BEA Workshop Product Family™

Release Notes

Version 10.1
Revised: July 2007

Copyright

Restricted Rights Legend

Trademarks and Service Marks

Contents

BEA Workshop Version 10.1 Release Notes

This document contains information on the following subjects:

What's New in Workshop Version 10.1?

Workshop for WebLogic and Workshop Studio have Merged

As of version 10.1, all of the functionality of WebLogic for Workshop and Workshop Studio have merged into one IDE.

To migrate your applications to the new IDE, see Upgrading to Workshop 10.1.

New features include:

- Support for multiple web application frameworks in one IDE: Struts, JSF and Beehive
- Flexible tools for creating and testing WebLogic web services
- DbXplorer and DBXaminer for working with databases

- ORM mapping support via BEA Kodo and Hibernate
- Automatic generation of Spring configuration and DAO classes

Deployment to versions 8.1, 9.x, and 10.0 of WebLogic Server

Workshop 10.1 now supports deployment to these versions of WebLogic Server:

- 8.1
- 9.0, 9.1, 9.2
- 10.0

  Note: Facet support for pre-9.2 versions is limited to Struts, JSF, JSP, and Spring. Development of Beehive, Workshop 8.1 runtime, WLS web services and EJBGen are not supported for deployment to pre-9.2 versions.

Improved Page Flow Support and AppXRay Integration

- JSP editor with WYSIWYG design view
- Code completion on NetUI tag attributes, both for static values and expressions
- Ctrl+click navigation on NetUI tag attribute values
- Page flow specific validation on NetUI tag attribute values
- Rich property sheet support via with lots of useful “value pickers”
- Integration with WTP JSP templates for very easy customization of initial JSP content

Workshop Quick Start Applications

Instead of creating one project at a time to build up an application structure, Workshop Quick Start Applications let you create multiple projects simultaneously. Project dependencies are created already in place, as is a WebLogic server configuration. For details see Workshop Quick Start Applications.

Support for Maven Build Integration

Workshop's Ant-based build supports integration with Maven-based build infrastructures.

Built on Eclipse 3.2.2 and WTP 1.5.4

Workshop version 10.1 is built on the Eclipse Platform, an open source framework that is now widely used for Java development. Workshop extends Eclipse 3.2.2 and the Web Tools Platform 1.5.4.

Considerations for Using Workshop Version 10.1

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 e-docs site.

Location of Workshop Source Code

To comply with the Eclipse Public License, BEA has made source code for derived work available for download at
https://submit-codesamples.projects.dev2dev.bea.com/servlets/Scarab?id=S372

Known Limitations for Workshop 10.1

Table 1 lists the known limitations found in Workshop for WebLogic.

Table 1 Known Limitations in BEA Workshop Version 10.1

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 the current version. Note that the original source files are not altered. **Platform**: All **Workaround**: Either drag and drop the 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.
CR264808	**Error is displayed in the New Server wizard even if an 8.1 domain is successfully upgraded** 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. **Platform**: All **Workaround**: The problem can be worked around by re-selecting the text in the **domain home** combo box.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR267837	**Javadoc attachments for Workshop libraries can be broken if Workshop for WebLogic is not installed into the default directory under <BEA-HOME>** During installation there is an option to specify the directory into which Workshop is installed. If you specify something other than the default, then the Javadoc for Workshop 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 **Platform**: All **Workaround**: Add Javadoc attachments to the jars contained in those libraries manually. For example, to add a javadoc attachment to the base Workshop 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.
CR267912	**Source Not Found when debugging JDK classes** 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. **Platform**: All **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. 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>/jrockit90_150_04/ and select src.zip. 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>/jrockit90_150_04/ and select src.zip.
CR271247	**Unexpected server error may require IDE restart** 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. **Platform**: All **Workaround**: Restart Workshop for WebLogic and restart the server.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR282777	**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** 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. **Platform**: All **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.
CR283022	**Xbean wrapper classes generated for a ServiceControl must not be visible to the target JWS.** 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. **Platform**: All **Workaround**: Place Service Controls that use wrapper types in a separate project than the Service that they call. If the SerivceControl classes are loaded from a utility project, then the Service Control and the target Service must be in separate applications.
CR283457	**XBeans are not supported as parameters or return type for doc/lit/bare operations.** 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. **Platform**: All **Workaround**: Use doc/lit/wrapped for services that use XBeans as parameters or return types.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR285560	**Upgrade does not copy some MBCS characters in .jcx files** 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. When this happens, the resulting file will have "????" in place of the original characters, which will not compile. **Platforms**: All **Workaround**: 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	**WebLogic EJB project properties not visible from generated Ant build scripts** **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. **Platforms:** All **Workaround:** Prior to building the project with the exported Ant script, specify all desired "Jar settings" properties using weblogic.ejbgen.JarSettings annotations in your EJB Java source files and add and desired EJBC flags directly to the build script where "weblogic.ejbc" is executed.
CR287585	**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.** 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. **Platforms**: All **Workaround**: To workaround this issue, you can either 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 10.1 (Continued)

