BEA WebLogic® Integration

9.2 Release Notes

Version 9.2 MP3
Document Revised: March, 2008

Contents

1. WebLogic Integration Release Notes

2. Known Limitations

3. Problems Fixed in 9.2 MP3

CHAPTER 1

WebLogic Integration Release Notes

This document provides release note information on the WebLogic Integration 9.2 release. This section includes the following topics:

Introduction

BEA WebLogic® Integration is a unified solution for integrating business systems within an enterprise. WebLogic Integration provides a development and run-time framework that unifies the components of business integration into a single flexible environment. The components include business process management, data transformation, trading partner integration, connectivity, message brokering, application monitoring, and user interaction.

WebLogic Integration combines divergent components of the business integration picture, including ERP, CRM, legacy applications, business users, supply chains, and trading partners. It enables IT operations to meet business goals by delivering process-centric composite applications quickly and efficiently.

What's New in BEA WebLogic Integration 9.2

WebLogic Integration 9.2 supports an Eclipse-based development environment, Apache Beehive integration controls, Java 5-compliant annotations, and a new Worklist subsystem. In addition to automated upgrade tools for applications from WebLogic Integration 8.1 SP4, SP5 and SP6, or 8.5 (including SP5 and SP6), WebLogic Integration 9.2 leverages the power of WebLogic Server 9.2, providing improved security, manageability, performance, scalability and availability.

WebLogic Integration 9.2 includes the following new features and enhancements:

Eclipse-Based Development Environment

The BEA Workshop for WebLogic Platform 9.2 product family and WebLogic Portal 9.2 provide an Eclipse-based development environment. WebLogic Integration 9.2 adds the design capabilities of Process and Task Plan (Worklist) to the Eclipse IDE (Integrated Development Environment), delivering a unified and extensible IDE that helps developers build standards-based, enterprise-class process applications.

For details on the Eclipse IDE, see Section Eclipse-Based Integration Development Environment in the *Introducing BEA WebLogic Integration* at the following URL:

http://edocs.bea.com/wli/docs92/overview/index.html

A tutorial that guides you through creating and testing a business process (JPD) using the new Eclipse-based design environment is provided. For more information, see "Build Your First Business Process (non-Worklist) Application" on page 1-10.

For information about the Workshop for WebLogic Platform 9.2 and WebLogic Portal 9.2, see the product documentation at the following URLs:

http://edocs.bea.com/workshop/docs92/

http://edocs.bea.com/wlp/docs92/

Standards Support

Components and applications you build with WebLogic Integration 9.2 are based on standard technologies such as Apache Beehive, Java 5 annotations, BPEL (import and export) and the XQuery 2004-compliant XQuery Mapper and engine.

Apache Beehive Controls

Controls are now based on the Beehive controls framework—a lightweight component framework based on annotated JavaBeans. For information about the Apache Beehive Open Source Project, see http://beehive.apache.org/.

Java 5 Annotations for Declarative Process Development

WebLogic Integration 9.2 provides support for Java 5-compliant annotations. This facility allows users to employ a declarative programming style in which the programmer annotates the process and the tool automatically generates the necessary Java code. BEA continues to provide tool support to keep the use of annotations simple—WebLogic Integration 9.2 includes a process annotation editor, in addition to the process property editor.

BPEL Import and Export

The BPEL Import/Export Wizard in now available from the Eclipse-based IDE.

Upgraded XQuery Mapper and Engine

WebLogic Integration 9.2 supports a new XQuery 2004-compliant Xquery Mapper and engine. It is the same robust engine used by AquaLogic Service Bus and AquaLogic Data Services Platform.

New Worklist Subsystem

WebLogic Integration 9.2 includes a new Worklist subsystem for integrating people into processes. The key areas of new functionality include a new metadata definition of multi-step tasks called a Task Plan, a new design editor for Task Plans in Eclipse and the Worklist User Portal.

Task Plan as Metadata Definition of Multi-Step Tasks

Task plans model the lifecycle of a task that can be specific to the business context. A task lifecycle can go through multiple steps with different human participants assigned to it at each step.

Form-Driven Testing in Worklist User Portal

Input forms are automatically generated based on task plan configurations to enable full scenario testing for each plan.

Graphical Design Environment in IDE

WebLogic Integration 9.2 includes a new Eclipse-based design canvas for Worklist, offering a visual drag-and-drop environment to define multi-step task plans.

Task Portlets

The new Worklist subsystem offers a set of pre-configured portlets for users and managers to manage their tasks.

Dynamic E-mail Notification

In WebLogic Integration 9.2 Worklist, users can configure system-generated E-mail notifications for task-related events such as creation, assignment, completion and cancellation.

