Oracle® WebLogic Server

Known and Resolved Issues

10*g* Release 3 (10.3)

August 2008

ORACLE®

Oracle WebLogic Server Known and Resolved Issues, 10*g* Release 3 (10.3)

Contents

WebLogic Server Known and Resolved Issues

WebLogic Server Known and Resolved Issues

The following sections describe known problems in WebLogic Server 10.3, as well as problems that were resolved in earlier versions. Entries include a description of the problem, and a workaround or solution where appropriate. A notation in the Fixed In column indicates that the problem has been resolved.

For information about new and changed functionality in WebLogic Server, see "What's New in WebLogic Server".

Administration Console

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR063594	Information about cached JDBC statements is not displayed on the JDBC Monitoring pages.	9.0 Beta	Will not fix
CR200627	After a page flow completes in the Administration Console, it forwards to a different page, typically a table. Pressing the browser Back button at this point results in an attempt to load the last JSP file in the completed assistant. At this point, all of the context for this assistant is discarded. **Workaround or Solution:** It is recommended that you do not use the browser Back button to step back into an assistant once changes are cancelled or finished, and that you do not go back to a previous step in an assistant. Instead, use the navigation links and buttons in the Administration Console.	9.0	Will not fix
CR208373	Message-Driven Beans (MDB) may specify adapter-jndi-name in the deployment descriptor to indicate that the MDB is receiving messages from a resource adapter rather than from JMS. The Administration Console does not currently provide a way to view or configure the Resource Adapter that an MDB is bound to. MDBs may specify `adapter-jndi-name` in the deployment descriptor to indicate that the MDB is receiving messages from a resource adapter rather than from JMS. Such a binding entails configuring an `ActivationSpec` object which is passed to the RA during deployment. The Administration Console does not currently provide a way to view or configure the `ActivationSpec` properties on the MDB.	9.1	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR300921	Some portions of the Console desktop were represented in the portal configuration only as portlets. This made it challenging for a Console extension to individually replace or enhance a single portlet.	9.2	10.3
	In WLS 10.3, the following new portal books and pages have been added:		
	• ChangeManagerBook, ChangeManagerPage wrap the change center portlet		
	• NavigationBook, NavigationPage wrap the navigation tree portlet		
	• QuickLinksBook, QuickLinksPage wrap the quick help links portlet		
	• SystemStatusBook, SystemStatusPage wrap the system status portlet		
	• ToolbarBook, ToolbarPage wrap the tool bar portlet		
	• LocationContextBook, LocationContextPage wrap the bread crumb portlet		
	• WorkpaceMessagesBook, WorkpaceMessagesPage wrap the messages portlet		
	Any Console extensions that replaced these portlets through other means should continue to work as before. They could, however, be re-implemented to leverage the new portal structure.		
CR306634	The WebLogic Server Administration Console includes pages which can be used to test Web applications, resource connectors, Web Services, and other deployments. Test pages are included for EJBs, but the test points for these listed EJBs are never available.	9.2	
	The omission of these EJB test points does not imply that the EJBs are not reachable or working. Administrators can use the console's EJB monitoring pages to monitor the activity and status of their deployments.		
CR333646	On the *Server_name* > **Configuration** > **Federation Services** >**SAML 2.0 General** tab, the following three fields in the Artifact Resolution Service section are not editable:	10.3	
	• Only Accept Signed Artifact Requests		
	• Artifact Cache Size		
	• Artifact Cache Timeout		

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR334479	Under some circumstances, canceled policy changes were subsequently saved. On Console policy pages which manage multiple resources (for example, all the methods in an EJB or a Web Service), if an administrator changed the conditions for a policy on one method, then selected another method, the following prompt was displayed:
	
	`Do you want to save your changes to the conditions before leaving this page? (Click OK to save your changes. Click Cancel to discard them.)`
	
	If the administrator clicked **Cancel** and subsequently specified conditions for a different method, clicking **Save** recorded the conditions for both resources.
	
	In WLS 10.3, **Save** and **Cancel** work correctly under these circumstances, and cancelled changes will no longer be saved.
	
	Also note that the confirmation prompt has changed as follows:
	
	`Are you sure you want to navigate away from this page?`
	
	`You have unsaved changes. To save your changes press Cancel to stay on this page. You can then complete your changes and save them.`
	
	`Press OK to continue, or Cancel to stay on the current page.`
		9.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR341605	There are two ways in the Administration Console to configure the domain's user lockout policy:	9.2	10.3

Method 1
 The domain's user lockout policy page:

1. In the right pane of the Console Home page, select **Domain** in the **Domain Configurations** section.
2. Select the **Security > Policies > User Lockout** tab.

Method 2
The default realm's global policy table:

1. Under **Domain Structure** in the left navigation pane, select **Security Realms**.
2. In the **Realms** table, click the name of the default realm.
3. Select the **Roles and Policies > Realm Policies** tab.
4. Click the + next to **Domain** in the **Policies** table.
5. Under **Domain**, click the + next to your domain name.
6. On the **Unlock User** row of your domain, click **View Policy Conditions** in the **Policy** column.

If you have a domain that was created prior to 10.3, and you used the global policy table (Method 2) to modify the domain's user lockout policy, the policy did not take affect and the user lockout policy from the domain's user lockout policy page (Method 1) was used instead. That is, the security policy you configured was ignored.

This is only an issue if you configured the user lockout policy by using the global policy table. It is not an issue if you used the domain's user lockout policy page.

If you used the global policy table (Method 2) to set the domain's user lockout policy in a pre-10.3 domain, when you migrate the domain to 10.3, you have to manually configure the user lockout policy again.

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR341605 (cont.)	Do this by navigating to the domain's global policy table's user lockout node, as follows: 1. Under **Domain Structure** in the left navigation pane, select **Security Realms**. 2. In the **Realms** table, click the name of the default realm. 3. Select the **Roles and Policies > Realm Policies** tab. 4. Click the + next to **Domain** in the **Policies** table. 5. Under **Domain**, click the + next to your domain name. 6. On the **User Lockout** row of your domain, click **View Policy Conditions** in the **Policy** column.	9.2	10.3
CR367671	In prior releases, an `IllegalArgumentException` would sometimes occur after clicking **Activate Changes** when creating or updating a request class on the **Configuration** tab for a Work Manager instance in the Administration Console. In WLS 10.3, the changes to the Work Manager MBean will activate successfully and the `IllegalArgumentException` error no longer occurs.	10.3	10.3
CR368311	Under some circumstances, the Administration Console does not automatically update the request URL to follow administration port configuration changes. If you have the **Automatically Acquire Lock and Activate Changes** Console option enabled (which is the default for development mode) and change the Console's address (for example, turn on the domain wide administration port, create an administration channel, or change to the SSL listen port), the Console will not automatically redirect to its new address. Instead, the browser will display an error. **Workaround or Solution:** Enter the URL address and protocol in the browser where the administration server is now listening for requests (for example, switch from `http://localhost:7001/console` to `https://localhost:9002/console`).	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR368482	The Administration Console does not always reflect external changes made in a deployment plan. If a change is made in a deployment plan outside of the Console (for example, using Workshop, editing the plan text files directly, or updating a deployment with a new plan using WLST or WebLogic.Deployer) while a Console user is also viewing that deployment plan, the Console user will not see those changes. **Workaround or Solution:** Navigate to a configuration page for a different deployment, then navigate back to the original deployment again.	10.3	
CR370004	The Administration Console permits the creation of Work Manager configurations that are not supported and do not function as intended. Incorrect Work Manager configurations may result in a number of exceptions being recorded in the server logs, most commonly `Validation problems were found` exceptions while parsing deployment descriptors. **Workaround or Solution:** Follow the guidelines described in the online help for Work Manager configurations. Specifically, you can only assign one request class to any given Work Manager, and that request class must be of the same or a broader scope than the Work Manager. You should not assign an application-scoped request class to a global Work Manager, and you should not create more than one application-scoped request class for an application-scoped Work Manager. Correcting the Work Manager configurations to match the documented constraints resolves these issues.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR370078	To improve its level of accessibility to blind users, in certain situations, the Administration Console provides title attributes for links whose purpose or target needs a better explanation than the link text alone provides. The primary case is when links are part of a 'simulated tab' group, and of which 'tab' is currently selected. However, screen reader users need to make a configuration change for these link title attributes to be read instead of the visible link text.	10.3	
	The following instructions are provided for the leading screen reader, JAWS? (by Freedom Scientific):		
	To configure JAWS to read link titles:		
	1. With WebLogic Server as the active browser window, activate JAWS' Personalized Settings dialog by pressing Insert+Shift+V.		
	2. For JAWS versions 6.x, 7.x, and 8.x:		
	Navigate to the **Links With Text Only** setting, then select the **Title** option (toggle through the values by pressing the spacebar).		
	For JAWS version 9.x:		
	Expand the **Links Options** node, navigate to the **Text Links** option, then select the **Show Using Title** option (toggle through the values by pressing the spacebar).		
	3. Select **Close** to save the configuration change.		
CR370592	The Server Status table on the **Cluster: Monitoring: Summary** page includes two default columns: **Primary** and **Secondary Distribution Names**. These fields do not always reflect all of the replication statistics that are collected and displayed on the **Cluster: Monitoring: Failover** page, depending on the replication scenario.	10.3	
	Please refer to the **Cluster: Monitoring: Failover** page for definitive information.		
CR373049	When using WLST Script Recording in the Administration Console, the current script operations are correctly written to the script file, but are not displayed on the **Preferences: WLST Script Recording: Control** page.	10.3	
	Workaround or Solution:		
	On the **Preferences: WLST Script Recording: General** page, set the file name to an explicit output file.		

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR373731	When defining security policies in the Administration Console for an EJB deployment that references types defined in a separate library deployment, exceptions can be observed if that library deployment is not available to the Console. **Workaround or Solution:** All library deployments should be targeted at the admin server as well as any managed servers needed to support referencing applications. This will ensure that when defining policies, the Console will have access to those library deployments so that referenced types can be class-loaded as needed.	10.3	
CR374586	In the Administration Console Online Help, some of the fields in the Persistence Plugin pages are missing content and/or contain links to MBean attribute pages that do not exist. For an example, see **Persistence: Plugins: Access Dictionary Configuration** at: http://edocs.bea.com/wls/docs103/ConsoleHelp/pagehelp/J2EEkodo jdbcconfdescriptorAccessDictionaryBeantitle.html This content will be updated in a future release.	10.3	

Apache Beehive Support

No issues at this time.

Clustering

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR333000	In prior releases, with consensus leasing, when a clustered server instance was resumed from the Admin state, the ClusterLeaderService would be started again even though it was already running. This would cause the server to fail. In WLS 10.3, the ClusterLeaderService no longer starts if it is already running, and the server resumes successfully from the Admin state.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR370084	In prior releases, when a clustered server using unicast messaging is artificially hung by pressing **Ctrl-z** on a Unix/Linux operating system (suspending the process), then the cluster messaging can throw SocketExceptions over a long period of time. This issue no longer happens in WLS 10.3. **Workaround or Solution:** Set the system property `-Dweblogic.unicast.HttpPing=true`. This flag should be needed only in test environments where running processes might be suspended when pressing **Ctrl-z**.	10.3	10.3

Connector (Resource Adapter)

No issues at this time.

Console Extensions

No issues at this time.

