR*`>\SAP BusinessObjects Enterprise XI 4.0\wdeploy\conf`.

   You need to provide information for the administration server. For example:

   ```
   as_dir=C:\Program Files\IBM\WebSphere\AppServer
   as_soap_port=8779
   as_virtual_host=default_host
   as_admin_is_secure=false
   enforce_file_limit=true
   ```

3. Open a command-line console and run the following command:

   ```
   wdeploy.bat websphere6 predeployall
   ```

4. Open the WebSphere Integrated Solutions Console.
5. Go to **Applications > Enterprise Applications** to separately install each web application to deploy from the following directory: `<`*BOE_INSTALL_DIR*`>\SAP BusinessObjects Enterprise XI 4.0\wdeploy\workdir\websphere`*X*`\application\`

   Note:
   All web applications are stored as `EAR` modules.

6. Follow the Install new application steps in the console to install the web application

   Note:
   In "Map modules to servers", make sure that you highlight the target cluster and select the module to install before clicking Apply.

7. After installing the application, click **Save**.
8. Repeat steps 5 to 7 for each EAR file to install.
9. Go to **System administration > Nodes** and select the nodes in the cluster that are not indicated as synchronized
10. Click **Synchronize**.
11. Go to **Applications > Enterprise Applications** to start the application just installed.

Known issues and work-arounds

5.1 Windows line endings in wdeploy.sh script

The `wdeploy.sh` script may contain Windows DOS line endings when copied from a Windows machine to a Unix or Linux system.

To resolve this problem, use a text editor or a utility like `dos2unix` to convert `wdeploy.sh` from Windows DOS line endings (carriage return, line break) to Unix line endings (carriage return).

5.2 Renaming BOE web application or web application source tree

When the SAP BusinessObjects Enterprise web application source tree in folder is renamed, it may not be properly deployed by the WDeploy tool. This occurs on all supported platforms. The source tree folder is located in:

`<BOE_INSTALL_DIR>/enterprise_xi40/warfiles/webapps`

To work around this issue, follow these steps:

1. Locate the BOE web application configuration file, `BOE.properties` in

   `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/apps`

2. Rename `BOE.properties` so that it reflects the new name for the web application WAR file.

   For example, if you want to rename the web application from BOE to MYBOE, and you have already renamed `BOE.war` to `MYBOE.war`, rename `BOE.properties` to `MYBOE.properties`.

3. Use a text editor to update the contents of the newly named `.properties` file. Update the web application information in the configuration file.

   ```
   #Business Objects Configuration Utility
   #BOE specific properties
   warfile=<WEB_APP_NAME>.war
   appvdir=<WEB_APP_NAME>
   buildfile=<WEB_APP_NAME>.xml
   osgisupported=true
   ```

   For example, if you want to rename the web application from BOE to MYBOE, replace `<WEB_APP_NAME>` above with `MYBOE`.

4. Rename `BOE.xml` so it reflects the new name for the web application WAR file.

For example, if you want to rename the web application from BOE to MYBOE, and you have already renamed `BOE.war` or `BOE.ear` to `MYBOE.war` or `MYBOE.ear` , rename `BOE.xml` to `MYBOE.xml`.

The WDeploy tool can now deploy the newly-named web application to the web application server.

5.3 Using WDeploy in a pure IPv6 environment

Using the `wdeploy` command in a pure IPv6 environment with the Sun Java Development Kit (JDK), the Sun JDK may not allow the web application server to bind with an IPv6 address, causing `wdeploy` commands to fail.

This is because the Sun JDK defaults to use IPv4 addresses. Change the Sun JDK settings so that Java uses IPv6 addresses by setting the `JAVA_OPTIONS` variable in your environment or start-up script:

```
JAVA_OPTIONS="-Djava.net.preferIPv6Addresses=true"
```

For example, on a WebLogic web application server, add `JAVA_OPTIONS="-Djava.net.preferIPv6Addresses=true"` to the `setDomainEnv.sh` or `setDomainEnv.cmd` scripts.

Restart your web application server, and verify connectivity with the `::1` or `localhost` alias in the `hosts` file.

5.4 Web Services on split web tier servers

Web Services is not supported on split web tier deployments. Split web tier deployments have separate web and web application servers.

5.5 WDeploy GUI translation

When using a non-English language in the WDeploy GUI tool:

- WDeploy does not support non-English characters in the server admin username or password parameters in the `config.<WEB_APP_SERVER>` configuration file.

5.6 Web application not removed from server

If a web application was not removed with the WDeploy undeploy or undeployall commands, use the administrative console to stop all web applications and restart the server, then re-run the `undeploy` or `undeployall` command.