Problem ID	Description
CR293197	**Lost JVMTI events (especially breakpoints) when debugging with JRockit** When debugging with the JRockit JVM you may experience performance problems and missed breakpoints. **Platforms:** All Workaround: Update to JRockit 5.0 R27.1 or later.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description

CR294199	**Invocation of buffered Control methods will fail at runtime if deployed (via the IDE) to a server with more than one JMS Server**
	
	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:
	
	`<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.>`
	
	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:
	
	`"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"`
	
	Note that this situation is most likely to occur when using domains that were not created with support for "Workshop for Weblogic Platform.
	
	Platforms: All
	
	Workaround:
	Manually deploy the library using the weblogic.Deployer command. The form of the command is:
	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 -libspecver 10.0
	-libimplver 10.0
	Where:
	WseeJmsServer is the name of the JMS Server to host the application-scoped

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR295684	**Clicking cancel during 8.1 Upgrade may cause exceptions.** During upgrade of an 8.1 application, clicking cancel in the upgrade preview window may result in an exception. **Platforms**: All **Workaround**: These exceptions may be safely ignored.
CR299999	**The Perforce Eclipse plugin (P4WSAD) does not add new .prefs files after upgrading an application from version 9.2** **Description:** When you upgrade a Workshop for WebLogic version 9.2 application the Perforce Pending Changelists view does not show all of the files created in the upgrade process. For example, the com.bea.workshop.wls.core.prefs which replaces the com.bea.wlw.libmodules.core.prefs file is not added to the view. **Platforms:** All **Description:** Add the files manually by selecting **Team > Open for Add** at the project level. The new .prefs file and any other new files added to the project will be added to the view.
CR301304	**When upgrading web applications, the Page Flow Controllers source folder may revert to the default location in some cases** 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. 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. **Platforms**: All **Workaround**: To change the value, select **Project > Properties > Associated Files Model > Page Flow Controllers** and select the source folder of interest.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description

CR301661	**Buffered methods on ServiceControl may fail over JMS protocol**
	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).
	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:
	`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.`
	The resulting error message will include text similar to:
	`javax.xml.rpc.soap.SOAPFaultException: Failed to receive message java.io.IOException: Request timed out`
	Note: this only occurs when the transport protocol for the request is JMS.
	Platforms: All
	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:
	`@MessageBuffer @TransactionAttribute(TransactionAttributeType.NOT_SUPPORTED) public void voidMethod();`
	Note that the presence of the @TransactionAttribute will not change the transactional behavior of actions that occur within the calling application.

CR303707	**Errors can result when importing Workshop 9.2 projects using Perforce plugin.** When Workshop 9.2 projects are imported 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. **Platforms**: 10.x **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**.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR304008	**Servlet 2.5 Implementation In WebLogic Server 10.0 Can Break 9.2 Beehive Applications** **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. **Platforms:** WebLogic Server 10.0 or higher **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: For each .war in the 9.2 .ear, modify the following section of the web-inf/weblogic.xml: <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> to look like: <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>

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR304502	**Users 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** 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 See the section labeled: 'Controls are Not Automatically Run Within the Scope of a Transaction' Users 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. 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). 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. **Platforms**: All **Workaround**: There are several workarounds available: 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. 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	**Need to undeploy applications before upgrading 9.2 domains** 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. **Platforms**: All

