



BEA Workshop for WebLogic Platform™

Release Notes

Version 10.0 Maintenance Pack 1
Revised: October 2007

Contents

What's New in BEA Workshop for WebLogic Platform	1
Built on Eclipse 3.2.	1
Apache Beehive	2
Java Server Faces	2
Standards-based Java Web Services	2
Java 5 Annotations	3
Upgrade Tools.	3
Blended Application Support	3
Considerations for Using Workshop for WebLogic Version 10.0.	3
Location of Supported Platform Information.	4
Documentation Updates.	4
Known Limitations for Workshop for WebLogic Version 10.0	4
Resolved Problems for Workshop for WebLogic Version 10.0.1	25

BEA Workshop Version 10.0 Release Notes

This document contains information on the following subjects:

- [What's New in BEA Workshop for WebLogic Platform](#)
- [Considerations for Using Workshop for WebLogic Version 10.0](#)
- [Location of Supported Platform Information](#)
- [Documentation Updates](#)
- [Known Limitations for Workshop for WebLogic Version 10.0](#)
- [Resolved Problems for Workshop for WebLogic Version 10.0.1](#)

What's New in BEA Workshop for WebLogic Platform

Workshop for WebLogic continues the ground-breaking innovation of the 8.1 release, providing powerful tools for developing WebLogic Platform applications.

Here are some of the most important new features in Workshop for WebLogic:

Built on Eclipse 3.2

Workshop for WebLogic is built on the widely used Eclipse Platform. Instead of the proprietary IDE framework used in previous releases, version 10.0 uses the Web Tools Platform 1.5.

Apache Beehive

Version 10.0 supports Apache Beehive, an open source framework for web applications. Support for Apache Beehive includes:

- **Beehive NetUI:** A web application framework based on Apache Struts. Introduced in version 8.1 as Page Flow technology, Beehive NetUI has evolved into an open-source web application framework. Version 10.0 includes advanced tooling features to help developers build Beehive NetUI applications, including graphical tools for JSP and Page Flow controller authoring.
- **Beehive Controls:** A lightweight component framework based upon annotated JavaBeans. Beehive controls provide an easy to use client model for accessing a variety of J2EE resource types. The built-in controls introduced in version 8.1 (database, EJB, JMS and web service controls) are available as system controls in version 10.0. You can also create your own custom controls using a new, transparent, POJO architecture.

Java Server Faces

Version 10.0 includes first-class support for Java Server Faces technology, including:

- Support for JSF pages in NetUI Beehive applications
- Tools support for JSF backing files and event handlers
- Integration of JSF with the Beehive NetUI life cycle

Standards-based Java Web Services

The same ground-breaking web service support in version 8.1 has been carried forward in version 10.0, now built on JSR-181. Web service support includes:

- Asynchronous callbacks
- State management with conversations
- Ease of iterative development with the test browser has been carried forward from version 8.1
- Reliable messaging with JMS and WS-Reliability
- Digital Signature and HTTPS support with WS-Security/WS-Policy

Java 5 Annotations

As with version 8.1, version 10.0 supports the use of annotations to simplify the development of complex components. While most of the functionality of the version 8.1 annotations carries forward into this version, the version 10.0 annotations are based on the JSR-175 standard, new in Java 5. Workshop for WebLogic continues to provide tool support to keep the use of annotations simple, including a property editor for intuitive annotation editing.

Upgrade Tools

Version 10.0 makes it easy to upgrade your version 8.1 applications. Upgrade support includes:

- Pre-upgrade reporting that doesn't affect the original source code
- Automatic handling of application and project structure upgrade
- Automatic file extension renaming
- Wizard and command line versions of the upgrade tool are available

Blended Application Support

Workshop for WebLogic supports blended application architecture combining the best of open source and BEA's innovative technologies. Blended application offers:

- Long-term, license-independent support for applications
- Best of breed hybridization
- Transparency and predictability

Considerations for Using Workshop for WebLogic Version 10.0

Workshop for WebLogic is targeted toward the iterative development experience rather than production deployment. As such, a number of features that work correctly in a standalone (development) server environment will not function as expected in a clustered deployment.

Important — Development and testing of applications using this release of Workshop for WebLogic should be done using standalone server environments.

Location of Supported Platform Information

For more information on platform support, including hardware and software requirements, see the [Supported Platforms](#) web site.

Documentation Updates

Updated documentation is available at the [Workshop for WebLogic e-docs site](#).

Known Limitations for Workshop for WebLogic Version 10.0

[Table 1](#) lists the known limitations found in Workshop for WebLogic.

Table 1 Known Limitations in BEA Workshop Version 10.0

Problem ID	Description
CR242934	On upgrade failure, files are not reverted to their original state When an upgrade fails the new files are not reverted to their original state but may not be fully upgraded. For example, the file extension may have been changed to .java, but annotations may not have been translated to Workshop for WebLogic version 9.2.1 style. Note that the original source files are not altered. Platform: All Workaround: Either drag and drop the version 8.1 file into the workspace and perform upgrade on that file again or re-import the entire project.
CR249614	Web service with modified parameter names requires @WebParam annotation post-upgrade A start-from-WSDL web service where the parameter names (in the Java signature) do not match the WSDL name require a @WebParam annotation be manually added post-upgrade. The most likely case where a parameter name is modified is when the WSDL name is an invalid Java identifier. Platform: All Workaround: Add a @WebParam annotation with the matching WSDL name to each parameter in the web service operation.