5.7 Launching WDeploy GUI on Red Hat Linux Enterprise servers

Starting WDeploy GUI on Red Hat Linux Enterprise Server 5.1 or Red Hat Linux Advanced Server 5.x with SE Linux enabled may result in the following error message:

```
Can't load 'perl/lib/site_perl/auto/XML/LibXML/Common/Common.so' for module XML::LibXML::Common:
[...]/perl/extlibs/libz.so.1: cannot restore segment prot after reloc
```

To work around this problem, disable SE Linux:

1. Run `/usr/sbin/setenforce 0` with root privileges.

   For example:

   ```
   sudo /usr/bin/setenforce 0
   ```

2. Update the configuration file `/etc/sysconfig/selinux` so that the `SELINUX` parameter is set to `disabled`.

   For example:

   ```
   SELINUX=disabled
   ```

3. Reboot the system.

The WDeploy GUI is now able to start normally.

5.8 Copy MobileOTA14.properties after performing Web Tier installation

If you plan to use SAP BusinessObjects Enterprise Mobile support, and you have performed a Web Tier installation, you must copy the `MobileOTA14.properties` file to the host that received the Web Tier installation (mostly likely, the web application server).

To copy `MobileOTA14.properties` to the host that received the Web Tier installation, use the following steps.

1. Locate `MobileOTA14.properties` on a host that received a "Full" or "Custom / Expand" installation of SAP BusinessObjects Enterprise.

   The file `MobileOTA14.properties` is located in:

   `<BOE_INSTALL_DIR>/enterprise_xi40/wdeploy/conf/apps`

2. Copy `MobileOTA14.properties` to the corresponding path on the host that received the Web Tier installation.

5.9 Cancel button in WDeploy GUI tool

Pressing the WDeploy **Cancel** button during the deployment of web applications to a web application server may not remove web applications or supporting files that have already been copied to the web application server.

To work around this problem, manually undeploy or delete the files from the web application server. With some web application servers, the **Cancel** button may be disabled. This is a known limitation of the JDK being used for deployment.

5.10 Tomcat

5.10.1 Undeploy web applications from Tomcat using WDeploy

The `wdeploy undeploy` command on a Tomcat web application server may encounter an error if the web application server is not shut down and still in use. Work around this issue by shutting down the Tomcat web application server before attempting to run the `wdeploy undeploy` command.

5.11 SAP NetWeaver

5.11.1 SAP NetWeaver AS Java 7.2 web application deployment

When using the WDeploy tool to deploy web applications to SAP NetWeaver AS Java 7.2 SP3 with a MaxDB database, the deployment may fail to complete.

This occurs when there is not enough space in the MaxDB log volume. To add a MaxDB log volume:

1. Install and launch SAP MaxDB Database Studio 7.7.06.09.

2. Navigate to the "Servers" screen for the database instance registered to SAP NetWeaver AS Java 7.2 SP3.
3. Select the database instance and click **Open Administration** on the Explorer tab.
4. Navigate to the "Log Area" tab, located in the right-hand pane.
5. Right-click the "LOG002" log and select **New**.

   After clicking OK, the "LOG002" volume turns blue.
6. Restart the MaxDB database instance.
7. Re-run the WDeploy command to deploy web applications to NetWeaver AS Java 7.2 SP3.

5.11.2 Deployment of dswsbobje web application fails

When deploying the dswsbobje web application on SAP NetWeaver AS Java 7.2, the deployment exception `javax.xml.transform.TransformerFactoryConfigurationError: Provider org.apache.xalan.processor.TransformerFactoryImpl not found` may be thrown, and the deployment may fail.

To work around this issue, restart SAP NetWeaver AS Java 7.2 and re-deploy the dswsbobje web application.

5.11.3 To undeploy web applications from Oracle 10g R3 or SAP NetWeaver 7.2

If you have already undeployed SAP BusinessObjects Enterprise web applications, it is possible that some web application related files or folders remain on the web application server. If this is the case, stop the web application server, manually remove the files or folders, and restart the web application server.

To undeploy web applications from Oracle 10g R3 or SAP NetWeaver web application servers, follow the steps below:

1. Log off and close all the browser sessions for web applications, such as BI launch pad, the Central Management Console (CMC), Web Services, and portals.
2. Restart the web application server.
3. Use the WDeploy command to undeploy the web applications from the web application server.

5.12 WebSphere

5.12.1 Deployment to WebSphere 6.1 or 7 in secured mode

A validation error may be generated when using WDeploy with a WebSphere installation that uses Secure Socket Layer (SSL) encryption. To work around this issue, use the `wsadmin` command to manually accept the SSL certificate before attempting to deploy with WDeploy.

For example, change directories to *<WEBSPHERE_INSTALL_DIR>*`/AppServer/bin` and run the following command:

```
./wsadmin -conntype SOAP -port <SOAP_admin_port> -user <as_admin_username> -password <as_admin_password>
```

Replace *<SOAP_admin_port>* with the port number used by WebSphere (e.g. 8880), *<as_admin_username>* with the username for your administrator account (e.g. `admin`), and *<as_admin_password>* with the password for the account you specified for *<as_admin_username>*.