Core Server and Core Work Manager

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR189353	When a machine that is hosting one of the managed servers is abruptly shut down, a network cable is pulled, or its network interface card has issues, and any server attempts communication with that managed server, threads become stuck waiting to get a connection. **Workaround or Solution:** This can currently be resolved by using a private flag: `-Dweblogic.client.SocketConnectTimeoutInSecs` and setting an appropriate timeout value that will release the thread attempting to make the connection and allow the request to fail quickly.	8.1SP01	Will not fix
CR346384	In prior releases, when creating a server with the same name as a previously deleted server, an `IllegalArgumentException` occurs after clicking **Activate Changes**. The deleted server was not properly removed from the `ServerLifeCycleService`, causing the `IllegalArgumentException` when a new server was created with the same name as the deleted one. In WLS 10.3, the `IllegalArgumentException` does not occur.	10.3	10.3
CR355337	When using an IPv6-formatted address for WLS, the URL should include square brackets ('[' and ']') for the host address. Otherwise, WLST may fail to connect to the running server. **Workaround or Solution:** Add square brackets to the host address. For example: `t3://[fe80:0:0:0:203:baff:fe2f:59e5]:9991`	10.0MP01	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR358428	If the Admin Server is down when a Whole Server Migration occurs for a clustered server, and the server migrates to a machine on which it was never run before, the server cannot be started on the new machine. **Workaround or solution:** Use one of the following workarounds for this issue: • Ensure that the Admin Server is up when the server migration is being performed. • Use a shared disk/NFS for all the migratable servers in the cluster	10.3	
CR358477	WorkManagerFactory's `findOrCreate()` API will only look into globally scoped Work Managers before creating a new Work Manager. As a result, the `findOrCreate()` API would create a new Work Manager, even when an application-scoped Work Manager of the same name is already defined. In WLS 10.3, WorkManagerFactory's `findOrCreate()` API will first look into globally scoped Work Managers and then application-scoped Work Managers before creating a new Work Manager.	10.0MP01	10.3
CR361830	In prior releases, deleting and recreating a Work Manager caused an `IllegalArgumentException`. In WLS 10.3, an `IllegalArgumentException` will not occur when deleting and recreating a Work Manager.	10.3	10.3
CR365822	WebLogic Server internally creates a Work Manager with a default name for every configured application. However, it is possible to explicitly override the default Work Manager by creating a Work Manager with the default name. This will cause an `IllegalArgumentException` and the Administration Console will show two default Work Managers for the same application. In WLS 10.3, if you override the default Work Manager, you will see only the overridden default Work Manager associated with the application.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR365859	In prior releases, Work Manager targeting was not being honored, so any globally defined Work Manager configurations were being instantiated for all deployed applications on all servers. In WLS 10.3, Work Manager targeting is now honored. Therefore, a globally defined Work Manager will only be available for applications to utilize on the servers to which it is targeted. Any applications that define dispatch policies which reference globally defined Work Manager configurations may be affected. If an application is referencing a global Work Manager configuration, the Work Manager must be targeted to the server where the application is deployed. Otherwise, the Work Manager instance referenced by the application will be unavailable.	10.3	10.3
CR366578	When Fastswap is enabled in a J2EE application, you can make certain types of changes to Java classes during development and expect to see the change without re-deploying, with all instance statesof the Java object being retained. One type of the change that does NOT retain the object state is that when a field name is changed, it is treated as follows: • the field with old name is deleted • the field with new name is added Thus, in this case, any state in the old field is not carried over to the renamed field. Using the Workshop or FastSwap ant task, you may see a `FastSwap operation completed successfully` message, even when an instance field name change causes a value reset. **Workaround or solution:** You should expect an instance value to be reset when you change a field name.	10.3	Will not fix
CR368836	In prior releases, when configured for unicast messaging, a cluster member would incorrectly create a log entry indicating that the server was listening on a multicast port, even though multicast messaging was not enabled. In WLS 10.3, the multicast log entry is reported only when multicast messaging is enabled for the cluster.	10.3	10.3

Deployment

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR067087	The `security-permission` element is available in the `weblogic.xml` and `weblogic-ejb-jar.xml` deployment descriptors, but is not available in the `weblogic-application.xml` descriptor. Therefore, in an Enterprise application, you can only apply security policies to JAR files that are EJBs or Web applications.	7.0	Will not fix
CR071138	The `weblogic.Deployer` tool interprets any extra string values between command-line arguments as a file specification. For example, if you enter the command: `java weblogic.Deployer -activate -nostage true -name myname -source c:\myapp\mymodule` the tool attempts to activate a file specification named `true`, because the `-nostage` option takes no arguments and `true` is an extraneous string value.	7.0	Will not fix
CR091020	If you deploy an application to a cluster and one or more clustered servers are unavailable (for example, servers partitioned from the cluster due to a network outage), the deployment operation may appear to hang. In addition, the partitioned servers may not deploy the application even after they successfully rejoin the cluster. **Workaround or Solution:** Reboot the partitioned servers after they rejoin the cluster.	8.1	10.0
CR220968	If you deploy a Web application with virtual hosts as targets, you cannot then change the targeting information unless you redeploy the entire Web application with new target information. **Workaround or Solution**: Redeploy the Web application with new target information.	9.0	10.0

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR282367	While using the WebLogic Administration Console with applications or EJBs deployed on a Managed Server that depend on a deployed library, you may encounter a `java.lang.NoClassDefFoundError`. **Workaround or Solution:** The WebLogic Server Administration Console needs access to any shared library deployments so that Java data types and annotations can be processed. Therefore, all shared library deployments should always be targeted to the Administration Server in addition to any Managed Servers or clusters.	9.0	Will not fix
CR297802	As part of the 9.0 release, WLS introduced the Admin state feature. Failures in the prepare phase of deployment cause the server to stay in the Admin state. If the `weblogic.deployment.IgnorePrepareStateFailures` system property is specified as `true`, then the server will not stay in the Admin state when failures for system resources occur in the prepare phase of deployment.	9.2	10.3
CR320305	Internal fields and methods were incorrectly documented in the API reference. These fields and methods have been deprecated and are no longer documented. You should remove any usage of these fields or methods. For a complete list, refer to "Deprecated Functionality" in *Release Notes*.	10.3	10.3
CR320503	In prior releases, after a rename, the run time deployment state contained information for both the old and new modules, instead of just containing information for the new child module TMIDs. This information was returned by the WebLogicDeploymentManager `getAvailableModules()` method. In WLS 10.3, WebLogicDeploymentManager `getAvailableModules()` returns only new child module TMIDs after rename.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR329849	Deployment may fail to complete and could timeout if a transient network error prevented the send of the completion message from the managed server to the administration server. In WLS 10.3, if the network error is transient, then deployment may succeed instead of waiting for the timeout. **Workaround or Solution:** Change the deployment service to retry the send. If the failure still occurs, ensure that the request is cancelled correctly.	9.1	10.3
CR330135	The cluster constraints setting is ignored by WLS deployment when deploying on a cluster. In WLS 10.3, WLS deployment was modified to honor the cluster constraints setting and fail deployment if all servers in the cluster are not running. If cluster constraints are enabled, then deployments targeted to a cluster succeed only if all servers in the cluster are running.	9.2	10.3
CR354822	If you start an edit session, install an application, and then undo the changes, a deployment task is not cleaned up appropriately. This task is left in the initializing state and causes issues with future Activate Changes.	10.3	

Documentation

Continuous improvements are made to the WebLogic Server documentation. The latest documentation for WebLogic Server is available from the WebLogic Server Product Documentation Web site at the following location:

http://e-docs.bea.com/wls/docs103/index.html

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR373628	The MBean Reference will not document the interfaces to the SAML 2.0 Identity Asserter and SAML 2.0 Credential Mapping provider. Instead, the Javadoc for these MBean interfaces has been generated in a separate location that will be accessible from e-docs. The sections in *Securing WebLogic Server* that explain how to configure and manage the SAML 2.0 providers have been revised to include a link to this separate location. For information about the MBean interfaces to the SAML 2.0 security providers, users need to refer to http://e-docs.bea.com/wls/docs103/saml2_javadocs.	10.3	
CR373811	1. The **Search** function in the MBean Reference does not work. 2. The Search function in the Samples viewer does not work when accessing the Examples documentation by selecting **Oracle Weblogic > Weblogic Server 10.3 > Examples > Documentation** from the Windows Start menu, or by clicking the Documentation link at the top of the Examples page. **Workaround or Solution:** There is no workaround for #1. For #2, to search the Sample Applications and Code Examples, you must start the Examples server and navigate to (http://localhost:7001/examplesWebApp/docs/core/index.html). Click **Instructions** and then **Search**.	10.3	
CR374891	The samples viewer displays the Japanese and English versions of some Avitek Medical Records topics simultaneously.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR375833	BEA WebLogic Express is no longer available from Oracle, therefore its description has been removed from the WebLogic Server 10.3 documentation. All WebLogic Express functionality is available and supported in other Oracle WebLogic Server products. You can upgrade your 10.0 and earlier WebLogic Express applications to WebLogic Server 10.3.	10.3	

EJBs

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR232765	The primary key in an Oracle table is a CHAR but the query field in the SQL table is a VARCHAR2. **Workaround or Solution:** Change the database schema from CHAR to VARCHAR2. Using CHAR as a primary key is not recommended for the Oracle database.	9.2	Will not fix
CR295394	There is no annotation for EJB3 beans or `Ejbgen` that enables creation of a clusterable timer. **Workaround or Solution:** Create a `weblogic-ejb-jar.xml` file and put the `<timer-implementation>` element and corresponding values into the file.	10.0	Will not fix
CR295746	When accessing a Stateful Session Bean, method invocation must acquire a lock first. When concurrent access occurs, the lock acquisition will fail first so that a `LockTimedOutException` wrapped in an `EJBException` will be thrown, rather than a `ConcurrentAccessException`.	10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR297054	Creating EJB 3.0 stateful session beans is much more expensive in terms of performance than creating equivalent EJB 2.1 stateful session beans. **Workaround or Solution:** Most of the performance difference can be regained by setting the system property `weblogic.ejb30.enableproxypool=true`. This system property is set to `false` by default. Note that it is not safe to turn on this flag if the bean has interceptors associated with it that contain state.	10.0	
CR303273	Kodo's `MappingTool` cannot generate schemas for classes that use BLOBs in their primary key. BLOBs can be used in a primary key, but the schema must be defined manually. Note that support for BLOB columns in primary keys is not mandated by either the JDO or JPA specifications.	10.0	
CR305173	Kodo components send log messages by default to the `.out` file of the server the bean is deployed to, rather than to a log file. **Workaround or Solution:** Configure the component log to a particular file.	10.0	10.3
CR306237	Extensions to the JPA metadata model can only be specified via annotations, and not via a structure similar to the `orm.xml` file defined by the specification. **Workaround or Solution:** To specify Kodo-specific metadata for your object model, either: • use the Kodo-specific annotations, or • convert your XML-based metadata to the JDO metadata format, which does support XML specification of extensions.	10.0	Will not fix

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR316709	When using the Sun VM, dynamic class enhancement sometimes does not occur. **Workaround or Solution:** Run the Kodo enhancer on your persistent classes after compilation but before deployment. You can do this with the `kodoc` script bundled with WebLogic Server, by using an ant task, or by running `java org.apache.openjpa.enhance.PCEnhancer`, passing your persistent classes as arguments to the process.	10.0	Will not fix
CR331713	The Weblogic Spring injection extension model doesn't support lookup method injection.	10.3	Will not fix

CR340082	The EJB container always uses an outer join in a CMP bean finder when eager relationship-caching is turned on. Typically, inner joins are faster to execute than outer joins with the drawback that inner joins do not return rows which do not have data in the corresponding joined table.
	
	In WLS 10.3, a new configuration element, `<use-inner-join>`, has been added in `weblogic-cmp-rdbms-jar.xml`, as an attribute of `<weblogic-rdbms-bean>`, as shown here:
	
	`<weblogic-rdbms-bean>`
	`<ejb-name>exampleBean</ejb-name>`
	`...`
	`<use-inner-join>true</use-inner-join>`
	`</weblogic-rdbms-bean>`
	
	This element should only be set to `true` if the CMP bean's related beans can never be null or an empty set.
	