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR307678	**Problems tab does not display correctly after upgrading a 9.2 workspace** After opening a 9.2 workspace in Workshop 10.x, the columns of the Problems view are displayed incorrectly. **Platforms**: All **Workaround**: Close and reopen the Problems view. To reopen, select **Window > Show View > Problems**.
CR308313	**Java Web Services (JWS) Do Not Allow Nested Types** The Web Service stack in WebLogic Server 9.x/10.x 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 **Platforms**: 9.x and 10.x **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.
CR308749	**Duplicate simple class names are not supported for web service controls with callbacks** When multiple 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 usable 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 condition was present. This was because the ant script did not attempt to run jwsc on garnered java files. **Platforms**: All **Workaround**: When using web service controls (SerivceControl) with callbacks make sure that each control file has a unique un-qualified class name. Differing the package name is not sufficient.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR311539	**Some xml files are generated using VM encoding instead of XML header encoding** In some cases generated XML files may be generated using VM encoding instead of XML header encoding. **Platforms:** Workshop 10.1 **Workaround:** 1. Open the XML file using the Workshop XML Editor. 2. Change the file property encoding to VM encoding. 3. When Workshop shows the Conflict Encoding dialog, click Yes. 4. Modify the encoding attribute to the preferred one (usually UTF-8). 5. Save the XML file. 6. Change the file property encoding back to default.

CR313306	**Service control generation can produce incorrect callback method names**
	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.
	For example, the following WSDL callback names:
	`a_b`
	`a4b`
	will result in the generation of the following callback method names:
	`a_B`
	`a4B`
	Note that the first character after a number or underscore has been capitalized.
	When the method is invoked, the following error will result.
	`javax.xml.rpc.JAXRPCException: SOAPFaultException - FaultCode [{http://schemas.xmlsoap.org/soap/envelope/}Client] FaultString [Failed to get operation name from the incoming request]`
	Platforms: 10.x
	Workaround: Avoid callback method names with a lower case letter following an underscore or number.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description

CR321162	**JSF Managed Bean Configuration Cannot Be Named the Same as a NetUI Implicit Object**
	NetUI defines implicit objects for use in databinding (e.g., 'backing', 'bundle', 'pageFlow', 'sharedFlow' and 'pageInput'). If a JSF managed bean is created and the name used in the faces-config.xml for that managed bean is the same as the NetUI implicit object names, a runtime FacesException for a bean property could occur. The following is an example that will cause this FacesException:
	`<managed-bean>`
	`  <managed-bean-name>backing</managed-bean-name>`
	`  <managed-bean-class>com.bea.example.MyBean</managed-bean-clas`
	`s>`
	`  <managed-bean-scope>request</managed-bean-scope>`
	`  <managed-property>`
	`  <property-name>minimum</property-name>`
	`  <property-class>int</property-class>`
	`  </managed-property>`
	`</managed-bean>`
	This issue will only occur when you are using JSF in conjunction with Page Flows and NetUI.
	Platforms: Workshop 10.1
	Workaround: Make sure you do not use any NetUI implicit object names for the <managed-bean-name> element of the faces-config.xml.