Press Y to accept the certificate, then proceed with the web application deployment using WDeploy.

5.12.2 Web Services on WebSphere 7.0

After using the WDeploy command to deploy Web Services to WebSphere 7.0, the List of Web Services link is not available in the Web Services interface. To work around this issue follow the steps below:

1. Start the WebSphere 7.0 web application server.
2. Use the WDeploy command to deploy `dswsbobje` and `BusinessProcessBI` web applications.
3. Stop the WebSphere 7.0 web application server.
4. Copy the `.mar` files from `WEB-INF`'s `modules` subfolder to `WEB-INF`'s `lib` subfolder, and change the `.mar` extension to `.jar`.

   To do this, copy the file *<WEBSPHERE7_INSTALL_DIR>*`/profiles/AppSrv01/installedApps/`*<WS7_NODE_NAME>*`/dswsbobje.ear/dswsbobje.war/WEB-INF/modules/addressing-1.3.mar` to a new file as: *<WEBSPHERE7_INSTALL_DIR>*`/profiles/AppSrv01/installedApps/`*<WS7_NODE_NAME>*`/dswsbobje.ear/dswsbobje.war/WEB-INF/lib/addressing-1.3.jar`.

   For example:

   ```
   cd <WEBSPHERE7_INSTALL_DIR>/profiles/AppSrv01/installedApps/<WS7_NODE_NAME>/dswsbobje.ear/dswsbobje.war/WEB-INF/modules
   cp addressing-1.3.mar ../lib/addressing-1.3.jar
   ```

   Note:
   Do not remove or rename `addressing-1.3.mar` in the `modules` folder. Ensure that the file is named `addressing-1.3.jar` in the `lib` folder.

5. Repeat step 4 for each `.mar` file in the `modules` folder.
6. Restart the WebSphere 7.0 web application server.
7. Launch Web Services.

The "List of Web Services" link now displays a list of Web Services.

5.12.3 WASX7017E: Exception deploying in WebSphere

If you receive the following exception while using WebSphere:

```
com.ibm.websphere.management.exception.ConfigServiceException

com.ibm.websphere.management.exception.ConnectorException

org.apache.soap.SOAPException: [SOAPException: faultCode=SOAP-ENV:Client;

msg=Read timed out; targetException=java.net.SocketTimeoutException: Read
timed out]
```

Try raising or removing the SOAP connection timeout set by the property:

```
com.ibm.SOAP.requestTimeout=0
```

In the files:

`<WS_INSTALL_DIR>/profiles/<PROFILE>/properties/soap.client.props` and `<WS_INSTALL_DIR>/profiles/<PROFILE>/temp/soap.client.props`

5.13 WebLogic

5.13.1 WebLogic 9.2 MP2 or 10.x with Sun JDK, IBM JDK, or JRockit on Unix or Linux

A deployment to WebLogic 9.2 MP2 or 10.x system with Sun JDK, IBM JDK, or JRockit may fail with the following error message:`java.util.zip.ZipException: error in opening zip file(too many files open).`

To work around this issue, increase the number of files that can be opened by both WebLogic and the operating system.

For WebLogic, increase the default value of the "ulimit" open file descriptor limit from 1024 to 2000. This can be done by modifying the file <*BEA_INSTALL_DIR*>/weblogic/common/bin/commEnv.sh to look like the example shown below:

```
maxfiles=`ulimit -H -n`
if [ "$?" = "0" -a `expr ${maxfiles} : '[0-9][0-9]*$'` - eq 0]; then
ulimit -n 2048
```

For Unix or Linux, increase the value of the "ulimit" open file descriptor to 4096 at the OS level. This can be done by modifying the file /etc/security/limits.conf to look like the example shown below:

Note:
Root access is required.

```
soft nofile 4096
hard nofile 4096
```

Restart of machine is required in case of Linux after making changes to /etc/security/limits.conf.

For more information, see http://support.bea.com/application_content/product_portlets/support_patterns/wls/TooManyOpenFilesPattern.html#Known_WebLogic_Server_Issues.

5.13.2 Undeploy Web Services from WebLogic

The wdeploy undeploy or wdeploy undeployall commands may not remove the BusinessProcessBI and dswsbobje Web Services web applications from a WebLogic 9.2 MP2, 10.0, or 10.3 web application server.

To work around this problem, stop the BusinessProcessBI and dswsbobje web applications with the WebLogic administrative console and restart the web application server. When the web application server has restarted, use the wdeploy undeploy or wdeploy undeployall commands to remove BusinessProcessBI.war and dswsbobje.war.