	The default value of `<use-inner-join>` is `false`. If you specify its value as `true`, all relationship cache query on the entity bean will use an inner join instead of a left outer join to execute a select query clause.
		10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR341886	Currently, there is no way to serialize a business object in the EJB3 specification, which is different than a traditional component object. **Workaround or Solution:** When you need to serialize a business object, first invoke `BusinessObject._WL_getBusinessObjectHandle()` to get the business handle object, then serialize the business handle object. To recover from this serialization, just deserialize to get the business handle object, then invoke its `getBusinessObject()`.	10.0	10.3
CR344667	Deserializing a JDO `PersistenceManagerFactory` in a managed environment may fail. The exception states that the `javax.jdo.PersistenceManagerFactoryClass` property is missing. Note that serializing a `PersistenceManagerFactory` should not generally be necessary in a managed environment.	10.3	
CR347183	If the `initial-beans-in-free-pool` setting is greater than the `max-beans-in-free-pool` setting for an EJB, some of the EJBs instantiated during deployment are immediately discarded. This inefficiency can slow down the EJB deployment and the pool size may end up being smaller than was intended. In WLS 10.3, the size of the free pool will be automatically increased for EJBs whose configuration is such that `initial-beans-in-free-pool` is greater than `max-beans-in-free-pool`.	10.0	10.3
CR348280	For entity beans with a high cache miss ratio, maintaining ready bean instances can adversely affect performance. **Workaround or Solution:** By setting the flag `<disable-ready-instances>` in the `<entity-cache>` element of an `<entity-descriptor>`, the container will not maintain the ready instances in cache. If the feature is enabled in the deployment descriptor, the cache will only keep the active instances. Once the involved transaction is committed or rolled back, the bean instance is moved from active cache to the pool immediately.	10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR354705	When switching JDK, all EJBs in the server will be automatically recompiled. All ejbc options which affect application behavior and which are not specified in deployment descriptors will be lost during automatic recompilation. **Workaround or Solution:** Settings which affect application behavior, such as the dispatch policy, should be configured in the deployment descriptors instead of the ejbc command line. This will ensure the settings will always be applied.	10.0MP1	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR355546	When filtering out just the `org.apache.openjpa.*` packages (but not the `com.solarmetric.*` and `kodo.*` packages), deployment of the application will fail with an exception message similar to this:
	
	`java.lang.IllegalArgumentException: interface org.apache.openjpa.event.CallbackModes is not visible from class loader`
	
	Note that the particular class or interface cited in the exception message may vary.
	
	Workaround or Solution:
	
	When deploying an application-provided version of OpenJPA, all three Kodo-related packages must be filtered using the `prefer-application-` libraries directive:
	
	`<weblogic-application>`
	  `<prefer-application-packages>`
	    `<package-name>org.apache.openjpa.*`
	    `</package-name>`
	    `<package-name>kodo.*</package-name>`
	   `<package-name>com.solarmetric.*`
	   `</package-name>`
	   `</prefer-application-packages>`
	`</weblogic-application>`
	
	The Kodo and `com.solarmetric` packages must be filtered even if you want to disable all Kodo features (that is, only use OpenJPA).
	
	Additionally, if you want to provide your own version of `openjpa.jar`, but use the WebLogic-provided Kodo jar, the application must still exclude `kodo.*` and `com.solarmetric.*`, and the application must bundle the Kodo `jar` from the WebLogic distribution.
	
	Applications may also need to exclude `serp.*` and bundle their own version of it at some point in the future if new APIs or bug fixes are introduced in that codebase. However, there are no interdependencies with `serp` as there are between the `org.apache.openjpa.*`, `kodo.*`, and `com.solarmetric.*` packages.
		10.0	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR355853	Indexes declared at the class level are not always created during schema creation.	10.3	Will not fix
	Workaround or Solution:		
	Create the indexes manually after running the schema generation tools.		
CR363318	OpenJPA throws an exception when `@Id` fields are also annotated as `@Unique` in some databases. Database primary keys are unique by definition. Some databases implement this by creating a unique index on the column.	10.3	Will not fix
	Workaround or Solution:		
	Do not specify both `@Id` and `@Unique` on a single field.		
CR363375	In prior releases, the `DataCacheTimeout` attribute was initialized and cached inside the ClassMetaData the first time it was accessed. After a configuration update, `DataCacheTimeout` still used the original cached value, although the configuration was updated.	10.3	10.3
	This issue has been fixed in WLS 10.3. `DataCacheTimeout` will use the newly configured global default value, with one limitation. For those classes that have a `@DataCache` annotation with a `DataCacheTimeout` value equal to the old default value, the `DataCacheTimeout` value will be equal to the newly configured default. For example, if the global configuration is set to 10 seconds, and a `@DataCache annotation` override is 10 seconds, if the global default is updated to 20 seconds, the `@DataCache` annotation is also updated to 20 seconds because it is the same as the original global default value.		

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR365276	Using the Javac compiler in WLS 10.3 can degrade performance in WLS 10.3. **Workaround or solution:** The JDT project provides a set of plug-ins that add the capabilities of a full-featured JAVA IDE to the Eclipse platform. Use the JDT compiler for improved performance as compared to Javac. Also note that in WLS 10.3: 1. Both JDT and Javac is supported in the EJB container. JDT is the default option. 2. You can set up to use different compilers in `appc` and WLS: For `appc`, use `-compiler`, such as `-java weblogic.appc -compiler javac ...` For WLS, use the `<ejb-container>` tag in `config.xml`. For example: ``` <ejb-container> <java-compiler>jdt</java-compiler> </ejb-container> ``` Please refer to http://www.bea.com/ns/weblogic/920/domain.xsd for more information. 3. If you use JDT in `appc`, only the `-keepgenerated` and `-forceGeneration` command line options are currently supported. These options have the same meaning as when using Javac. Additional options will be supported in the future.	10.3	
CR366354	When using the Weblogic-supplied Sybase driver instead of the Sybase-supplied JDBC driver, attempting to insert a record in the database for an entity whose ID is auto-incremented can fail, as the default query to fetch the next available identity value always returns zero. **Workaround or solution:** Override the lastGeneratedKeyQuery property in the `kodo.properties` file as shown here: ``` openjpa.jdbc.DBDictionary: lastGeneratedKeyQuery='SELECT MAX({0}) FROM {1}' ```	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR366512 CR369221	Using generics in EJB will cause a problem in the following cases: 1. When the business interface extends `java.rmi.Remote`, and extends some generic methods from a super class, the deployment will fail. 2. When the business interface doesn't extend `java.rmi.Remote`, the invocation on the generic business methods will fail. **Workaround or Solution:** The first case is a limitation in WLS 10.3. The second case can be resolved by downloading the needed classes from the server side. If network downloading is disabled, however, the invocation will fail still. If network downloading isn't permitted in the user's environment, it is recommended that you run `appc` first, then add the generated classes to the `classpath` of the client side.	10.3	
CR368196	The default data cache's random eviction algorithm has changed to increase concurrency in OpenJPA 1.1.0, which is used in WebLogic Server 10.3. This algorithm change has also decreased the randomness of the eviction algorithms. As a result of this change, the data cache eviction process is less random than it used to be, especially for high ratios (greater than 16 to 1) of eviction-to-insertion.	10.3	
CR368558	The cache hit and miss counts may rise unexpectedly when manipulating entities without version data. The extra cache access occurs when the EntityManager closes and all contained entities are detached. Entities without version fields appear to the system to be missing their version data, and the system responds by checking their version in the cache before detachment. **Workaround or solution:** Entities with version fields or other version strategies do not cause extra cache access.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR368579 CR371182	The Sun JVM included with WLS 10.3 is JDK 1.6. After upgrading to WLS 10.3 from previous releases which used JDK 1.5, you may see a `java.lang.OutOfMemory Error: permGen space` error when running existing applications. **Workaround or Solution:** Increase the `MaxPermSize` value for the Sun JVM.	10.3	
CR369229	When trying to persist an empty byte array field within an entity to a Sybase database, the value gets stored as a NULL rather than as bytes. Therefore, the value is retrieved as NULL. This is a limitation of the Sybase drivers, which convert the empty byte array to a NULL while storing it in the database. This happens with both Weblogic JDBC drivers as well as the proprietary Sybase drivers.	10.3	
CR369427	When using the MySQL database, and OpenJPA is configured to automatically run the mapping tool at runtime and create tables within the default schema (for example, `<property name='openjpa.jdbc.SynchronizeMappings' value='buildSchema'/>` `<property name='openjpa.jdbc.Schema' value='<MySQL database name>' />`), OpenJPA will try to create the table even if the table already exists in the database. A PersistenceException will be thrown to indicate that the table already exists and the table creation statement fails. **Workaround or Solution:** To avoid this problem, if you are using the MySQL database, don't configure OpenJPA to automatically run the mapping tool at runtime and specify the default schema at the same time.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR370788	In previous WLS releases, it was possible that some classes could be loaded without enhancement if the module in which the persistence.xml was defined was declared after some of the dependent modules referencing those persistent classes (for example, if SB.jar is dependent on persistent classes defined in domain.jar, and domain.jar is declared after SB.jar in an EAR). In this case, the dependent module which does not have the persistence.xml declared in it gets loaded, and the persistent entity classes are not enhanced. This can result in the following error:	10.3	10.3

```
java.lang.NoSuchMethodError:
pcGetManagedFieldCount
    at com.bea.medrec.model.Prescription.
<clinit>(Prescription.java)
    at java.lang.Class.forName0(Native Method)
    at java.lang.Class.forName(Class.java:247)
```

Workaround or Solution:

In WLS 10.3, the classloader hooks for runtime enhancement for all the EJBs are installed first. The webapp modules are then processed in order after their classloader initialization completes. This resolves the issue.

There is a narrow possibility that the NoSuchMethodError message could still occur if all of the following conditions are true: (a) a custom class loader configuration is defined, (b) two webapp modules share the same classloader and persistence classes, (c) one of the modules has persistence.xml and the other module doesn't, and (d) the webapp module that has the persistence.xml is not declared or occurs earlier in the order of the modules of the EAR. In this case, some entity classes may load without enhancements. To prevent this, the modules should be ordered in the opposite way.

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR373734	EJB applications that use IIOP and send JPA entities from the server to the client will fail during deserialization if the entities are Serializable (but not Externalizable) and do not declare a `writeObject()` method.
	
	Workaround or Solution:
	
	Add a `writeObject()` method to such entity classes. The write object can be trivial:
	
	<pre>private void
	writeObject(java.io.ObjectOutputStream out)
	    throws IOException {
	  out.defaultWriteObject();
	}</pre>
		10.3	

Examples

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR191354	Medical Records does not yet use deployment plans.	9.0 Beta	
CR208965	The `medrec.wls.config` target in `SAMPLES_HOME/server/medrec/setup/build.xml` has a known issue with respect to security configuration.	9.0 Beta	Will not fix

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR226622	The `../xml/stax` example contains two files with the same root but different extension: `StreamParser.java` and `StreamParser.jsp`. The samples viewer build, however, creates just one corresponding HTML file, rather than two for each type of file. In this case only the `StreamParser.jsp` file has an equivalent HTML file; the `StreamParser.java` file does not. The problem occurs because of a setting in the `build.xml` file that controls the behavior of `java2html` to generate the files for the documentation. When using `java2html`, the `useShortFileName="true"` parameter crops off the file extensions for the source files to create the file names for the HTML output files. If two files have the same name and different file extensions, whichever HTML file is generated last will overwrite previous ones. **Workaround or Solution:** Set the `useShortFileName` parameter to `"false"`. This setting generates HTML files with the file extensions included in the name. The drawback to this solution is that every link that points to the HTML output file needs to be revised, regardless of whether the files in question were affected by the bug.	9.0	Will not fix
CR276963	When you start the `medrec` or `samples` domains, you may see a warning message similar to this: `<Warning> <WorkManager> <BEA-002919> <Unable to find a WorkManager with name weblogic.wsee.mdb.DispatchPolicy. Dispatch policy weblogic.wsee.mdb.DispatchPolicy will map to the default WorkManager for the application bea_wls_async_response>` This warning message appears in the standard output of the Console while starting a WebLogic Server sample application with an asynchronous Web Service deployed. **Workaround or Solution:** The warning is harmless and can be ignored.	9.2	Will not fix

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR317037	The documentation for the WebLogic Server samples assumes that the default WebLogic Server installation directory is `weblogic100`; instead, the actual default WebLogic Server installation directory is `wlserver_10.0`.	10.0	10.3

HTTP PublishSubscribe Server

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR359872	The HTTP PublishSubscribe server does not support authentication and authorization of the local client. The local client has full permissions to operate on channels of the HTTP PublishSubscribe server, which means the local client can create/delete channels and publish/subscribe events from channels.	10.3	
CR364102	In a clustering environment, event messages published by a local client on a server can be received only by subscribed clients connected to the same server. These messages cannot be received by subscribed clients connected to other servers in the cluster.	10.3	
CR371454	Event messages published to a channel by a local client will not go through the Message Filters configured to that channel.	10.3	