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR264808	<p>Error is displayed in the New Server wizard even if an 8.1 domain is successfully upgraded</p> <p>If a user selects a version 8.1 domain in the New Server wizard, a hyperlink is displayed to allow user to launch the Domain Upgrader. Upon successfully upgrading the domain, the state of wizard is not refreshed under some circumstances. The old error message on the top of the wizard and the upgrade hyperlink remain, and the Next button is not enabled.</p> <p>Platform: All</p> <p>Workaround: The problem can be worked around by re-selecting the text in the domain home combo box.</p>
CR265853	<p>Eclipse hangs if you put a breakpoint in an interface</p> <p>The Eclipse debugger allows the user to place a breakpoint on an interface declaration. Sometimes, when this breakpoint fires, Eclipse will hang. This is a known problem in Eclipse 3.1, which Workshop for WebLogic version 9.2.1 is based upon.</p> <p>Platform: All</p> <p>Workaround: Place breakpoints only on, or in, actual method implementations, never on a method declaration found in an interface file.</p>
CR267837	<p>Javadoc attachments for Workshop for WebLogic libraries can be broken if Workshop for WebLogic is not installed into the default directory under <BEA-HOME></p> <p>During installation there is an option to specify the directory into which Workshop for WebLogic version 9.2.1 is installed. If you specify something other than the default (which is “workshop92”), then the Javadoc for Workshop for WebLogic code such as the service control or timer control cannot be found by the IDE. Thus, actions like Shift-F2 to show the documentation for the classes will not work</p> <p>Platform: All</p> <p>Workaround: Add Javadoc attachments to the jars contained in those libraries manually. For example, to add a javadoc attachment to the base Workshop for WebLogic controls jar, go to Windows > Preferences > WebLogic > J2EE Libraries. Select weblogic-controls-1.0 and click Edit. Expand each jar in the Classpath Contribution: tree, right-click the Javadoc location node, and select Edit. Next set the Javadoc location path: to <BEA-HOME>/<your-workshop-dir>/workshop4WP/docs/api, where <your-workshop-dir> is the non-default workshop directory name you entered during install.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR267912	<p data-bbox="243 383 705 418">Source Not Found when debugging JDK classes</p> <p data-bbox="243 423 1180 487">In some cases while debugging an application the source for the JDK classes cannot be found, and will result in a “Source Not Found” page for the class.</p> <p data-bbox="243 493 387 527">Platform: All</p> <p data-bbox="243 532 1180 597">Workaround: The workaround is to add a Source attachment manually. This can be done either at the workspace preference level, or if already running a debug session it can be done right there.</p> <p data-bbox="243 602 1180 753">If not yet in a debug session, go to the Windows > Preferences > Java > Installed JREs preference page. Select the jdk150_04 JRE and click Edit. Deselect “Use default system libraries” on the Edit JRE dialog. Open the node for rt.jar and select the Source attachment node. Click Edit, then select External File... and navigate to <BEA-HOME>/jrocket90_150_04/ and select src.zip.</p> <p data-bbox="243 758 1180 857">If you are already in a debug session and you get an editor page indicating “Source Not Found” for a JDK class, select the “Edit Source Lookup Path...” button. Click Add, then select “External Archive”. In the file dialog navigate to <BEA-HOME>/jrocket90_150_04/ and select src.zip.</p>
CR269490	<p data-bbox="243 874 790 909">Threads display “(may be out of sync)” in the debugger</p> <p data-bbox="243 914 1180 1048">When debugging an application on the server, occasionally when saving code changes, you may see the following displayed in the debugger window on one or more of the threads: “(may be out of sync)”. If this occurs, you will need to republish your application in order to resync the server with the code you have just entered. This is a known problem with Eclipse 3.1/3.2.</p> <p data-bbox="243 1053 387 1088">Platform: All</p> <p data-bbox="243 1093 776 1128">Workaround: Republish the application to the server.</p>
CR271247	<p data-bbox="243 1135 727 1170">Unexpected server error may require IDE restart</p> <p data-bbox="243 1175 1180 1274">Some uncontrolled server errors or terminations may cause publish status and publish operations errors. For example, if an Out Of Memory condition occurs in the server process, both Workshop for WebLogic and the server may require a restart.</p> <p data-bbox="243 1279 387 1314">Platform: All</p> <p data-bbox="243 1319 928 1354">Workaround: Restart Workshop for WebLogic and restart the server.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR272082 CR273414 CR272245	<p>JSP tag variables cannot be resolved in the JSP editor</p> <p>Due to a problem in Eclipse, some JSP tags (such as <auth:login> and <portlet:actionURL>) and variables declared from JSP tags are marked as containing an error when they are actually correct; although no error actually exists, Eclipse will not publish (deploy) the application.</p> <p>Platform: All</p> <p>Workaround: If this situation occurs, you must turn off JSP validation before publishing. Leave JSP validation on until you have fixed any problems except those caused by these tags; before deploying, select Window > Preferences, select Validation in the tree, and uncheck the JSP Syntax Validator check box.</p>
CR277475	<p>The console used for server startup on Linux may not display multi-byte characters correctly</p> <p>Because the IDE uses xterm to start the server, multi-byte characters may not display correctly in the console.</p> <p>Platform: Non-English Linux Operating Systems</p> <p>Workaround: Start the server from the command line in a console that is correctly configured for the language in use, outside of the IDE. After creating the server definition in the IDE, it will be able to connect to the already running server.</p>
CR280928	<p>Tuxedo Control is no longer available with Workshop for WebLogic version 9.2.1</p> <p>Tuxedo Control will no longer be available with Workshop for WebLogic.</p> <p>Platform: All</p> <p>Workaround: Refer to the Tuxedo Control Migration white paper provided in PDF format. The PDF provides migration options from the Tuxedo Control to other alternatives.</p>