CR325304	**Auto deployment of new J2EE libraries to a 10.0 domain could lead to a class not found during production deployment** 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. However when the user deploys the application to the 10.0 domain, the domain may not have access to the newer libraries. **Platforms:** Workshop 10.1 **Workaround:** Manually update the 10.0 domain to a 10.1 domain. To upgrade the domain follow the instructions in the release note CR325421 below.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR325421	**Upgrading Workshop for WebLogic 10.0 to 10.1: manual upgrade of domain may be required** 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. 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. **Platforms:** Workshop 10.1 **Workaround:** To add the 10.1 versions of the libraries to your domain. 1. Manually change the setDomainEnv script to point to the 10.1 binaries. 2. Start the server and deploy the following new libraries into the domain. 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 The following topics will help you deploy the libraries. using weblogic deployer and weblogic console. To deploy the libraries using WebLogic Deployer, see: http://edocs.beasys.com/wls/docs100/deployment/deploy.html#wp1020594 To deploy the libraries using the WebLogic Server console, see: http://edocs.beasys.com/wls/docs100/ConsoleHelp/taskhelp/deployment/InstallApplicationsAndModules.html.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR326326	**Workshop startup.jar cannot be used to run in headless mode on Linux unless an additional system property is defined** The startup.jar provided with Workshop cannot be used to run in headless mode on Linux, unless the following system property is set to true: m7.disable.swt.init **Platforms:** Workshop 10.1 **Workaround:** Launch workshop in headless mode with the system property m7.disable.swt.init set to true. Note that the DISPLAY system property must not be set for this workaround to succeed.
CR326466	**Issue with Hibernate JPA deployment on Weblogic Server 10.0** Hibernate JPA projects created in Workshop 10.1 or imported from Workshop Studio 3.x fail to deploy on Weblogic Server 10.0. The below workaround will help Hibernate JPA projects deploy but redeployment will require a server restart **Platforms:** Workshop 10.1 **Workaround:** For EAR Projects - Modify weblogic-application.xml and add the following <prefer-application-packages> <package-name>antlr.*</package-name> <package-name>org.apache.commons.*</package-name> <package-name>org.apache.oro.*</package-name> <package-name>oracle.*</package-name> </prefer-application-packages> For Web Applications - Add the web project to an EAR and then modify weblogic-application.xml as described above. Note: Redeployment of the project requires a server restart.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR327575	**Manual edits to weblogic.xml and weblogic-application.xml are not automatically reflected in in-memory model** Manual edits to the weblogic.xml and weblogic-application.xml files are not automatically reflected in the in-memory model. For example, if you manually change the a library reference in weblogic.xml, that change will not show up in the WebLogic Deployment Descriptor in the Project Explorer view. **Platforms:** Workshop 10.1 **Workaround:** Close, re-open, and clean the project. (Right-click the project and select Close, Open and Clean in turn.)
CR327602	**After renaming project, old project is not undeployed from the server** If you rename a project that belongs to an EAR, the previous project name will not be undeployed from WebLogic Server. **Platforms:** Workshop 10.1 **Workaround:** To remove the project from the server, undeploy and redeploy the application from WebLogic Server.
CR327849	**Problems deploying EJB project upgraded using command line upgrader** You may encounter problems deploying an EJB project that has been upgraded using the command line upgrade tool upgradeStarter. **Platforms:** Workshop 10.1 **Workaround:** Upgrade the project using the IDE instead of the command line tool. To upgrade using the Workshop IDE.
CR329277	**WebLogic Server samples will not be installed if Workshop Runtime Framework is unselected** During installation of Workshop 10.1, if the Workshop Runtime Framework entry is unselected the samples directory at BEA_HOME/wlserver_10.0/samples will not be installed. **Platforms:** Workshop 10.1 **Workaround:** Reinstall and select Workshop Runtime Framework.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR329690	**Domain upgrade fails on Linux** Linux users who run the domain the upgrader from the IDE will see the following error: "./upgrade.sh -plan=resources/wlw-exec-plan.xml Calling Wizard framework for upgrade: ..." **Platforms:** Workshop 10.1 **Workaround:** Edit the file upgrade.sh located in BEA_HOME/wlserver_10.0/common/bin. Replace the line: ${JAVA_HOME}/bin/java -Dprod.props.file=${WL_HOME}/.product.properties -classpath ${PRE_CLASSPATH}:${WEBLOGIC_CLASSPATH}:${POST_CLASSPATH}:${POINTBASE_CLASSPATH} weblogic.Upgrade ${ARGUMENTS} -type ${TYPE} -out ${LOG} With: ${JAVA_HOME}/bin/java -Dprod.props.file=${WL_HOME}/.product.properties -classpath ${PRE_CLASSPATH}:${WEBLOGIC_CLASSPATH}:${POST_CLASSPATH}:${POINTBASE_CLASSPATH} weblogic.Upgrade -type ${TYPE} ${ARGUMENTS} -out ${LOG}.

Table 1 Known Limitations in BEA Workshop Version 10.1 (Continued)

Problem ID	Description
CR329755	**Cannot create projects where workspace contains space in directory path** Users cannot create projects in workspaces which contain a space in their directory path. If you attempt to create a project in such a workspace, you will see the error: “The application location is not valid. The character ' ' is not legal in directory names.“ **Platforms:** Workshop 10.1 **Workaround:** Cancel out of the project creation wizard and create a new workspace without any spaces in the directory path. Note that you cannot resolve this issue by unselecting “Use default location“ and specifying a path with no spaces.
CR330719	**Problems using HQL query with Hibernate and Weblogic Integrated Commons Logging facets** Users may experience problems when running HQL queries in the HQL Editor of DBXaminer in a project where both the Hibernate and WebLogic Integrated Commons Logging facets are activated. In this case the Query Result tab will show "weblogic/logging/LogEntry" in the table. **Platforms:** Workshop 10.1 **Workaround:** Either remove the WebLogic Integrated Commons Logging facet from the project or place an empty commons-logging.properties file in the root of the src directory of the project. Note that this will disable the bridge that integrates Commons Logging with the WebLogic log mechanism, so be sure to re-enable it for deployment by removing the empty commons-logging file or adding back the facet if that functionality is desired.