Installation

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR307273	On installation, the installer may incorrectly report `Insufficient disk space`. This error is the result of using a version of the UNIX `df` tool that is incompatible with POSIX. **Workaround or Solution:** Use a version of `df` that is compatible with POSIX.	9.2	
CR344551	The `startWLS` script, which has been deprecated since release 8.1, has been removed in WLS 10.3.	10.3	
CR365263	On a Microsoft Vista system, the generic installer normally must be run with administrative permissions for all installation tasks to be successful. Administrative permissions are required if: • The product will be installed in a directory that requires special permission. • Start Menu items will be created for All Users. • Services such as Node Manager will be installed to start when the system boots. Note that starting with Vista, an administrative user will not have full administrative authority until they approve the request via the UAC prompt. **Workaround or Solution:** 1. Create an administrative command window by right-clicking on the **Command Prompt** menu item. 2. Select **Run as administrator**. 3. Respond to the UAC prompt. A new command window will open with full permissions. 4. Run the `java -jar ...` command as normal in this window.	10.3	

Java EE

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR347787	FastSwap may relax the access modifiers of fields and methods. Private and protected members may be made public at runtime. This changes the behavior of reflection and may affect reflection-based frameworks such as Struts.	10.3	Will not fix
CR372525	FastSwap does not support redefinition of the Entity bean and ejbClass (Session/MDB). Therefore, any updates to entity classes will cause redefinition errors. **Workaround or Solution:** After updating an entity class, redeploy the application.	10.3	

JDBC

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR272583	There is a performance degradation in the WebLogic Type 4 JDBC Driver for Oracle `getAsciiStream()` method call caused by fixes and enhancements to character set support in the driver. In general, the updated driver outperforms the previous version of the driver. **Workaround or Solution:** Use the `getCharacterStream()` method instead of `getAsciiStream()`.	9.1	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR292068	The utils.Schema -s option has been removed. The -s option was for the obsolete WebLogic OCI driver, which did not accept the server name in the URL. All supported drivers now take all needed properties in the URL, so the -s option is no longer needed. It was removed because setting the server name in both the -s option and the URL caused problems. **Workaround or Solution:** Using the -s option will result in a warning saying that the -s option is no longer functional. If your connection fails, make sure all DBMS properties are in the URL.	10.0	NA
CR366527	The shutdown of a pool also kills its asynchronous connection testing process. When the pool is restarted, the asynchronous testing job does not restart, and the DataSource can't detect database failures by test frequency until WLS is rebooted. This issue no longer happens, as asynchronous testing is always restarted when the pool is restarted.	10.0	10.3

JMS

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR270257	Using the JMS WLMessageProducer forward() API on a newly created message causes a JMSClientException. The JMS WLMessageProducer forward() API can only be used on unmodified received messages.	9.2	
CR274706	Deployment descriptor validation fails when descriptor validation is enabled, and an EAR file contains only JMS modules. **Workaround or Solution:** Make sure that there is at least one J2EE specification-compliant module in the EAR.	9.2	Will not fix

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR277493	When multiple JMS producers use the same JMS Client SAF instance (within a single JVM), depending on the timing of the JMS SAF client creation, you might receive the following exception: `Error getting GXA resource [Root exception is weblogic.jms.common.JMSException: weblogic.messaging.kernel.KernelException: Error getting GXA resource]` **Workaround or solution:** When using multiple JMS SAF client producers, try introducing a small delay between the creation of each new client.	9.2	Will not fix
CR278644	Reconnecting `transactedSession.commit()` or `transactedSession.rollback()`calls may throw exceptions more than once when reconnecting. When a transacted session allows reconnect (that is, when a `WLConnection.getReconnectPolicy()` is not none), the first call to `session.commit()` or `session.rollback()` will throw an exception after the server `WLConnection` has reconnected after a failure. This is expected behavior. Unfortunately, the second call to `commit()`or `rollback()` may also throw an exception. Afterwards the reconnected transacted session will work normally.	9.2	10.3
CR278960	There is no support for multi-byte characters in WebLogic Store file and directory names. For instance, when the WebLogic Server name has multi-byte characters, the default store cannot be created, and the WebLogic Server will not boot. **Workaround or solution:** Create WebLogic Servers without multi-byte characters in the path name and use that path name rather than the default store. Do not use multi-byte characters in the Weblogic Server name.	9.2	Will not fix
CR280041	When the `WLConnection.getReconnectPolicy()` method is set to `all`, JMS `CLIENT_ACKNOWLEDGE` sessions may sometimes throw an extra `LostServerException` when `session.acknowledge()` is called. This occurs after the JMS session has been reconnected to the server.	9.2	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR280150	When the `WLConnection.getReconnectPolicy()` method is set to `all`, JMS `CLIENT_ACKNOWLEDGE` sessions do not always throw a `LostServerException` when `session.acknowledge()` is called and there are unacknowledged, non-persistent topic messages.	9.2	10.3
CR309020	Certain JMS applications could cause a memory leak on a WebLogic server if periods (that is, dots) or slashes are present inside Connection IDs or Subscriber IDs. This issue typically occurs only for applications that both (a) continuously create and destroy durable subscriptions on topic destinations, and (b) specify a unique string prior to the last '.' or '/' in the Connection ID or Subscriber ID for each new durable subscription, instead of reusing strings from past destroyed subscriptions. **Workaround or Solution:** Use caution when specifying a period '.' or slash '/' in a Connection ID and Subscriber ID.	10.0	
CR332252	When a FILE based persistent store is targeted to a MigratableTarget, it must have an explicit store `<directory>` parameter defined. This configuration requirement was not validated in prior releases and resulted in messages being stored in a private store directory that cannot be migrated. In WLS 10.3, if there are FILE stores defined in the domain that are targeted to a MigratableTarget, and no explicit `<directory>` parameter is defined, the server will boot with an appropriate error message in the server log.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR367126	A C program that uses the JMS C client library may crash with a JVM failure. This could be related to a known intermittent race-condition that is only known to occur with certain JVM products, where the likelihood of failure can change based on the JVM version and patch level, operating system, and hardware. Specifically, the JMS C-Client library implicitly attaches C-threads to the JVM, but fails to detach them when it is done with them. **Workaround or Solution:** The workarounds are: 1. Add code in the client to detach the JVM from any C thread that exits and that has previously called into the JMS C-API. 2. Do not allow any C thread that has previously called into the JMS C-API to exit before the entire process exits. Sun 1.5 and later can specifically handle this problem, although it is still recommended to call detach even with the Sun JVM. For more information, see: http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6219874 3. Upgrade to a newer JVM. Version 1.5 and later of the Sun JVM, and version R27.6 of the JRockit JVM do not have this problem, although it is still recommended to call detach even with the updated JVMs. For more information about this issue with the Sun JVM, see http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6219874	10.3	
CR368312	In WLS 9.0 or greater, setting the JMS quota parameters values `bytes-maximum` and `messages-maximum` to `-1` causes a failure with a validation error, instead of disabling the quota limitation. In WLS 10.3, when quota parameters `bytes-maximum` and `messages-maximum` are set to `-1`, the quota limitation will be disabled.	9.1	10.3

JNDI

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR264754	JMS message consumers will not always reconnect after a service migration when an application's `WLConnection.getReconnectPolicy()` attribute is set to `all`. If the consumers do not get migrated, either an exception is thrown or `onException` will occur to inform the application that the consumer is no longer valid. **Workaround or Solution:** The application can refresh the consumer either in the exception handler or through `onException`.	9.2	Will not fix

JSP and Servlet

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR276071	In prior releases, access logging could not be configured at the web application level. This created some overhead for empty web services calls. In WLS 10.3, the `access-logging-disabled` property has been added to the `container-descriptor` configuration in `weblogic.xml`. By default, access logging for web applications is enabled. To disable access logging for web applications, set `access-logging-disabled` to `true`.	9.2	10.3
CR306960	Previous releases of WLS's JSP compiler used the Javelin java compiler to generate a servlet class. The JSP compiler in WLS 10.3 uses Eclipse's JDT Java compiler to generate a servlet class. Because of the different Java compiler, diagnostic content for errors from JSP's Java scriptlets in WLS 10.3 could be different from previous releases.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR322017	JavaEE requires that if the service method of a servlet returns with an active transaction on the thread, the container should abort that transaction. However, for asynchronous servlets, suspending active transactions is not feasible because the thread doing the asynchronous work is not managed by the web container. Therefore, suspending any active transaction disables the ability of that thread to manage the transaction after the response has been completed. **Workaround or Solution:** For servlets which are of type `AbstractAsyncServlet` or `FutureResponseServlet`, the container will not abort any active transactions when completing the response. It is the responsibility of the asynchronous thread to do the proper cleanup.	10.0MP01	10.3
CR338873	Deployment plans cannot be used to override the following two descriptors during deployment of a web application or a web module: `WEB-INF/classes/META-INF/persistence.xml` and `WEB-INF/classes/META-INF/persistence-configuration.xml`. Deployment plans can otherwise be used to override any descriptor. **Workaround or Solution:** Package `WEB-INF/classes/META-INF/persistence.xml` and `WEB-INF/classes/META-INF/persistence-configuration.xml` (if present) along with related class files into a `jar` file. The `jar` file must then be placed in the `WEB-INF/lib` directory of the web application or web module. A deployment plan can be used to override the two descriptors in such a `jar` file.	10.3	
CR368064	With the Spring extension model enabled, WLS 10.3 does not support Spring Dependency Injection (DI) on JSP tag handlers for performance reasons. Currently, WLS supports Spring DI on most web components, for example, servlets, filters and listeners. Spring DI is not, however, presently supported on JSP tag handlers for performance reasons.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR368962	When a session is persistent and an old version of a servlet context is retired, accessing the application with a valid `sessionid` will cause a 503 error. For example, the session-persistent type of a versioned web application is 'file'. A user can access the application successfully. Later, version 2 of the application is redeployed and version 1 is retired. If the same user accesses the application, they will get a 503 error.	10.3	
CR376251	A NoClassDefFoundError may happen when accessing a tag (from a jsp) which was precompiled on an earlier version of WLS and placed under the WEB-INF/classes directory of the web application. **Workaround or Solution:** Recompile the tag classes and place them under WEB-INF/classes.	10.3	

JTA

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
366855	When a `RollbackException` is thrown, no root cause of the rollback is shown. In WLS 10.3, the root cause of the rollback is printed when throwing the standard exception stack trace.	10.3	10.3

Java Virtual Machine (JVM)

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR232043	Due to a known Sun Microsystems VM bug (513552), a 1.4 Thin Client Applet cannot contact WebLogic Server 9.0 or later. This is because the VM does not distinguish correctly between a client and a server connection. The VM creates a server-type connection and caches it. It then attempts to make a client-type connection, finds the cached connection and tries to use that, but then encounters an error because clients are not allowed to use server connections. **Workaround or Solution:** None. This issue must be resolved by Sun Microsystems.	9.0	Vendor fix required
CR305879	Some web applications may experience a performance degradation compared to WebLogic Server 9.2 when running WebLogic Server on JRockit version R26.4 using the default GC algorithm on multi-CPU/core machines. **Workaround or Solution:** Turning on the JVM flag -Xgc:parallel may fix this problem.	10.0	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR363053	Applications that run on RH Linux on Intel G5 processors and that also directly or indirectly use system time calls may experience intermittent time issues if the `ClockSource` is set to `tsc` (the default). The standard POSIX C `gettimeofday()` call, and consequently also the Java `System.currentTimeMillis()` and `java.util.Date()` calls can intermittently return a value that is approximately 4400 seconds in the future, even in a single-threaded application.	10.3	Vendor fix required

This issue is not unique to WebLogic or Java, but applies to any application running on RH Linux on Intel G5 processors. Issues can occur for applications that either explicitly make a time call using standard Java, or explicitly by using any time-based application server services.

Possible symptoms include, but are not limited to, premature transaction timeouts, unexpected expiration of JMS messages, and incorrectly scheduled timers.

If you're interested in a standalone reproducer for this problem, contact Oracle and reference CR363053.

Workaround or Solution:

There's no known official patch for Linux. Instead, change the clock source from `tsc` to `hpet`. After making this modification on test systems, exceptions due to invalid `System.currentTimeMillis()/gettimeofday()` return values were no longer seen. To change the system clock from `tsc` to `hpet` on a trial basis, perform the following steps as `root`:

1. disable ntpd (if running).