CR282777	<p>The first time you run Project > Clean after importing a project, Workshop for WebLogic may not clean all of the files from the .apt_src directory</p> <p>When a user imports a project including build directories such as .apt_src, all files in that directory might not get removed the first time a clean is performed. This only happens the first time, subsequent cleans correctly handle the generated directories.</p> <p>Platform: All</p> <p>Workaround: Manually delete the files in the .apt_src directory when performing a clean the first time after import. Once deleted, the new files that go into that directory are correctly handled by the clean action.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR283022	<p data-bbox="243 383 1180 423">Xbean wrapper classes generated for a ServiceControl must not be visible to the target JWS.</p> <p data-bbox="243 430 1180 578">When generating Xbean types for a Service Control, there are a limited number of situations where the wrapper type from the WSDL is exposed as a type in the Service Control (e.g. when there are multiple occurrences of the same Document type in the same operation signature). If the generated types jar is visible to the target JWS classloader, a duplicate type error will be thrown during deployment.</p> <p data-bbox="243 585 1180 614">Platform: All</p> <p data-bbox="243 626 1180 725">Workaround: Place Service Controls that use wrapper types in a separate project than the Service that they call. If the ServiceControl classes are loaded from a utility project, then the Service Control and the target Service must be in separate applications.</p>
CR283457	<p data-bbox="243 736 1180 775">XBeans are not supported as parameters or return type for doc/lit/bare operations.</p> <p data-bbox="243 782 1180 843">Use of Xbeans as a parameter or return type with doc/lit/bare bindings is not supported in operations or callbacks and will result in a failure during deployment.</p> <p data-bbox="243 850 1180 880">Platform: All</p> <p data-bbox="243 892 1180 935">Workaround: Use doc/lit/wrapped for services that use XBeans as parameters or return types.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR283533	<p>Not all Beehive APIs shipping with Workshop for WebLogic are final</p> <p>Workshop for WebLogic ships an SVN snapshot of Apache Beehive (post v1.0.2). There are certain APIs that were introduced into Beehive at Apache, post v1.0.2 but have not officially shipped with a release from Apache.</p> <p>The following APIs should not be considered frozen.</p> <ul style="list-style-type: none"> • Support for partial data sets in netui data grid SVN 431515 implements supports for partial data sets and adds a "partialDataSet" flag to the configurePager tag. This feature was not included in the Apache Beehive release v1.0.2, but was included in Workshop for WebLogic's post v1.0.2 shapshot release of Beehive. • Plug points into low level features to handle AJAX requests (.xhr) for the Tree and DivPanel in NetUI... In the 1.0 release of Beehive, the NetUI Tree and DivPanel are both AJAX enabled. There are some new plug points to allow more advanced handling and routing of the requests. The API changes include: <ul style="list-style-type: none"> – The URLRewriter getAjaxUrl() method can be used in URL rewriting to provide information for an AJAX request. – Chain of Responsibility (CoR) / command pattern for servicing AJAX requests. This is a new command handling infrastructure that can be used to handle requests to render markup or data to a client. At a high level, this is done by implementing command handler classes that are composed together using the Chain of Responsibility (CoR) pattern and including the commands in the NetUI configuration. <p>Platform: All</p> <p>Workaround: None</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR285560	<p data-bbox="244 385 1176 420">Upgrade does not copy some MBCS characters in .jcx files</p> <p data-bbox="244 427 1176 552">When upgrading some .jcx files, the upgrade process may not be able to copy some MBCS characters into the resulting upgraded file. This happens when a WSDL with UTF-8 encoding is inside a file with some other sort of MBCS encoding, like MS932 (Japanese) and MBCS characters appear in that file outside of the WSDL definition.</p> <p data-bbox="244 558 1176 621">When this happens, the resulting file will have "?????" in place of the original characters, which will not compile.</p> <p data-bbox="244 628 1176 663">Platforms: All</p> <p data-bbox="244 670 1176 704">Workaround:</p> <ol data-bbox="244 711 1176 864" style="list-style-type: none"> 1. Change the wsdl definition to VM encoding instead of UTF-8 or others, if these are different. 2. Upgrade source 3. Change the wsdl definition to original like UTF-8 both within Service Control source comment and generated wsdl file.
CR286141	<p data-bbox="244 871 1176 906">WebLogic EJB project properties not visible from generated Ant build scripts</p> <p data-bbox="244 913 1176 1010">Description: The "Jar settings" properties and EJBC flags that can be set on WebLogic EJB projects via Project->Properties->WebLogic EJB are only used when the IDE build executes; these settings are not visible when exported Workshop Ant build scripts are executed.</p> <p data-bbox="244 1017 1176 1052">Platforms: All</p> <p data-bbox="244 1058 1176 1159">Workaround: Prior to building the project with the exported Ant script, specify all desired "Jar settings" properties using weblogic.ejbcgen.JarSettings annotations in your EJB Java source files and add and desired EJBC flags directly to the build script where "weblogic.ejbc" is executed.</p>
CR287585	<p data-bbox="244 1166 1176 1229">When a web service or other artifact is in a web project that is part of more than one EAR, the Run on Server can produce the wrong URL.</p> <p data-bbox="244 1236 1176 1333">Using the Run on Server action on artifacts in a web project that is associated with multiple EARs in the workspace can launch the browser to the incorrect URL. That is, it will use the URL associated with the last EAR with which is was associated.</p> <p data-bbox="244 1340 1176 1374">Platforms: All</p> <p data-bbox="244 1381 1176 1416">Workaround:</p> <p data-bbox="244 1423 1176 1440">To workaround this issue, you can either</p> <ol data-bbox="244 1447 1176 1550" style="list-style-type: none"> a) close the EAR that is not to be used for the launch, b) remove the web project from the EAR that is not used for the launch, or c) manually update the URL in the browser to hit the correct EAR.

