

Release Notes for WebLogic Portal and WebLogic Integration

For the release notes for WebLogic Portal, see the following link:

<http://edocs.bea.com/wlp/docs81/relnotes/index.html>

For the release notes for WebLogic Integration, see the following link:

<http://edocs.bea.com/wli/docs81/relnotes/index.html>

WebLogic Workshop Enhancements

Service Pack 4 includes the following enhancements to WebLogic Workshop 8.1:

Significantly improved build performance. Some improvements to build performance include: building EJBs for controls referenced from page flow (JPF), JSP, web service (JWS), and JPD files, on an as-needed basis; performing up-to-date checks on Java and Control projects to determine whether building is necessary; generating fewer EJBs upon building controls; and the ability to build without performing a clean operation first. For more details about build performance improvements in the 8.1 SP4 release of WebLogic Workshop, see [Understanding the Build Process](#) and [Using the Build Process More Efficiently](#).

1. Launch WebLogic Workshop.
2. Create a new application by choosing *File-->New-->Application*.



results when you try to start WebLogic Server from the IDE. The server may not appear to start, and the progress dialog may not be dismissed.

Workaround: Install WebLogic Workshop into a directory whose name contains no spaces.

Windows: Use WebLogic Workshop as User with Administrator Privileges

You should install and use WebLogic Workshop on Windows as a user with Administrator privileges. Otherwise, you may encounter the following error when running a WebLogic Workshop application.


```
memoryInitialSize="8m"  
memoryMaximumSize="256m"  
fork="true"  
failonerror="true" />
```

Edit the `memoryMaximumSize` attribute to allocate more memory to the Schema project compilation, save the `build.xml` file, and re-compile the project.

Class File Might Not Be Removed from File System When Corresponding Source File is Deleted or Renamed

WebLogic Workshop might not remove a class file from the file system when the corresponding source file is deleted. This can result in unexpected behavior if your code is expecting those classes to no longer be present.

Workaround: Delete old classes from the `WEB-INF/classes` directory and from the `.workshop` directory, or run the `cleanApp` command described above.

WebLogic Workshop Help Topics: Topic URL

The URL for a topic in WebLogic Workshop Help does not appear in the browser address bar. The unique URL for each topic in Help is available at the bottom of the topic. Because these URLs are generated by script at the time of viewing, they are not indexed by the full-text search engine, and thus are not searchable.

EJB Control Wizard Creates EJB Control that Incorrectly References Local Rather Than Remote Interfaces

In you generate an EJB with the EJB control wizard, then the generated EJB control file will always extend the local interface instead of the remote interface, even if you use the wizard to select the EJB's remote JNDI name.

Workaround: After the EJB control file has been created, edit the file to extend the remote home and bean interfaces rather than the local interfaces.

Changes Made through the XQuery Mapper Dialog are not Committed to the Web Service Source File

In some cases, when you edit an XQuery map through the XQuery Mapper dialog, your changes are not saved to the XQuery map in the JWS source file. To ensure that your edits are saved to the JWS file, click the Test tab of the XQuery Mapper dialog before you exit the dialog.

WebLogic Server Startup Scripts

In the SP2 release, the scripts for starting WebLogic Server (`<BEA_HOME>\weblogic81\samples\domains\workshop\startWebLogic.sh`, etc.) no longer support the `-memmax` flag. You should use the environmental variables `JAVA_OPTIONS` and `MEM_ARGS` instead.

Package Change from `weblogic.jws.control.DatabaseFilter` to `com.bea.control.DatabaseFilter`

As of the WebLogic Workshop 8.1 release, the package `weblogic.jws.control.DatabaseFilter` has been changed to `com.bea.control.DatabaseFilter`.

Changes to JAR Files Modified Outside of WebLogic Workshop Are Not Automatically Reflected in Application Window

XA Drivers and Conversational State

Note that the error does not occur when you use a JMS Crcs.2 not oc4of the ertype inS Cweb service. Also, withinS page

Workaround: Rebuild the Schemas project by right-clicking the Schemas project in the Application pane and choosing Build Schemas.

No Warning/Exception when Client Certificate is not Obtained

The transport security methods `useClientKeySSL()`, `setClientCert()`, and `setKeystore()` don't generate a warning or throw an exception when the client certificate is not retrieved successfully, for instance, because the key doesn't exist, a wrong password was used or the keystore was corrupted.

@common:security runAs Annotation

The `@common:security runAs` annotation is supported on a control file (JCX or JCS file) only when the control is called from a page flow file (JPF), not from a web service (JWS) or business process (JPD).

Workaround: Use the `@common:security runAs` annotation directly on the web service or business process file, not the controls called from those files.

If a load balancer and a cluster are configured in such a way that the load balancer can talk to the cluster but the cluster cannot talk to the load balancer, a problem may result with an application consisting of a web

If you do not have any applications that are currently using the Tuxedo Control, simply create a Web Service project in a new empty application. Install the Tuxedo Control by right-clicking on the name of the application in the Application Pane and select Install -> Controls -> Tuxedo. This installs the TuxedoControl.jar file into the application and unpacks the sample files. The samples are available in <BEA_HOME>/<WEBLOGIC_HOME>/samples/partners/Tuxedo directory.

With SP02 and later, the Tuxedo Control help files are available in the WebLogic Workshop help content.

Migrating Applications that Use Tuxedo Control 8.1 to SP04

WebLogic Workshop 8.1 SP04 contains a new version of the Tuxedo Control. If you previously had WebLogic Workshop 8.1 or 8.1 SP02 or SP03 installed on your system and you upgrade to WebLogic Workshop 8.1 SP04, follow these steps to upgrade the Tuxedo Control.

1. Delete the Tuxedo Control directory and all of its contents from the <BEAHOME>\ext_components directory. Make sure you do not delete the TuxedoControl.zip file that is contained in ext_components directory.
2. If you have existing applications that use the Tuxedo Control, open each application and perform the following steps:

CR122600 Tuxedo Control does not serialize/deserialize inherited fields.

CR125169 Tuxedo Control does not accept the WTC TypedBuffer classes directly as input to or return values from control extension methods.

CR128102 The Tuxedo Control configuration wizard does not recognize a WLS domain with fully qualified domain name, for example machine.beasys.com.

CR131562 Tuxedo Control displays an exception error when using old style network address format in configuration.

CR136225 Tuxedo Control does not handle dynamic field tables.

CR176837 Tuxedo Control Tuxedo service using alias name fails.

CR176963 Tuxedo Control Service call using the default WTC configuration using the configuration wizard fails.

Tuxedo Control Problems Fixed in SP02

CR106017: Need to support Java primitives instead of just wrapped primitives.

CR106299: Add support for JavaBean style setters and getters.

CR110291: Unable to import services from Tuxedo 6.5.