2. echo 'hpet' > /sys/devices/system/clocksource/clocksource0/current_clocksource

3. enable ntpd

Note that this change will not survive a reboot. For more information, please see:
http://www.gossamer-threads.com/lists/linux/kernel/813344

Monitoring

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR285328	`ServerLifeCycleRuntimeMBean` is not available to the WLDF harvester. The WLDF harvester looks only at runtime MBeans in the local runtime MBean server. On the Administration Server, the `ServerLifeCycleRuntimeMBean` instances are registered only in the federated MBean server and are not available in the local runtime MBean server. Thus, it is not possible to harvest its attributes or configure watch rules based on their attributes, such as the State attribute. **Workaround or Solution:** It is possible to configure an SNMP monitor on the attributes of `ServerLifeCycleRuntimeMBean` to receive notifications based on their values.	10.0	10.3
CR299918	Configuration of `SNMPJMXMonitorMBean` and `SNMPAttributeChangeMBean` is now validated. For the `SNMPJMXMonitorMBean`, the `MonitoredMBeanType` must be a valid WLS MBean type and the `MonitoredAttributeName` must be a valid attribute of that MBean type. Similar validation is added for the `SNMPAttributeChangeMBean`'s `AttributeMBeanType` and `AttributeName` attributes. Invalid values will cause errors. **Workaround or solution:** Use valid attribute values.	10.0	
CR300617	Because WebLogic Server has changed its SNMP implementation, the `DebugLevel` attribute on the `SNMPAgentMBean` has been removed without deprecation. **Workaround or solution:** If you need debug information for an SNMP agent, use the `DebugSNMPToolkit` attribute on `ServerDebugMBean`.	10.0	NA

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR300681	The `TargetedTrapDestinations` attribute on the `SNMPAgentMBean` interface is now obsolete. **Workaround or solution:** Use the `SNMPTrapDestinations` attribute in place of the `TargetedTrapDestinations` attribute.	10.0	NA
CR300893	The `-C <prop-file>` option of the `SnmpTrapLogger` sub-command of the SNMP command line tool, `weblogic.diagnostics.snmp.cmdline.Manager`, does not work. As a result, you cannot specify a log config properties file with the `SnmpTrapLogger` sub-command of the SNMP command line tool. **Workaround or solution:** None at this time.	10.0	
CR310492	The `@unharvestable` tag is not being honored at the interface level. If MBean attributes are not explicitly marked as `@unharvestable`, they are considered to be harvestable and will appear as harvestable in the WebLogic Administration Console. **Workaround or Solution:** You can explicitly mark MBean attributes as `@unharvestable`.	10.0	
CR311787	When WebLogic Server sends SNMP INFORM notifications, if the receiving manager fails to acknowledge, retries are attempted in a separate thread. Control is immediately returned to the caller with success status. Thereafter, if the manager fails to respond, the failure is not reported. **Workaround or Solution:** There is not yet a solution for this issue. The failure can be observed if the DebugSNMPToolkit debug is enabled. However, this will produce very verbose output.	10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR312358	The `Servlet_Before_Service`, `Servlet_After_Service` and `Servlet_Around_Service` diagnostic instrumentation monitors fail to weave diagnostic code into a servlet's `doGet` or `doPost` methods, if those methods have protected access. The `doGet` and `doPost` methods do get properly instrumented if they have public access qualifiers. **Workaround or Solution:** Consider making the `doGet` and `doPost` methods in your servlet public.	10.0	10.3
CR321077	The commons logging API jar `com.bea.core.apache.commons.logging.api_xxx.jar` was on the system classpath via the manifest classpath. This could cause issues if you wanted to use a different version of commons. In WLS 10.3, classes in the commons API jar are no longer on the system classpath (by default). These classes are available through `weblogic.server.modules.extra_10.2.0.0.jar`. Therefore, if necessary, you can add this jar to the system classpath.	10.0	10.3

CR326098	Server upgraded from 9.2 to 10.0 fails to start with the following error:
	
	`<Server subsystem failed. Reason:`
	`java.lang.NoClassDefFound`
	`Error: weblogic/logging/DateFormatter`
	`java.lang.NoClassDefFoundError:`
	`weblogic/logging/DateFormatter`
	`        at`
	`weblogic.logging.log4j.ConsoleLayout.<init>(Con`
	`soleLayout.java:18)`
	
	Workaround or Solution:
	
	When upgrading WLS to a newer version, copies of `wllog4j.jar` in use should be replaced with the new version from the `server/lib` install directory.
		10.0	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR361085	In prior releases, the pointcut for diagnostic monitors `JDBC_Before_Connection_Internal` and `JDBC_After_Connection_Internal` was very verbose, resulting in a lot of noise in generated data. In WLS 10.3, these monitors generate data only while executing the `weblogic.jdbc.jta.DataSource::getConnection()` and `weblogic.jdbc.jts.Driver::connect()` methods.	10.3	10.3
CR362287	When production mode is enabled/disabled with the `-Dweblogic.ProductionModeEnabled=true|false` command line switch, values for some configuration parameters (for example, the **Files To Retain** log-rotation parameter) shown in the Console may not reflect the actual values used at run time. This affects any configuration parameter for which default values differ between production and development mode. This is because the Console shows values which do not reflect runtime overrides applied via the command line. In WLS 10.3, the server emits warning messages in the server log when it detects that the effective production or development mode differs from what is specified in the `config.xml` file (message-ids: 141274 and 141275 respectively).	10.3	
CR367260	In WLS 10.0, a message indicating the number of data retirement tasks that were scheduled was periodically logged. This message was logged even when 0 tasks were scheduled, causing unnecessary noise in the log file. The data retirement log messages are now logged only when at least one data retirement task is scheduled.	10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR367280	The name-space and schema-location for the diagnostics descriptor `weblogic-diagnostics.xml` has changed from:
	
	`targetNamespace='http://www.bea.com/ns/weblogic /90/diagnostics'`
	
	`schemalocation='http://www.bea.com/ns/weblogic/ 920/diagnostics.xsd'`
	
	to:
	
	`targetNamespace='http://www.bea.com/ns/weblogic /weblogic-diagnostics'`
	
	`schemalocation='http://www.bea.com/ns/weblogic/ weblogic-diagnostics/1.1/weblogic-diagnostics.x sd'`
	
	However, old descriptors will continue to work. It is recommended that new descriptors should follow the new scheme.
		10.3	

CR367304	WLST uses Jython. At startup, Jython scans the `jar` files and stores information in a cache directory. By default, this is the `{java.io.tmpdir}/wlstTemp` directory. WLST checks that the user can write to the `java.io.tmpdir` directory, but does not check that the user can write to the `wlstTemp` subdirectory. If the `wlstTemp` directory is owned by another user, the user gets write errors from the Jython caching code. In WLS 10.3, WLST will utilize a user-specific cache directory if the user cannot write to the common `wlstTemp` directory. This allows WLST to start without generating cache write errors.	10.3	10.3
CR370495	In WLS 10.0, `log4j.jar` was implicitly included on the WebLogic Server classpath via the manifest classpath. This was unintentional, and could interfere with the ability to use a different version of `log4j`. In WLS 10.3, `log4j.jar` is no longer on the WebLogic Server classpath. If you rely on the fact that `log4j.jar` is on the WebLogic Server classpath, you must manually put it in the classpath of the WebLogic Server process.	10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR371740	Previous releases of WLS used to ship the instrumentation library configuration as a serialized object available as a resource. WLS 10.0 ships it as an XML descriptor. It is converted to a serialized object and cached on first use. The serialized object is stored as the following file: `$DOMAIN/servers/$SERVER/cache/diagnostics/Instr umentationEngineConfig.ser` where `$DOMAIN` is the domain directory and `$SERVER` is the name of the managed server. Instrumentation engine configuration is now shipped as an XML file. It is converted to a serialized object and cached as a serialized object on first use.	10.3	10.3
CR377794	When a Harvester watch rule variable resolves to multiple metric data points triggers, the WatchData field in the outgoing notification would be empty; normally this is supposed to contain a list of the MBean instances and the actual values of the attributes used to evaluate the rule.	10.3	

Node Manager

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR255457	The following NodeManager MBean APIs and properties were removed in WebLogic Server 9.x, but not deprecated: DEFAULT_CERTIFICATE_PASSWORD_BYTE_ARRAY getCertificatePasswordEncrypted() getCertificate() getCertificatePassword() getCertificateType() getTrustedCertsFile() setCertificate(java.lang.String) setCertificatePassword(java.lang.String) setCertificatePasswordEncrypted(byte[]) setCertificateType(java.lang.String) setSSLEnabled(boolean) setTrustedCertsFile(java.lang.String)	9.0	NA

Operations, Administration, and Management

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR255460	The following ServerMBean APIs and properties were removed in WebLogic Server 9.x, but not deprecated:
	
	isIORServletEnabled();
	setHelpPageURL(java.lang.String);
	setIORServletEnabled(boolean);
	setWorkspaceShowUserKeysOnly(boolean);
		9.0	NA

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR309553	JRockitRuntimeMBean exposes functionality from the JRockit MAPI ProfilingSystem API, which is being deprecated. Consequently, the following JRockitRuntimeMBean methods are deprecated:
	
	 public boolean isExceptionCountEnabled(Class throwableClass);
	 public long getMethodTiming(Method method);
	 public long getConstructorTiming(Constructor constructor);
	 public boolean isMethodInvocationCountEnabled(Method method);
	 public boolean isConstructorInvocationCountEnabled(Constructor constructor);
	 public boolean isMethodTimingEnabled(Method method);
	 public boolean isConstructorTimingEnabled(Constructor constructor);
	 public long getExceptionCount(Class throwableClass);
	 public long getMethodInvocationCount(Method method);
	 public long getConstructorInvocationCount(Constructor constructor);
		10.0	NA

CR348043	Prior to WLS 10.3, the isSet and unSet methods were not available for Security MBeans because the static BeanInfos did not contain the appropriate methods. In WLS 10.3, isSet and unSet information has been added to the static BeanInfo for the Security MBeans, and you can call isSet and unSet on Security MBeans.	10.3	10.3
CR351225	In prior releases, Node Manager failed to start because the -d64 flag was not specified when WLST created the Node Manager process. In WLS 10.3, if the data architecture is 64-bit and the vendor is Sun or HP, WLST will pass -d64 when starting Node Manager, and will successfully start the Node manager on 64-bit Sun or HP platforms.	10.3	10.3
CR363262	In prior releases, when logged in as a user with the Deployer role, trying to access the **AdminServer > Monitoring > Default Store** tab caused an exception in the administration server log. In WLS 10.3, no exception will be logged for users with the Deployer role.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR364818	In prior releases, if a JVM is hung, the weblogic.Admin PING command never times out. In WLS 10.3, the weblogic.Admin PING command will time out if WLS server does not respond.	10.3	10.3

Protocols

No issues at this time.