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR293197	<p>Lost JVMTI events (especially breakpoints) when debugging with JRockit</p> <p>When debugging with the JRockit JVM you may experience performance problems and missed breakpoints.</p> <p>Platforms: All</p> <p>Workaround: Update to JRockit 5.0 R27.1 or later.</p>
CR294199	<p>Invocation of buffered Control methods will fail at runtime if deployed (via the IDE) to a server with more than one JMS Server</p> <p>The IDE will auto-deploy required Workshop libraries when deploying an application. The use of @MessageBuffer on Control methods creates a dependency on application-scoped JMS resources in the weblogic-controls library. If the weblogic-controls library is deployed (by the IDE) to a server with more than one JMS server, the library will deploy, but the application-scoped JMS resources will not be available. This is because the IDE depends on default sub-module targeting, and default sub-module targeting relies on the target containing exactly one JMS Server. A message similar to the following warns that there is an issue with the deployment:</p> <pre><The JMS module named "WlwRuntimeAppScopedJMS" inside application "testLibWebApp" does not have a sub-deployment stanza named "WlwRuntimeAppScopedJMS". Without such a stanza no entities inside the module will be deployed, since the sub deployments inside of the sub-deployment stanza named "WlwRuntimeAppScopedJMS" control where JMS entities inside this module are targeted.></pre> <p>Note that even though there is a warning message, the library is deployed to the server. This means that applications that are dependent on the library will also successfully deploy. However, invocation of buffered Control methods will fail at runtime with a message similar to the following:</p> <pre>"Failed to invoke end componentFailed to invoke methodMessage buffering is not available - either the buffering MDB did not deploy or we are in a standalone WAR"</pre> <p>Note that this situation is most likely to occur when using domains that were not created with support for "Workshop for Weblogic Platform.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
	<p>Platforms: All</p> <p>Workaround: Manually deploy the library using the weblogic.Deployer command. The form of the command is: <pre>java weblogic.Deployer -username weblogic -password weblogic -adminurl t3://localhost:7001 -deploy -name weblogic-controls-10.0 -source %WL_HOME%/common/deployable-libraries/weblogic-controls-10.0.ear -targets cgServer -submoduletargets cgJMSServer@WlwRuntimeAppScopedJMS@WseeJmsServer -library -libspectver 10.0 -libimplver 10.0</pre> </p> <p>Where: - WseeJmsServer is the name of the JMS Server to host the application-scoped destinations. - targets is the development server</p> <p>The command updates the library definition in config.xml. Therefore it only needs to be run once.</p>
<p>CR295684</p>	<p>Clicking cancel during 8.1 Upgrade may cause exceptions.</p> <p>During upgrade of an 8.1 application, clicking cancel in the upgrade preview window may result in an exception.</p> <p>Platforms: All</p> <p>Workaround: These exceptions may be safely ignored.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR298397	<p>Controls Message Buffering in a cluster requires submodule targeting.</p> <p>If your application enables message buffering on controls using the annotation <code>com.bea.control.annotations.MessageBuffer</code>, you may have the following error when you try to deploy it on a cluster: "While attempting to create destination <code>MSG_BUFFER_QUEUE</code> in module <code><name_of_your_ear>!WlwRuntimeAppScopedJMS</code> the <code>JMSServer</code> of name <code><name_of_your_cluster></code> could not be found".</p> <p>Platforms: All</p> <p>Workaround: In this case, you must target the <code>MSG_BUFFER_QUEUE</code> to a specific JMS Server with the following <code>submoduletargets</code> option to your <code>weblogic.Deployer</code> command:</p> <pre>-submoduletargets cgJMSServer@WlwRuntimeAppScopedJMS@WseeJmsServer_auto_1</pre>
CR299397	<p>Migration of a web project with XDoclet facet enabled results in error</p> <p>Migration of a Workshop for WebLogic 9.2 application that has the XDoclet facet enabled to version 10.0 results in the following error message.</p> <pre>"The preferences for the xdoclet runtime does not point to a valid installation"</pre> <p>Platforms: 10.0</p> <p>Workaround: There is no workaround. XDoclet it not supported in version 10.0.</p>
CR299731	<p>Command line upgrade can occasionally generate a spurious non-fatal error.</p> <p>Occasionally when performing a command line upgrade of an 8.1 application, you may see a non-fatal <code>org.xml.sax.SAXParseException</code>.</p> <p>Platforms: All</p> <p>Workaround: This is a spurious error and can be safely ignored. Upgrade from within the IDE does not exhibit this problem.</p>
CR29999	<p>The Perforce Eclipse plugin (P4WSAD) does not add new .prefs files after upgrading an application from version 9.2 to version 10.0</p> <p>Description: When you upgrade a Workshop for WebLogic version 9.2 application to version 10.0 the Perforce Pending Changelists view does not show all of the files created in the upgrade process. For example, the <code>com.bea.workshop.wls.core.prefs</code> which replaces the <code>com.bea.wlw.libmodules.core.prefs</code> file is not added to the view.</p> <p>Platforms: 10.0</p> <p>Description: Add the files manually by selecting Team > Open for Add at the project level. The new <code>.prefs</code> file and any other new files added to the project will be added to the view.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR300529	<p data-bbox="244 385 1174 437">"exact-match" element value in weblogic.xml and weblogic-application.xml files are ignored during project migration</p> <p data-bbox="244 461 1174 576">Description: When you upgrade a Workshop for WebLogic version 9.2 application to version 10.0, the <exact-match> element is ignored with respect to library modules. Even if <exact-match> is set to true, the upgrade process will update the library modules to the latest versions available in 10.0.</p> <p data-bbox="244 590 411 618">Platforms: 10.0</p> <p data-bbox="244 621 1174 680">Description: Use Workshop 10.1 to upgrade the 9.2 application. Workshop 10.1 support older library versions.</p>