5.14 Oracle

5.14.1 To undeploy web applications from Oracle 10g R3 or SAP NetWeaver 7.2

If you have already undeployed SAP BusinessObjects Enterprise web applications, it is possible that some web application related files or folders remain on the web application server. If this is the case, stop the web application server, manually remove the files or folders, and restart the web application server.

To undeploy web applications from Oracle 10g R3 or SAP NetWeaver web application servers, follow the steps below:

1. Log off and close all the browser sessions for web applications, such as BI launch pad, the Central Management Console (CMC), Web Services, and portals.
2. Restart the web application server.
3. Use the WDeploy command to undeploy the web applications from the web application server.

5.14.2 Before the deployment of web applications to Oracle 10g R3

Before deploying web applications to Oracle 10g R3, follow the steps below:

1. Ensure that the web application configuration file `dswsbobje.properties` has been updated.
   The file `dswsbobje.properties` file is located in the folder:

   `<BOE_INSTALL_DIR>/sap_bobj/enterprise_xi40/wdeploy/conf/apps`

   a. Replace all occurrences of `xercesImpl-2.8.1.jar` with `xercesImpl.jar`.
   b. Replace all occurrences of `xml-apis-1.3.03.jar` with `xml-apis.jar`.
2. Save `dswsbobje.properties` and proceed with the deployment.

5.15 Sun Java System Application Server

5.15.1 Instance configuration was modified in Sun Java Web Server

After deploying to a Sun Java System Web server, the following warning may be displayed in the Administration console:

```
Instance configuration was modified
```

To work around this problem, select "**Pull and deploy configuration from <instance name>**" and click **OK**.

5.15.2 Sun Java System Web Server connector information for WebLogic

When using Sun Java System Web Server as a dedicated web server with WebLogic, add the following XML code to the Sun Java System Web Server configuration file `obj.conf`:

```
<Object name="weblo">
   Service fn="wl_proxy" WebLogicHost="<WL_SERVER>" WebLogicPort="<PORT>"
</Object>
```

Note that `weblo` is the name of the connector passed to the WDeploy command with the `connector_name` parameter. Substitute *<WL_SERVER>* with the name of the WebLogic server, and *<PORT>* with the port number used by the WebLogic server to listen for new connections.

5.15.3 Sun Java System Web Server connector information for Tomcat

When using Sun Java System Web Server as a dedicated web server with Tomcat, add the following XML code to the Sun Java System Web Server configuration file `obj.conf`:

```
<Object name="jknsapi">
   ObjectType fn=force-type type=text/plain
   Service fn="jk_service" method="*" worker="worker1"
</Object>
```

Note that `jknsapi` is the name of the connector passed to the WDeploy command with the `connector_name` parameter.

More Information

Information Resource	Location
SAP BusinessObjects product information	http://www.sap.com
SAP Help Portal	Navigate to http://help.sap.com/businessobjects and on the "SAP BusinessObjects Overview" side panel click **All Products**. You can access the most up-to-date documentation covering all SAP BusinessObjects products and their deployment at the SAP Help Portal. You can download PDF versions or installable HTML libraries. Certain guides are stored on the SAP Service Marketplace and are not available from the SAP Help Portal. These guides are listed on the Help Portal accompanied by a link to the SAP Service Marketplace. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.
SAP Service Marketplace	http://service.sap.com/bosap-support > Documentation • Installation guides: https://service.sap.com/bosap-instguides • Release notes: http://service.sap.com/releasenotes The SAP Service Marketplace stores certain installation guides, upgrade and migration guides, deployment guides, release notes and Supported Platforms documents. Customers with a maintenance agreement have an authorized user ID to access this site. Contact your customer support representative to obtain an ID. If you are redirected to the SAP Service Marketplace from the SAP Help Portal, use the menu in the navigation pane on the left to locate the category containing the documentation you want to access.
Docupedia	https://cw.sdn.sap.com/cw/community/docupedia Docupedia provides additional documentation resources, a collaborative authoring environment, and an interactive feedback channel.
Developer resources	https://boc.sdn.sap.com/ https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary

Information Resource	Location
SAP BusinessObjects articles on the SAP Community Network	https://www.sdn.sap.com/irj/boc/businessobjects-articles These articles were formerly known as technical papers.
Notes	https://service.sap.com/notes These notes were formerly known as Knowledge Base articles.
Forums on the SAP Community Network	https://www.sdn.sap.com/irj/scn/forums
Training	http://www.sap.com/services/education From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style.
Online customer support	http://service.sap.com/bosap-support The SAP Support Portal contains information about Customer Support programs and services. It also has links to a wide range of technical information and downloads. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.
Consulting	http://www.sap.com/services/bysubject/businessobjectsconsulting Consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in topics such as relational and multidimensional databases, connectivity, database design tools, and customized embedding technology.

Index

R

S

T

U

V

W