RMI-IIOP

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR371491	In prior releases, when the remote implementation class implemented more than one Remote interface and network downloading was enabled, the implementation class would be downloaded on the client when using IIOP protocol with the full client jar. **Workaround or Solution:** In WLS 10.3, the implementation class is not downloaded to the client in this situation.	10.3	10.3

Security

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR231321	The option -Dweblogic.system.StoreBootIdentity only works if the appropriate server security directory exists. This directory is usually created by the Configuration Wizard or upgrade tool. However, the appropriate server security directory could be absent in domains checked into source-control systems.	9.0	Will not fix
CR346913 CR346067	WLS allows for a NULL cipher to be used with an SSL connection, which results in data not being encrypted. In WLS 10.3, the use of the NULL cipher is now disabled by default. In order for a client to enable the NULL cipher, set the weblogic.ssl.AllowUnencryptedNullCipher system property to true. For example: -Dweblogic.ssl.AllowUnencryptedNullCipher=true In WLS 10.3, client SSL connections requiring a NULL cipher will fail unless this system property explicitly enables the use of the NULL cipher.For NULL cipher to be used, you need to enable NULL cipher on both the server side and client side. If not enabled on both sides, the SSL handshake will fail.	10.3	
CR354701	Upgrade uses an XSLT script to transform a pre-9.0 config.xml file. This XSLT script moves attributes from the Security MBean to the SecurityConfiguration MBean. However, the read of the config file fails because it found 9.0 or higher attributes in a pre-9.0 config.xml. In WLS 10.3, the ConnectionFilter, ConnectionFilterRules, and ConnectionLoggerEnabled attributes were added to the exclude list and do not generate errors if they are present in a transformed pre-9.0 config.xml. With the implemented fix, WLS 8.1 or prior domains that contain ConnectionFilter, ConnectionFilterRules, or ConnectionLoggerEnabled attributes on the SecurityMBean can be successfully upgraded to WLS 10.3.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR356527	After upgrading a domain from WLS 6.1, the WLS Server instance will not boot due to an authentication failure. **Workaround or Solution:** A system user password must be set up in the WLS 6.1 domain before or after the upgrade process in order for the WLS Server instance to boot properly.	10.3	
CR361984	A WLS Sever instance can experience a boot time failure with a SecurityServiceException when the RDBMS Security Data Store is configured for a DB2 database using the WLS-supplied DB2 driver. **Workaround or Solution:** When RDBMS Security Data Store is using the AlternateId connection property for a DB2 database, you must also set the additional property BatchPerformanceWorkaround as true when using the WLS-supplied DB2 driver.	10.3	
CR372329	After you configure either the Identity Provider or Service Provider services for SAML 2.0 and attempt to publish the SAML 2.0 services metadata file, an InvalidParameterException message may be generated and displayed in the Administration Console. **Workaround or Solution:** When configuring the SAML 2.0 federation services for a WebLogic Server instance, be sure to enable all binding types that are available for the SAML role being configured. For example, when configuring SAML 2.0 Identity Provider services, you should enable the POST, Redirect, and Artifact bindings. When configuring SAML 2.0 Service Provider services, enable the POST and Artifact bindings. Optionally, you may choose a preferred binding.	10.3	

Spring Framework on WebLogic Server

The 2.0.2 version of the Spring Framework is certified for WLS 10.3.

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR319968	The OpenJPA ClassFileTranformer doesn't work when running WebLogic Server on JRockit. **Workaround or Solution**: Use an alternative method of applying enhancements at build time through an OpenJPA enhancer compiler; do not use the LoadTimeWeaver.	10.0	
CR371212	For the SpringSource 'petclinic' sample, the `petclinic.war` deploys without any problem. The `petclinic.ear` will not deploy on WLS because it is not packaged correctly. A request has been sent to SpringSource to fix the `petclinic.ear` packaging.	10.3	

Upgrade

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR252571	`EJBException` occurs when you upgrade the petstore domain from WebLogic Server 7.0.x to 9.x. **Workaround or Solution:** Prior to upgrading the domain, edit the `config.xml` file to ensure that the `xa-connection-factory-enabled` element is set to true for the `MyTopicConnectionFactory`.	9.1	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR256082	The JWS in WebLogic Server 9.2 and later does not support the callback protocol in WebLogic Workshop 8.1. Therefore, a 9.2 or later JWS cannot callback to an 8.1 WebLogic Workshop application or an 8.1 WLI JPD. Also, a 9.2 or later JWS client cannot receive callbacks from an 8.1 WLW application or an 8.1 WLI JPD. **Workaround or Solution:** Front-end the client side call with a 9.2 Service Control.	9.2	
CR265099, CR268712	Weblogic Workshop 8.1 supported returning the `java.util.Map` type from Web Service operations. The runtime provided a Workshop specific serialization of the Map to and from XML. The schema for that serialization was included in the WSDL for the Web Service. In WebLogic Server 9.2 and later, the `java.util.Map` type can no longer be returned from Web Service operations. **Workaround or Solution:** Provide an application-defined type that supports the `key/value` features provided by `java.util.Map`. That type must conform to JAX/RPC Java<->XML Serialization rules. If the application-type is going to contain subclasses of type `key` or value `type`, then the `weblogic.jws.Types` annotation must be used to specify the types that could be contained at runtime. WebLogic Web Services (and their clients) that previously returned a `java.util.Map` will have to be manually updated to use this new application-defined type.	9.2	

Web Applications

No issues at this time.

Web Logic Scripting Tool (WLST)

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR280234	The WLST `loadProperties` command does not support loading a property with a name that contains "`.`" characters. For example, if the property `myapp.db.default` is present in the property file, WLST throws a name exception: ```\nProblem invoking WLST - Traceback (innermost last):\n File "<iostream>", line 7, in ?\n File "<iostream>", line 4, in readCustomProperty\n NameError: myapp\n``` This is a system limitation of Python and the `loadProperties` command. WLST reads the variable names and values and sets them as variables in the Python interpreter. The Python interpreter uses "`.`" as a delimiter to indicate module scoping for the namespace and/or package naming. Therefore, the properties file fails because `myapp.db.default.version=9i` is expected to be in the `myapp.db.default` package. This package does not exist.	9.0	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR280234 (cont'd)	**Workarounds:**
	
	Use variable names that do not have periods. This will allow you to load the variables from the property file and refer to them in WLST scripts. You could use another character such as "_" or lowercase/uppercase character to delimit the namespace.
	
	As an alternative, you can set variables from a properties files. When you use the variables in your script, during execution, the variables are replaced with the actual values from the properties file. For example:
	
	`myapp.py`
	`var1=10`
	`var2=20`
	
	`import myapp`
	`print myapp.var1`
	`10`
	
	`print myapp.var2`
	`20`
	
	This will work for one level of namespaces (`myapp.var1`, `myapp.var2`). It will not work for top level variables that share the same name as the namespace (for example, `myapp=oracle and myapp.var1=10`). Setting the `myapp` variable will override the `myapp` namespace.		

CR280234 (cont'd)	If you need multiple levels, then you can define a package namespace using directories. Create a `myapp/db/default` directory with a `vars.py` file as follows:
	
	`var1=10`
	`var2=20`
	
	Then import:
	
	`import myapp.db.default.vars`
	`print myapp.db.default.vars.var1`
	`10`
	
	You may need to add `__init__.py` files to the subdirectories. Refer to the Python documentation for more information on packages:
	
	http://docs.python.org/tut/node8.html		

Web Server Plug-Ins

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR306253	The ISAPI plug-in always looks for the `iisproxy.ini` file under the same directory in which the `iisproxy.dll` file resides. Therefore, if you wish to use SSL, you must rename `iisproxy128.dll` to `iisproxy.dll` and reconfigure IIS after renaming the file.	9.2	
CR347731	Apache versions built with the special flags `-D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64` are not compatible with the Apache Web Server plug-ins included with previous WLS releases. **Workaround or Solution:** In WLS 10.3, WLS includes special plug-ins that have been compiled with these flags, for the sol_sparc, linux_i686, and sol_x86 platforms.	10.3	10.3
N/A	The WLS 10.3 installation does not include the Apache HTTP server plug-ins. For instructions on how to download and install them, see "Installing the Apache HTTP Server Plug-In" in *Using WebLogic Server Plug-ins with WebLogic Server*.	10.3	

Web Services and XML

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR189158	WebLogic Server does not support Sparse Arrays and Partially Transmitted Arrays as required by the JAX-RPC 1.1 Spec.	9.0	Will not fix
CR228385	The Web Service Description Language (WSDL) compiler does not generate serializable data types, so data cannot be passed to remote EJBs or stored in a JMS destination.	9.0	Will not fix

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR258245	The Web Services default `JdkSSLAdapter` class directly instantiates the Sun HTTPS protocol handler (`sun.net.www.protocol.https.Handler`). In IBM JVMs this class does not exist, causing a `noClassDefFoundError` whenever the `JdkSSLAdapter` was used. In WLS 10.3, `JdkSSLAdapter` works properly when executing in IBM JVM. **Workaround or Solution:** In WLS 10.3, it is determined if the Sun class name is present by calling `Class.forName()`. If the class is successfully loaded and a Class object is returned, it is used to instantiate the Sun protocol handler, which is then used as before. If the class is not loaded, the same is attempted using the name of the equivalent IBM class (`com.ibm.net.ssl.ww2.protocol.https.Handler`). In the unlikely event that neither class can be loaded, an `IOException` is thrown and the operation fails.	9.1	10.3
CR260810	WebLogic Server does not support using a custom exception on a callback that has a package that does not match the target namespace of the parent Web Service. **Workaround or Solution:** Make sure that any custom exceptions that are used in callbacks are in a package that matches the target namespace of the parent web service.	9.2	Will not fix
CR262036	You cannot use JMS transport in an environment that also uses a proxy server. This is because, in the case of JMS transport, the Web Service client always uses the `t3` protocol to connect to the Web Service, and proxy servers accept only HTTP/HTTPS.	9.2	Will not fix
CR265983	Clientgen fails when processing a WSDL that uses the complex type `http://www.w3.org/2001/XMLSchema{schema}` as a Web Service parameter. **Workaround or Solution:** There is no known workaround.	9.2	Will not fix

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR266759	Web Services that define a Callback interface with a mixed-case package name will fail to compile with `jwsc`. **Workaround or Solution:** Create the Callback interface in a package with a lower-case name.	9.2	10.3
CR269696	WebLogic Server 9.2 and later does not support JAX RPC handlers in callback Web Services. **Workaround or Solution:** If JAX RPC handlers were used with Web Services created with WebLogic Workshop 8.1, then such applications must be redesigned so that they do not use callback handler functionality.	9.2	Will not fix
CR273307	WebLogic Server 9.2 and later does not support message-level security in callback Web Services. **Workaround or Solution:** Web Services created with WebLogic Workshop 8.1 that used WS-Security must be redesigned to not use message-level security in callbacks.	9.2	Will not fix

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR273446	WebLogic Server does not support handling of Java method arguments or return parameters that are JAX-RPC-style JavaBeans that contain an XmlBean property. For example, applications cannot have a method with a signature like this:
	`void myMethod(myJavaBean bean);`
	
	where myJavaBean class is like:
	
	```\npublic class MyJavaBean {\n private String stringProperty;\n private XmlObject xmlObjectProperty;\n public MyJavaBean() {}\n String getStringProperty() {\n return stringProperty;\n }\n void setStringProperty(String s) {\n stringProperty = s;\n }\n XmlObject getXmlObjectProperty() {\n return xmlObjectProperty;\n }\n void getXmlObjectProperty(XmlObject x) {\n xmlObjectProperty = x;\n }\n}\n```
	
	Workaround or Solution:
	
	There is no known workaround.
		9.2	Will not fix

CR280155	WebLogic Server 9.2 and later does not support the use of double-byte characters in URLs in Web Services. **Workaround or Solution:** Web Services created with WebLogic Workshop 8.1 that used double-byte characters in URLs must be redesigned to remove all double-byte characters from URLs.	9.2	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR282449	Using a two dimensional `XmlObject` parameter (`XmlObject[][]`) in a JWS callback produces an `IllegalArgumentException`.	9.2	Will not fix
	Workaround or Solution:		
	There is no known workaround.		
CR282728	Using `SoapElement[]` as a Web Service parameter with `@WildcardBinding(className="javax.xml.soap.SOAP Element[]", binding=WildcardParticle.ANYTYPE)` will always result in an empty array on the client.	9.2	y
	Workaround or Solution:		
	Do not use the `@WildcardBinding` annotation to change the default binding of `SOAPElement[]` to `WildcardParticle.ANYTYPE`. The `SOAPElement[]` default binding is set to `WildcardParticle.ANY`.		
CR293674	In certain circumstances, the invocation of a WebLogic Web Service configured for reliable messaging will throw the following `PersistentStoreRuntimeException` error: `weblogic.store.PersistentStoreRuntimeException: [Store:280051]The persistent store was not able to create a new record.`	10.0	10.3
	Workaround or Solution:		
	When starting the WebLogic Server instance to which the reliable Web Service is deployed, set the following property: `-Dweblogic.wsee.exclude.properties=weblogic.wse e.ejb.target`		

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR297653	The WLS web service client generated from the Google `wsdl` can not call the Google service. A SOAP Fault is returned from calling. **Workaround or Solution:** The Google services require the elements in the request SOAP to define the `xsi:type` attribute. You can set the `WLStub.MARSHAL_FORCE_INCLUDE_XSI_TYPE` property to `'true'` for the client stub. Here is an example code snippet: ```\nimport weblogic.wsee.jaxrpc.WLStub;\nimport javax.xml.rpc.Stub;\n...\n GoogleSearchService service = new\nGoogleSearchService_Impl();\n GoogleSearchPort port =\nservice.getGoogleSearchPort();\n\n //**set the property**//\n ((Stub)port)._setProperty(WLStub.MARSHAL_\nFORCE_INCLUDE_XSI_TYPE,'true');\n\n googlesearch.GoogleSearchResult result =\nport.doGoogleSearch(\n 'CyaccPpQFHKPBgXIqSuExba5IyHTMBT/',\n 'spanish empire',0,1,true,'',\ntrue,'lang_en','UTF-8','UTF-8');\n\n ...\n```	10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR309493	When Web Service A wants to invoke Web Service B, Web Service A should use the `@ServiceClient` annotation to do this. If Web Service B needs a custom policy file that is not attached to Web Service B's WSDL, then Web Service A will fail to run. Web Service A will look for the policy file at `/Web-Inf/classes/policies/xxx.xml`. Since no policy file exists at that location, WebLogic Server will throw a file not found exception. **Workaround or Solution:** Attach the custom policy file to Web Service B, as in this example: `@Policy(uri="CustomPolicy.xml",         attachToWsdl=true) public class B {   ... }`	10.0	Will not fix
CR313836	In prior WLS releases, when a WLS Client was talking to a .NET Service using WS-SecureConversation to secure messages, WLS Client invocations would sometimes fail on the .NET Server with a general error message. This does not occur with WLS 10.3.	10.3	10.3
CR314559	The WebMethod name of a JAX-WS Web Service cannot include non-ASCII or multi-byte characters. This is due to an i18n bug in the `com.sun.tools.javac.main.JavaCompiler` class. **Workaround or Solution:** Set the WebMethod name to a value that includes only ASCII characters.	10.0	10.3
CR314660	In prior releases, a WLS Client using WS-SecureConversation to secure message exchanges could have messages rejected, with the following message displayed, due to date format problems: `System.FormatException: String was not recognized as a valid DateTime.` In WLS 10.3, this .NET exception no longer occurs for Client applications.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR321695	If you turn on the XMLRegistry service on Weblogic Server, a cache service is enabled which can cache the DTD/Schema file content after an XML file has been validated. If another XML file using the same DTD/Schema is validated, the DTD/Schema file content will be retrieved from the cache service. If the actual DTD/Schema file content is changed before the cached content expires, the cache service doesn't know to update the cached content. **Workaround or Solution:** You can set `<handle-entity-invalidation>` in `config.xml` to indicate that the cached DTD/Schema content will be invalidated when an XML file has failed to be validated. By doing so, the cache service will update the cached content and the XML file will be validated automatically again. If the validation always fails, then the validation fault is thrown. You must manually configure `config.xml` to set `<handle-entity-invalidation>`. The following examples show how to do this for a single entity or for a multiple entity. Example 1. To enable handling cache invalidation for a single entity (also applicable to the SAX Parser):	10.0	10.3