CR301661	<p data-bbox="244 704 1174 756">Buffered methods on ServiceControl may fail over JMS protocol</p> <p data-bbox="244 760 1174 836">Buffered operations on a ServiceControl must be void. However the Message Exchange Pattern (MEP) in the underlying WSDL can be either request/response (with an empty response), or oneway (a request with no response).</p> <p data-bbox="244 850 1174 927">In the case of a request/response MEP over JMS, the presence of @MessageBuffer will cause the request to deadlock and eventually timeout. The following warning message will generally be produced:</p> <pre data-bbox="287 954 1147 1086"> Potential blocking operation {http://someNamespace}someOperation: a synchronous request/response invocation within a transaction using the JMS transport can cause deadlocks. Please refer to WebLogic documentation for details. </pre> <p data-bbox="244 1100 784 1128">The resulting error message will include text similar to:</p> <pre data-bbox="287 1145 1161 1197"> javax.xml.rpc.soap.SOAPFaultException: Failed to receive message java.io.IOException: Request timed out </pre> <p data-bbox="244 1211 956 1239">Note: this only occurs when the transport protocol for the request is JMS.</p> <p data-bbox="244 1253 397 1281">Platforms: All</p> <p data-bbox="244 1295 1174 1409">Workaround: If you can influence the design of the target JWS, having the JWS operation annotated with @Oneway will direct that the underlying MEP be oneway, and will avoid this situation. If you can not influence the design of the target JWS, then the workaround is to add the TransactionAttribute annotation to the ServiceControl operation:</p> <pre data-bbox="287 1426 1134 1503"> @MessageBuffer @TransactionAttribute(TransactionAttributeType.NOT_SUPPORTED) public void voidMethod(); </pre> <p data-bbox="244 1517 1174 1576">Note that the presence of the @TransactionAttribute will not change the transactional behavior of actions that occur within the calling application.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR301304	<p>When upgrading web applications to version 10.0, the Page Flow Controllers source folder may revert to the default location in some cases</p> <p>Certain 9.2 web projects with the Beehive NetUI facet may not retain the value of the Associated Files Model settings. Specifically, if such a project has more than one source folder, and the Page Flow Controllers property panel has been set to a source folder other than the default, this value will revert to the default when the project is upgraded to 10.0.</p> <p>In a project that has both the Beehive NetUI facet and the JSF facet, the same situation may occur for the Project > Properties > Associated Files Model > JSF Backing Files property.</p> <p>Platforms: All</p> <p>Workaround: To change the value in 10.0, select Project > Properties > Associated Files Model > Page Flow Controllers and select the source folder of interest.</p>
CR302849	<p>Leaving the "upgrade projects" wizard open for a long time before canceling or finishing can leave the Workshop IDE in a bad state.</p> <p>Leaving the "upgrade projects" wizard open for a long period of time before canceling or finishing, can result in an error in the Workshop IDE. This will leave Workshop in a bad state and can cause further problems downstream.</p> <p>Platforms: All</p> <p>Workaround: Don't leave the "upgrade projects" wizard dialog open for long periods of time. Or, if you do get an error, restart the IDE.</p>
CR303707	<p>Errors can result when importing Workshop 9.2 projects into Workshop 10.0 using Perforce plugin.</p> <p>When Workshop 9.2 projects are imported into Workshop 10.0 using the Perforce plugin (P4WSAD) you may encounter errors or you may be prompted to close the IDE. These errors and prompts arise because the automatic build cycle encounters inconsistent data that arises during the upgrade to 10.0.</p> <p>Platforms: 10.0</p> <p>Workaround: Sync the projects from Perforce using an external client such as p4win. Import the projects into the workspace by selecting File > Import > General > Existing Projects into Workspace.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
<p>CR303710</p>	<p>Multiple workspace builds can be triggered upon cleaning an application migrated from Workshop for WebLogic 9.2</p> <p>In certain cases an application migrated from Workshop for WebLogic 9.2 will repeatedly build.</p> <p>Platforms: 10.0</p> <p>Workaround: Disable auto build (Project -> Build Automatically) temporarily until the workspace build completes. Then re-enable auto build.</p>
<p>CR304008</p>	<p>Servlet 2.5 Implementation In WebLogic Server 10.0 Can Break 9.2 Beehive Applications</p> <p>Description: An issue may surface when a 9.2 Beehive application is deployed on WebLogic Server 10.0. The symptom with this scenario is a java.lang.IllegalStateException being thrown. The underlying issue is with the javax.servlet.ServletException which changed from Servlet 2.4 to Servlet 2.5.</p> <p>Platforms: WebLogic Server 10.0 or higher</p> <p>Workaround: Upgrade the application to use Beehive 10.0 libraries (which work with either Servlet 2.4 or Servlet 2.5). If the 9.2 application cannot be compiled in 10.0, then manually updating the deployment descriptors in the 9.2 application EAR to use the 10.0 Beehive libraries will resolve this issue. For example:</p> <p>For each .war in the 9.2 .ear, modify the following section of the web-inf/weblogic.xml:</p> <pre data-bbox="243 1003 907 1147"> <wls:library-ref> <wls:library-name>beehive-netui-1.0</wls:library-name> <wls:specification-version>1.0</wls:specification-version> <wls:implementation-version>1.0</wls:implementation-version> </wls:library-ref> </pre> <p>to look like:</p> <pre data-bbox="243 1203 940 1347"> <wls:library-ref> <wls:library-name>beehive-netui-1.0.1-10.0</wls:library-name> <wls:specification-version>1.0</wls:specification-version> <wls:implementation-version>1.0.1.1</wls:implementation-version> </wls:library-ref> </pre>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR322374	<p>Incorrect documentation for for running help in stand alone mode</p> <p>Description: The Workshop for WebLogic Platform version 10.0 documentation gives an incorrect command for starting help in stand alone mode.</p> <p>Platforms: Workshop for WebLogic Platform 10.0</p> <p>Workaround: Run the following command to start help in standalone mode.</p> <pre>%BEA_HOME%/jdk150_06/bin/java -classpath %BEA_HOME%/tools/eclipse32/eclipse/plugins/org.eclipse.help.base_3.2.1.R321_v2006082 2.jar org.eclipse.help.standalone.Infocenter -command start -eclipsehome %BEA_HOME%/tools/eclipse32/eclipse -port 7034 -noexec -product com.bea.workshop.product.wl.workshop</pre>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR304502	<p data-bbox="243 383 1176 453">User's who upgrade an 8.x application to a 9.x/10.x application may experience issues when making multiple method calls to a JDBC control from a single page flow method</p> <p data-bbox="243 458 1176 557">Due to a change in transaction scope from page flows which has been documented in http://e-docs.bea.com/workshop/docs92/ws_platform/upgrading/conChangesDuringUpgrade.html</p> <p data-bbox="243 562 1176 597">See the section labeled: 'Controls are Not Automatically Run Within the Scope of a Transaction'</p> <p data-bbox="243 602 1176 753">User's who upgrade an 8.x application to a 9.x/10.x application may experience issues when making multiple method calls to a JDBC control from a single page flow method. The crux of the issue is that when the first call is made to the JDBC control a new transaction is created by our transaction interceptor. When that call returns the transaction is either committed or rolled back.</p> <p data-bbox="243 758 1176 857">On the next call to the JDBC control a new transaction is created, but the JDBC connection being used by the control cannot be used in another transaction (it has been associated as a resource of the first transaction).</p> <p data-bbox="243 862 1176 1013">The behavior in 8.x page flows was for the JDBC control to release its JDBC connection after each method invocation. The transaction scope for a control method being invoked from a page flow was to start a transaction at the beginning of the control method invocation, and end the transaction on the return of the method. If the control rolled back the transaction, all operations performed within that transaction would be rolled back as well.</p> <p data-bbox="243 1019 1176 1053">Platforms: All</p> <p data-bbox="243 1058 1176 1093">Workaround: There are several workarounds available:</p> <ol data-bbox="243 1098 1176 1275" style="list-style-type: none"> <li data-bbox="243 1098 1176 1180">1) If you do not want to use transactions (they were implicit in 8.x) the transaction interceptor annotations (inserted by the upgrader) can be removed from the control methods. <li data-bbox="243 1185 1176 1275">2) If you want to use transactions, create a JTA transaction in the page flow method and either commit or rollback once the calls to the JDBC control are completed.
CR306317	<p data-bbox="243 1281 1176 1315">Need to undeploy applications before upgrading 9.2 domains</p> <p data-bbox="243 1321 1176 1420">All applications must be undeployed before upgrading 9.2 domains in order for the migrated applications to be successfully redeployed. You may use the WebLogic console to list and remove existing applications.</p> <p data-bbox="243 1425 1176 1459">Platforms: All</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR306339	<p>Workshop for WebLogic and Workshop Studio must not be installed into the same Eclipse installation</p> <p>Users must not install both Workshop for WebLogic and Workshop Studio as plugins to the same Eclipse installation. Multiple errors and failures will occur.</p> <p>Platforms: All</p> <p>Workaround: Install Workshop for WebLogic and Workshop Studio into different Eclipse installations.</p>
CR305233	<p>Project references that are part of a web app but not part of an ear may result in NoClassDefFound errors</p> <p>If a web project in an ear references a java project via the J2EE Modules Dependencies -> Web Libraries settings *and* if the java project is not part of the ear, then the classes are not loaded and you see NoClassDefFound errors. This only occurs during iterative development. Export ear will correctly copy the jar to the web-inf/lib directory of the web app in the ear.</p> <p>Platforms: All</p> <p>Workaround: Use a utility project, add it to the ear, and in the properties for the web app use the 2EE Modules Dependencies -> J2EE Modules settings to reference the java project. The classes will then be found both during iterative dev and in the exported ear.</p>
CR307209	<p>Types not generated when a service control is generated from a RPC/Enc WSDL</p> <p>When a service control is generated from a RPC/Enc WSDL with complex types the service control generation wizard does not generate a types jar.</p> <p>Platforms: BEA Workshop for WebLogic Platform 10.0</p> <p>Workaround: Generate a types jar using the types generation wizard. Right-click the WSDL and select Web Services > Generate Types JAR File.</p>
CR307678	<p>Problems tab does not display correctly after upgrading a 9.2 workspace</p> <p>After opening a 9.2 workspace in Workshop 10.0, the columns of the Problems view are displayed incorrectly.</p> <p>Platforms: All</p> <p>Workaround: Close and reopen the Problems view. To reopen, select Window > Show View > Problems.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR308313	<p>Java Web Services (JWS) Do Not Allow Nested Types</p> <p>The Web Service stack in WebLogic Server 9.x/10.0 does not support nested types as parameters and/or return values. However, Beehive Controls often use inner classes as the pattern for data values (e.g., The JDBC Control contains a commented out example of an inner class to hold the return values from the database queries). Therefore the use of Beehive Control nested values in a JWS is not supported</p> <p>Platforms: 9.x and 10.0</p> <p>Workaround: Any Beehive Control containing inner classes that will be used by a JWS will need to convert the inner class to a standalone class.</p>
CR308749	<p>Duplicate simple class names are not supported for web service controls with callbacks</p> <p>When mutiple web service controls, with callbacks, have identical class names (ignoring package name) an error will occur in jwsc. This error will appear in the publish step in the ide, during the asseble step in exported ant scripts, or when exporting an ear file from the ide. In previous versions of Workshop the exported ant scripts would incorrectly report that the assemble step had succeeded even though this conditon was present. This was because the ant script did not attempt to run jwsc on generated java files.</p> <p>Platforms: All</p> <p>Workaround: When using web service controls (ServiceControl) with callbacks make sure that each control file has a unique un-qualified class name. Differing the package name is not sufficient.</p>
CR311168	<p>Workshop for WebLogic hangs after upgrading 8.x Portal application</p> <p>In some cases Workshop can hang after upgrading a 8.x Portal application.</p> <p>Platforms: 10.0</p> <p>Workaround: Disable the HTML Syntax Validator and attempt upgrade again. To disable the validator select Window > Preferences > Validation. Uncheck the boxes for HTML Syntax Validator.</p>