```
<xml-registry>
  <name>dtdCache</name>
  <document-builder-factory>
com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderFactoryI
mpl </document-builder-factory>
  <xml-entity-spec-registry-entry>
    <name>deliveryResponse</name>
    <public-id>DELIVERY</public-id>
    <system-id>http://jasemulator:7011/Emulator/dtd/delivery.dtd
</system-id>
  <entity-uri>http://jasemulator:7011/Emulator/dtd/delivery.dtd
    </entity-uri>
    <when-to-cache>cache-on-reference</when-to-cache>
    <cache-timeout-interval>60000</cache-timeout-interval>
    <handle-entity-invalidation>true</handle-entity-invalidation>
  </xml-entity-spec-registry-entry>
</xml-registry>
```

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR321695 (cont.)	Example 2. To enable handling cache invalidation for multiple entities (also applicable to the SAX Parser):
	
	`<xml-registry>`
	`  <name>dtdCache</name>`
	`  <document-builder-factory>`
	`com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderFactoryImpl</document-builder-factory>`
	`  <handle-entity-invalidation>true</handle-entity-invalidation>`
	`  <xml-entity-spec-registry-entry>`
	`   <name>deliveryResponse</name>`
	`   <public-id>DELIVERY</public-id>`
	`   <system-id>http://jasemulator:7011/Emulator/dtd/delivery.dtd`
	`   </system-id>`
	`   <entity-uri>http://jasemulator:7011/Emulator/dtd/delivery.dtd`
	`   </entity-uri>`
	`  </xml-entity-spec-registry-entry>`
	`  <xml-entity-spec-registry-entry>`
	`   <name>purchaseOrder</name>`
	`   <public-id>PO</public-id>`
	`   <system-id>http://jasemulator:7001/Emulator/dtd/purchase.dtd`
	`   </system-id>`
	`   <entity-uri>http://jasemulator:7001/Emulator/dtd/purchase.dtd`
	`   </entity-uri>`
	`  </xml-entity-spec-registry-entry>`
	`</xml-registry>`		

CR329752	`wsdlc` for JAX-WS does not use external binding customizations for generating a correct service implement template file. In WLS 10.3, the name of the implement template file has been changed, as well as the `srcServiceName` and `srcPortName` options in the wsdlc task. For details, please see the documentation for the `wsdlc` Ant task in the *WebLogic Web Services Reference*.	10.0	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR333308	The XML Schema 'particle valid restriction' rule is no longer enforced when generating or using XMLBeans classes generated from XML Schema in a WLS Web Service. A number of Web Services interfaces defined by industry-specific organizations (for example, Heath Level Seven at http://www.hl7.org) specify metadata that does not conform to the XML Schema particle valid restriction.
	
	In previous releases of WLS, such schema would fail to compile (for example, during execution of the `wsdlc` task or during deployment) with an error such as:
	
	`error: rcase-Recurse.2: Invalid Restriction. The following particles of the derived <sequence> cannot be mapped to the base <sequence>'s particles:`
	
	`<element name='thumbnail@urn:hl7-org:v3'>`
	
	In WLS 10.3, this restriction has been relaxed, allowing XMLBeans classes to be fully and correctly generated from such schema.
	
	Note: These schema and the Java classes generated from them are in all ways correct and will yield correct results when included in a Web Service.
		10.0MP01	10.3

CR335345	If you are not using the JDK6/JRockit6 shipped with WLS 10.3, an error may occur while invoking JavaDoc to inspect your source files. This may be due to the fact that `$JAVA_HOME/lib/tools.jar` does not seem to be in your system classloader. One common case in which this happens is when using the ant tool, which uses a special context classloader to load classes from `tools.jar`. **Workaround or Solution:** Sun has provided a fix for this at: http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=2158389 Alternatively, you can avoid this issue by simply including `$JAVA_HOME/lib/tools.jar` in the java `-classpath` parameter. If you are running ant, you will need to modify the standard ant script to include `tools.jar` in the `-classpath`.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR336924	When the security policy has one of these Token Assertions, the client side may fail to validate the signature on the server response message.	10.3	

```
<sp:WssX509PkiPathV1Token11/>
<sp:WssX509Pkcs7Token11/>
<sp:WssX509PkiPathV1Token10/>
<sp:WssX509Pkcs7Token10/>
```

In addition, when there are more than two certifications in the chain for X509 certification for `<sp:WssX509Pkcs7Token11/>` or `<sp:WssX509Pkcs7Token10/>` Token Assertion, the server side may fail to validate the signature on the incoming message.

A policy such as the following policy is not supported, unless the entire certificate chain remains on the client side.

```
<sp:AsymmetricBinding>
    <wsp:Policy>
      <sp:InitiatorToken>
         <wsp:Policy>
          <sp:X509Token
            sp:IncludeToken='. .
./IncludeToken/AlwaysToRecipient'>
             <wsp:Policy>
               <sp:WssX509Pkcs7Token11/>
             </wsp:Policy>
          </sp:X509Token>
        </wsp:Policy>
      </sp:InitiatorToken>
      <sp:RecipientToken>
        <wsp:Policy>
          <sp:X509Token
            sp:IncludeToken='. .
./IncludeToken/Never'>
             <wsp:Policy>
               <sp:WssX509Pkcs7Token11/>
             </wsp:Policy>
          </sp:X509Token>
        </wsp:Policy>
      </sp:RecipientToken>
  . . .
    </wsp:Policy>
  </sp:AsymmetricBinding>
```

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR336924 (cont.)	**Workaround or solution:** Use either of the following two solutions: 1. Configure the response with the `<sp:WssX509V3Token10/>` Token Assertion, instead of `WssX509PkiPathV1Token11/>`. The policy will look like this:	10.3	

```
 <sp:AsymmetricBinding>
    <wsp:Policy>
      <sp:InitiatorToken>
         <wsp:Policy>
          <sp:X509Token
             sp:IncludeToken='. .
./IncludeToken/AlwaysToRecipient'>
               <wsp:Policy>
                 WssX509PkiPathV1Token11/>
               </wsp:Policy>
             </sp:X509Token>
          </wsp:Policy>
       </sp:InitiatorToken>
       <sp:RecipientToken>
         <wsp:Policy>
           <sp:X509Token
             sp:IncludeToken='. .
./IncludeToken/Never'>
               <wsp:Policy>
                 <sp:WssX509V3Token10/>
               </wsp:Policy>
             </sp:X509Token>
          </wsp:Policy>
       </sp:RecipientToken>
. . .
      </wsp:Policy>
   </sp:AsymmetricBinding>
```

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR336924 (cont.)	2. Configure the response with the WssX509PkiPathV1Token11/> token assertion, but include it in the message. The policy will look like this:	10.3	