CR313196	<p>Upgrading a Service Control from 8.1 to 10.0 with unused callbacks may generate compilation error</p> <p>In a Workshop 8.1 application, if a WSDL file specifies a callback that is not used in the web service file, upon upgrade that callback will be generated into the upgraded 10.0 web service file. If the callback uses external types a compile error may be generated.</p> <p>Platforms: 10.0</p> <p>Workaround: Comment out the code inside Service Control's call back interface or completely remove the callback.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR313306	<p>Service control generation can produce incorrect callback method names</p> <p>Service control generation based on WSDL callbacks named with underscores and numbers can result in incorrect callback method names. This will occur if a lower case letter follows an underscore or number.</p> <p>For example, the following WSDL callback names:</p> <pre>a_b a4b</pre> <p>will result in the generation of the following callback method names:</p> <pre>a_B a4B</pre> <p>Note that the first character after a number or underscore has been capitalized.</p> <p>When the method is invoked, the following error will result.</p> <pre>javax.xml.rpc.JAXRPCException: SOAPFaultException - FaultCode [http://schemas.xmlsoap.org/soap/envelope/}Client] FaultString [Failed to get operation name from the incoming request]</pre> <p>Platforms: 10.0</p> <p>Workaround: Avoid callback method names with a lower case letter following an underscore or number.</p>
CR322374	<p>Documentation gives incorrect command from running help in stand alone mode</p> <p>The documentation for Workshop for WebLogic Platform 10.0 gives an incorrect command from running help in stand alone mode.</p> <p>Platforms: Workshop for WebLogic Platform 10.0</p> <p>Workaround: Use the following command to run help in stand alone mode:</p> <pre>%BEA_HOME%/jdk150_06/bin/java -classpath %BEA_HOME%/tools/eclipse32/eclipse/plugins/org.eclipse.help.base_3.2.1.R321_v2006082 2.jar org.eclipse.help.standalone.Infocenter -command start -eclipsehome %BEA_HOME%/tools/eclipse32/eclipse -port 7034 -noexec -product com.bea.workshop.product.wl.workshop</pre>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR325304	<p data-bbox="243 383 1176 453">Auto deployment of new library modules to a 10.0 domain could lead to a class not found during production deployment</p> <p data-bbox="243 458 1176 579">In Workshop 10.1, when a user creates a new target runtime which points to a Workshop for WebLogic 10.0 installation, and develops an application against a domain from that installation, Workshop 10.1 will automatically publish newer versions of library jars for all facets in the application, if newer versions exist.</p> <p data-bbox="243 585 1176 654">However when the user deploys the application to the 10.0 domain, the domain may not have access to the newer libraries.</p> <p data-bbox="243 659 1176 685">Platforms: Workshop for WebLogic Platform 10.1</p> <p data-bbox="243 696 1176 765">Workaround: The user will need to manually update the 10.0 domain to a 10.1 domain. To upgrade the domain follow the instructions in the release note CR325421 below.</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR325421	<p data-bbox="314 388 1210 418">Upgrading Workshop for WebLogic 10.0 to 10.1: manual upgrade of domain may be required</p> <p data-bbox="314 435 1228 517">In moving from a Workshop for WebLogic 10.0 or earlier domain to Workshop version 10.1, you should initiate domain upgrade by using the IDE. This will make the updates to the domain so that you will have the latest runtime binaries for Workshop version 10.1.</p> <p data-bbox="314 534 1228 616">If you are managing the domain of an environment that doesn't have the IDE available (for example a platform in which the IDE is not supported or a deployed server) manual upgrade may be required.</p> <p data-bbox="314 633 807 663">Platforms: Workshop for WebLogic Platform 10.1</p> <p data-bbox="314 680 1005 710">Workaround: To add the 10.1 versions of the libraries to your domain.</p> <ol data-bbox="327 718 1036 777" style="list-style-type: none"> <li data-bbox="327 718 1036 748">1. Manually change the setDomainEnv script to point to the 10.1 binaries. <li data-bbox="327 756 1036 786">2. Start the server and deploy the following new libraries into the domain. <pre data-bbox="354 803 736 1020"> beehive-controls-1.0.2.1.ear beehive-controls-1.0.2.1.war beehive-netui-1.0.2.1.war -- new beehive-netui-resources-1.0.2.1.war jsf-myfaces-1.1.3.war -- new weblogic-controls-10.1.ear -- new weblogic-controls-10.1.war -- new </pre> <p data-bbox="314 1038 1228 1090">The following topics will help you deploy the libraries. using weblogic deployer and weblogic console.</p> <p data-bbox="314 1107 1036 1159">To deploy the libraries using WebLogic Deployer, see: http://edocs.beasys.com/wls/docs100/deployment/deploy.html#wp1020594</p> <p data-bbox="314 1194 1228 1272">To deploy the libraries using the WebLogic Server console, see: http://edocs.beasys.com/wls/docs100/ConsoleHelp/taskhelp/deployment/InstallApplicationsAndModules.html</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR336014	<p>A runtime exception occurs while invoking a 10.0 application with a service control on WebLogic Server 10.0 MP1</p> <p>A ServiceClassCacheException occurs when a 10.0 application with a service control is invoked after deploying the application on WebLogic Server 10.0 MP1. This exception occurs due to a mismatch in the serial version UUIDs for the javax.xml.namespace.QName class in JDKs 150_06 and 150_11.</p> <p>Platforms: All</p> <p>Workaround: Pass the following JVM argument to the WebLogic Server startup command in the startWeblogic script:</p> <pre>-Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0</pre>
CR342323	<p>Publishing of a web service application fails</p> <p>Publishing of a web service application fails when Workshop for WebLogic Platform 10.0 is upgraded to 10.0.1 using the upgrade installer because the upgrade process overwrites the plugins and the cache becomes obsolete.</p> <p>Platforms: All</p> <p>Workaround: After the upgrade, run workshop4WP.exe with <code>-clean</code> and <code>-initialize</code> options to refresh the cache.</p>
CR342348	<p>Unable to edit Installed Runtimes after upgrading from 9.2 MP2 to 10.0</p> <p>Workshop returns an error when editing Installed Runtimes after performing an application upgrade from 9.2 MP2 to 10.0.</p> <p>Platforms: All</p> <p>Workaround: Delete the 9.2 Installed Runtimes before editing the 10.0 Installed Runtimes.</p> <p>Note: If the 9.2 workspace is opened in Workshop 10.0, two Installed Runtimes are created. One is created for 10.0 from where the workspace is opened and the other for 9.2 from where the workspace is originally created.</p>
CR346002	<p>An unexpected EOF error occurs when downgrading from 10.0 MP1 to 10.0</p> <p>Product Uninstaller in console mode returns a harmless EOF error and does not return a confirmation message on the successful completion of the downgrade.</p> <p>Platform: Red Hat Linux</p> <p>Workaround: None</p>