```
<sp:AsymmetricBinding>
   <wsp:Policy>
      <sp:InitiatorToken>
         <wsp:Policy>
          <sp:X509Token
            sp:IncludeToken='. .
./IncludeToken/AlwaysToRecipient'>
            <wsp:Policy>
               WssX509PkiPathV1Token11/>
            </wsp:Policy>
          </sp:X509Token>
         </wsp:Policy>
      </sp:InitiatorToken>
      <sp:RecipientToken>
         <wsp:Policy>
          <sp:X509Token
            sp:IncludeToken='. .
./IncludeToken/AlwaysToInitiator'>
            <wsp:Policy>
               WssX509PkiPathV1Token11/>
            </wsp:Policy>
          </sp:X509Token>
         </wsp:Policy>
      </sp:RecipientToken>
 . . .
      </wsp:Policy>
   </sp:AsymmetricBinding>
```

When there are multiple certifications in the X509 Certificate chain, WssX509PkiPathV1Token11/> or <sp:WssX509PkiPathV1Token10/> should be used, instead of <sp:WssX509Pkcs7Token11/> or <sp:WssX509Pkcs7Token10/>.

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR337064	WebLogic Web Services expects that each WebLogic Server domain will contain specific resources needed to support web services. Some domains, however, are not created with these resources. For example, creating a default WebLogic Server domain in the configuration wizard (without applying any other templates) will not create the needed Web Services resources. A domain that doesn't contain Web Services resources will still boot and operate correctly for non-webservice scenarios, and any Web Service scenario that doesn't involve asynchronous request/response. You will, however, see INFO messages in the server log indicating that async resources have not been configured and that the async response service for web services has not been completely deployed. **Workaround or Solution:** Web Services that use async request/response will *not* function properly in a domain that doesn't have Web Services resources configured in it. To configure these resources, there are two approaches: • Use the configuration wizard and apply the `wls_webservice.jar` template to your domain. • Manually configure these resources according to the rules given in the online documentation under domain configuration for web services. **Note:** The configuration wizard approach mentioned above is not advised for domains that already have JMS servers configured and that enable JMS resource 'default targeting' on JMS resources such as destinations. The wizard automatically creates additional JMS servers, and the default targeted resources can automatically appear on the newly created JMS servers, yielding, for example, distributed destinations that suddenly span many more JMS servers than intended.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR344175	Reliable Messaging cannot support sending async responses and acks using a client certificate. Web Services using async or Reliable Messaging will automatically use the server's SSL certificate when establishing a new connection (back from the receiving service to the sending service) for the purposes of sending async responses, acknowledgments, etc.	10.3	
CR344724	For the `xmlcatalog` element in `build.xml`, the location of an entity must be a file on the local file system. It cannot be a remote file (for example, `http:`) or a file in an archive (for example, `jar:`). **Workaround or Solution:** If necessary, define the remote element as an entity in a catalog file instead.	10.3	Will not fix
CR347076	The `public` element in a catalog file is not supported when using the XML Catalogs feature. It is not supported to be consistent with JAX-WS EntityResolver implementation. WLS only supports defining the `system` element in a catalog file.	10.3	Will not fix
CR350154	When trying to generate WLS 9.x client artifacts in WLS 10.3, the built-in task `clientgen` cannot be used directly. **Workaround or Solution:** In WLS 10.3, the Ant built-in `clientgen` task's `typedef` class is `weblogic.wsee.tools.anttasks.ClientGenTask`. If you want to generate WLS 9.x client artifacts in WLS 10.3, you must either: • use the `clientgenx` task, whose `typedef` class is `weblogic.ant.taskdefs.webservices.clientgen.ClientGenTask` or • `taskdef` the `clientgen` as the following line in the Ant build project file: `<taskdef name='clientgen' classname='weblogic.ant.taskdefs.webservices.clientgen.ClientGenTask'/>`	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR350930	The local `xmlcatalog` element does not work well due to an Ant limitation. **Workaround or Solution:** In the ant `build.xml` file, you have to define a local element above a `clientgen(wsdlc)` task when you are in the same target, or define the element out of any targets.	10.3	Will not fix
CR353291	In prior releases, WSSC+RM interop from WLS to .NET fails during trust handshake because the algorithm suite policy is not used for the WSSC key length. WLS correctly interprets the WS-SecurityPolicy specification with respect to maximum key length for symmetric keys. Some implementations, including Microsoft and possibly others, treat the key length specified as the minimum in the policy's algorithm suite as the 'correct' key length, and will not accept the maximum length, even though it is legal. In WLS 10.3, client side behavior has been changed to always use the minimum key length specified by the algorithm when deriving keys, even though the maximum key length is legal and more secure. This allows WLS to interop properly with Microsoft. As a result, WLS 10.3 uses shorter (and therefore less secure) key lengths than in previous releases. All derived keys, however, will be at least the minimum key length specified by the policy's algorithm suite.	10.3	
CR353484 CR363329	WLS 10.3 honors the `includeToken` attribute setting on `SecureConversationToken` Policy elements. In previous releases, the Token was always included, which was incorrect.	10.3	10.3
CR361779	When you use an inner class of an inner class as a parameter of `webmethod` in JWS, a warning occurs in the JWSC task. **Workaround or Solution:** Avoid using an inner class of an inner class as a parameter of `webmethod` in JWS.	10.3	
CR365414	Unlike previous releases, WSL 10.3 now honors 'sign before encrypt' policies.	10.3	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR367420	In WLS 10.0, X509 certification was not used for authentication even if the `UseX509ForIdentity` flag was set. This has been corrected in WLS 10.3. **Workaround or solution:** Set `UseX509ForIdentity` to `false` when you create the default `x509` handler using a Python script to configure domains, as shown here: ``` wsm = cmo.lookupWebserviceSecurity ('default_wss') wtm = wsm.createWebserviceTokenHandler ('default_x509_handler') wtm.setClassName('weblogic.xml.crypto.wss. BinarySecurityTokenHandler') wtm.setTokenType('x509') wtm.setHandlingOrder(2) cpm = wtm.createConfigurationProperty ('UseX509ForIdentity') cpm.setValue('false') ``` If the above code is not added into startup scripts, some secure web services that worked in WLS 10.0 may not work in WLS 10.3.	10.3	10.3
CR368031	A Web Service that uses JMS transport does not load balance requests after restart of a failed clustered server, or if a server starts up long after other servers in the cluster have started. This is because the newly booted clustered server does not have a chance to receive and process messages. This issue may impact the load balance performance, but does not have any impact on fail-over capability.	10.3	
CR369432	The WLS Web Service JAXRPC client doesn't encode the HTTP SOAPAction header with multi-byte characters, but WLS server only supports ASCII for HTTP headers. **Workaround or Solution:** Change the SOAP action to ASCII in the wsdl.	10.3	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR369721	In WLS 10.0, when using JAX-WS, the following error could occur when there were no duplicate response wrapper bean names: `Response wrapper bean names must be unique and must not clash` In WLS 10.3, this no longer happens.	10.0	10.3
CR371461	An external catalog file cannot be used in the `xmlcatalog` element of a clientgen task. For example, this snippet of an ant build file will not work: ``` <clientgen ... <xmlcatalog> <catalogpath> <pathelement location='wsdlcatalog.xml'/> </catalogpath> </xmlcatalog> ``` This is a limitation of the Ant XML Catalog. **Workaround or Solution:** Resource locations can be specified either in-line or in an external catalog file(s), or both. In order to use an external catalog file, the `xml-commons` resolver library (`resolver.jar`) must be in your classpath. External catalog files may be either plain text format or XML format. If the `xml-commons` resolver library is not found in the classpath, external catalog files, specified in `<catalogpath>` paths, will be ignored and a warning will be logged. In this case, however, processing of inline entries will proceed normally. Currently, only `<dtd>` and `<entity>` elements may be specified inline. These correspond to the OASIS catalog entry types PUBLIC and URI respectively.	10.3	Will not fix

WebLogic Tuxedo Connector

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR127660	View classes are not set on a per connection basis.	8.1 SP2	Will not fix
	A shared WebLogic Tuxedo Connector hash table can cause unexpected behavior in the server if two applications point to the same VIEW name with different definitions. There should be a hash table for the view classes on the connection as well as for the Resource section.		
	Workaround or Solution:		
	Ensure that all VIEW classes defined across all your WebLogic Workshop applications are consistent, meaning that you have the same VIEW name representing the same VIEW class.		

Miscellaneous

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR198347	The database drivers were unable to support any type of encryption for the connections to their respective database servers.	9.0	10.3
	DataDirect now provides SSL support to enable the encryption of JDBC connection attributes. As a result, JDBC connections can now be encrypted. Further information on what is available is outlined herein:		
	Oracle		
	• New `EncryptionMethod` connection option for configuring data encryption across the network.		
	• New connection options for configuring how the driver implements SSL encryption, including		
	– `ValidateServerCertificate` and `HostNameInCertificate` connection options for configuring certificate validation behavior.		
	– `TrustStore` and `TrustStorePassword` connection options for configuring truststore information.		
	– `KeyStore`, `KeyStorePassword`, and `KeyPassword` connection options for configuring keystore information for SSL client authentication.		
	SQL Server		
	• New `EncryptionMethod` connection option for configuring data encryption across the network.		
	• New connection options for configuring how the driver implements SSL encryption, including:		
	– `ValidateServerCertificate` and `HostNameInCertificate` connection options for configuring certificate validation behavior.		
	– `TrustStore` and `TrustStorePassword` connection options for configuring truststore information.		

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR198347 (cont.)	**Sybase**
	
	• New `EncryptionMethod` connection option for configuring data encryption across the network.
	
	• New connection options for configuring how the driver implements SSL encryption, including:
	
	  – `ValidateServerCertificate` and `HostNameInCertificate` connection options for configuring certificate validation behavior.
	
	  – `TrustStore` and `TrustStorePassword` connection options for configuring truststore information.
	
	DB2
	Currently on iSeries only. Support on LUW and z/OS is planned.
	
	• DB2-specific encryption support.
	
	• New `EncryptionMethod` connection option for configuring data encryption across the network.
	
	Informix
	No encryption support planned
	
	See the Reference guide at http://media.datadirect.com/download/docs/jdbc/alljdbc/wwhelp/wwhimpl/js/html/wwhelp.htm for more information relating encryption changes to the Oracle, SQL Server, and Sybase drivers.
	
	See the Reference Supplement guide at http://media.datadirect.com/download/docs/jdbc/alljdbc/wwhelp/wwhimpl/js/html/wwhelp.htm for more information relating to the DB2 driver encryption changes.		

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR223728	WebLogic Server now packages the standard Ant distribution in `WL_Home/modules/org.apache.ant_1.6.5`. If you use the standard WebLogic Server environment scripts created as a part of the installation, you will not need to make any changes to continue using Ant. If instead you directly reference the Ant jars from the WebLogic Server installation, you will need to modify the classpath to point to the new location of the Ant distribution. The `lib/ant-all.jar` in that directory has a manifest classpath that refers to all of the other JARs in the `lib` directory. You should use `ant-all.jar` rather than `ant.jar` if you want to maintain the previous behavior, where the manifest classpath of `ant.jar` was modified to include all of the JARs in the `lib` directory.	10.0	NA
CR272167	The MS SQL Server driver reports `conn.prepareStatement(String sql, String[] columnNames)` as an unsupported method. This is true for the following methods for Microsoft Server 7 and SQL Server 2000 only: Connection.prepareStatement(String sql, int[] columnIndexes) Connection.prepareStatement(String sql, String[] columnNames) Statement.execute(String sql, int[] columnIndexes) Statement.execute(String sql, String[] columnNames) Statement.executeUpdate(String sql, int[] columnIndexes) Statement.executeUpdate(String sql, String[] columnNames) These methods will return the value of the identity column. However, the driver can't implement these methods correctly because these DBMSs do not allow you to return the value of an arbitrary column from an insert, update, or delete statement. **Workaround or Solution:** Set the `WorkArounds` connection option to 1. When `Workarounds=1`, calling any of the auto-generated keys methods listed above returns the value of the identity column regardless of the name or index of the column specified to the method. If multiple names or indexes are specified, the driver throws an exception indicating that multiple column names or indexes cannot be specified if connected to Microsoft SQL Server 7 or SQL Server 2000. The `WorkArounds` value causes the driver to support the previously unsupported methods.	8.1SP5	10.3

Change Request Number	Description and Workaround or Solution	Found In	Fixed In
CR290964	The manifest classpath for `weblogic.jar` no longer contains `ant-contrib.jar`. As a result, if you use `ant-contrib` tasks and get an error message such as `"Could not load definitions from resource net/sf/antcontrib/antlib.xml. It could not be found"`, this means that `ant-contrib.jar` is not on the classpath. If you use WebLogic Server's `setWLSEnv` scripts, `ant-contrib.jar` is added to the classpath; however if you are using your own scripts that previously put `weblogic.jar` on the classpath, you need to update your scripts to put `ant-contrib.jar` on the classpath. **Workaround or Solution:** Either add the standard distribution `ant-contrib.jar` to the classpath or add the one supplied with WebLogic Server: `BEA_HOME/modules/net.sf.antcontrib_1.0b2.0/lib/ant-contrib.jar`	10.0	NA
CR293355	The default Ant version for the machine may cause incompatibilities. Some older Ant installations on UNIX may create a file named `/etc/ant.conf`. This file points to the Ant installation to be used on the machine. Older installations of Ant do not contain all the features required by WebLogic Server. This problem may be evident by errors such as: `ant -version` `java.lang.NoClassDefFoundError: org/apache/tools/ant/launch/Launcher` **Workaround or Solution:** Delete the `/etc/ant.conf` file, or use `ant -noconfig`.	10.0	

Change Request Number	Description and Workaround or Solution	Found In	Fixed In

CR312952	When you use the `JarBuilder` utility to build a client, you may receive an error like this:
	
	`Invalid implementation version between Ant core and Ant optional tasks.`
	
	This occurs because the resulting `wlfullclient.jar` includes Ant classes that have a version number that is inconsistent with the Ant version you have in your classpath.
	
	Workaround or Solution:
	
	Add `ant.jar` from your local Ant installation ahead of `wlfullclient.jar` in your client classpath. An alternative is to add the Ant module from the WLS installation ahead of `wlfullclient.jar` on the classpath:
	
	`$BEA_HOME/modules/org.apache.ant_1.6.5/lib/ant-all.jar`
		10.0	10.3

CR323840	When you install version 3 of Smart Update on an existing BEA_HOME, and attempt to use it in command-line mode, an `Unsupported product release` error occurs and Smart Update exits.	10.0	10.3
CR363918	The Oracle ojbdc14.jar file has been changed to ojdbc6.jar, for use with JDK 5 or 6. As a result, any explicit references you make to ojdbc14.jar must be changed to ojdbc6.jar.	10.3	

CR375831	Under Linux 5.0, when running an Ant script that uses `wlappc`, the following error occurs because Ant could not find the `wlappc` task or a class upon which it relies:
	
	`Could not create task or type of type: wlappc.`
	
	Workaround or solution:
	
	Place the `wlappc` task into the build xml file, for example:
	
	`<taskdef name='wlappc' classname='weblogic.ant.taskdefs.j2ee.Appc'/>`
		10.3	