Table 1 Known Limitations in BEA Workshop Version 10.0 (Continued)

Problem ID	Description
CR346976	<p>Uninstallation of Workshop 10.0 is not properly completed</p> <p>After downgrading from Workshop 10.0 MP1 to 10.0, when you uninstall the 10.0 installation, some files in the BEA_HOME directory are not automatically removed.</p> <p>Platform: All</p> <p>Workaround: Clean up the directory manually.</p>
CR347082	<p>Product upgrade from 10.0 to 10.0 MP1 using Smart Update fails</p> <p>When Smart Update is used to upgrade Workshop 10.0 to 10.0 MP1, the upgrade fails with a fatal error on a HPUX 64-bit machine.</p> <p>Platform: HPUX 64</p> <p>Workaround: Use the package upgrade installer to upgrade from a 10.0 installation to 10.0 MP1.</p>

Resolved Problems for Workshop for WebLogic Version 10.0.1

Table 2 Resolved Problems for Workshop for Weblogic Version 10.0.1

Problem ID	Description
CR316572	<p>An infinite loop used to cause performance issue when building or modifying EJBs.</p> <p>This problem has been resolved.</p>
CR321172	<p>When building the Workshop Samples application after starting Workshop with a clean workspace, the build process used to hang.</p> <p>This problem has been resolved.</p>
CR327144	<p>A ModuleException used to occur when deploying a web service using Apache XML Beans as the binding type. This exception used to occur because XML Bean objects were not successfully identified as XML Bean artifacts.</p> <p>This problem has been resolved.</p>
CR329436	<p>The Ant tasks, build-type-library and generate-webservice-control, used to fail when building a web service control from a WSDL.</p> <p>This problem has been resolved.</p>