Automated Task Balancing

For cases in which more than one user is available to perform a task, Worklist automatically assigns the task to the user who has the fewest outstanding items due.

Upgrade Tools for Applications Created in WebLogic Integration 8.1 or 8.5

You can upgrade many of your application artifacts from WebLogic Integration 8.1 SP4, SP5 and SP6, or 8.5 (including SP5 and SP6), to WebLogic Integration 9.2 automatically.

WebLogic Integration 9.2 provides a source upgrade tool that handles operations such as upgrading the application to the Eclipse-based project structure and converting the WebLogic Integration 8.x style Java annotations into Java 5-compliant annotations. You can also convert XQuery 2002 to XQuery 2004.

For detailed information on how to upgrade to 9.2, see the following URL:

http://edocs.bea.com/wli/docs92/upgrade/index.html

A new tutorial that describes how to use the 9.2 upgrade tools to upgrade an 8.1 (SP4 or higher) application to 9.2 is provided. For more information, see "Upgrade a WebLogic Integration 8.1 Application Source to 9.2 Environment" on page 1-9.

Deprecated Items

Following is a list of items that have been deprecated in 9.2. They will not be supported and will be removed in the next major release of WebLogic Integration. It is recommended that you follow the alternatives, where mentioned, to avoid conflicts in design of your process applications in the future.

- Application Integration — All components of Application Integration are deprecated.
- Using JPD as a Web Service — It is recommended you use Java Web Services as an interface with the JPD.
- Service Broker Control — This control is designed to support backward compatibility. Its use is recommended only when communicating with WebLogic Integration 8.x JPDs.
- Worklist APIs — Most of the 8.x Worklist API are no longer supported and have been replaced by their 9.2 variants.

Platform Support and System Requirements

This section describes the platforms on which WebLogic Integration 9.2 is supported and other system requirements. It includes the following topics:

Prerequisites

Install WebLogic Integration 9.2, apply the required patches, and create a WebLogic Integration domain as described in "Installation and Required Patches" on page 1-6.

Operating Systems

For information about Operating Systems, see List of Supported Operating System Configurations in the Supported Configurations document which is available at: http://edocs.bea.com/platform/suppconfigs/configs92/92_over/overview.html#1122259

Databases and Database Drivers

For more information about databases and database drivers, see http://edocs.bea.com/platform/suppconfigs/configs92/92_over/overview.html and http://edocs.bea.com/platform/suppconfigs/configs92/92_over/supported_db.html#1170001.

Upgrading Applications

You can upgrade WebLogic Integration 8.1 SP4, SP5 and SP6, or 8.5 (including SP5 and SP6), applications to WebLogic Integration 9.2 using the upgrade tools. For more information on the upgrade tools see, "Upgrade Tools for Applications Created in WebLogic Integration 8.1 or 8.5" on page 1-4.

Ensuring That the Upgrade Wizard Has Enough Memory

Before using the import wizard to upgrade applications, consider temporarily increasing the maximum amount of memory that the Java Virtual Machine allows to Workshop for WebLogic. The upgrade process requires a compilation step that potentially includes a large number of files. The recommended maximum memory is 1 GB.

You can increase maximum memory in the following way: Before starting the IDE, edit the `workshop4WP.ini` file to replace the `-Xmx` value with a sufficiently high memory maximum. By default, the `workshop4WP.ini` file is located at: `BEA_HOME\workshop92\workshop4WP\workshop4WP.ini`.

For example, you can consider changing the setting from `-Xmx768m` (the default) to `-Xmx1G` (to set it to 1 GB). After you finish using the upgrade tools, set the memory maximum back to a level that is appropriate for development.

For more information about the upgrade wizard, see *How To: Use the Import Wizard to Upgrade Version 8.1 Applications* at the following URL:
http://e-docs.bea.com/workshop/docs92/ws_platform/upgrading/howUseUpgradeWizard.html

Installation and Required Patches

Complete the following steps to install and configure the WebLogic Integration 9.2 software:

1. "Install the WebLogic Integration 9.2 Software" on page 1-6
2. "Install the Required Patches" on page 1-7

Install the WebLogic Integration 9.2 Software

Install the software using either graphical mode, console mode, or silent mode. The installation program prompts you to enter specific information about your system and configuration. See the following URL for information about installing BEA Products:

`http://edocs.bea.com/common/docs92/install/index.html`

Install the Required Patches

After completing the installation of WebLogic Integration, you have the option to allow the installer to apply patches to WebLogic Server, Workshop, and WebLogic Portal (if present), required for development with WebLogic Integration. If you cleared the **Apply Patches** check box, and opted to apply them yourself, you can use Smart Update to apply the patches.

The automatic process of applying patches may fail due to machine or general software failure or patch-related errors: in these cases also, you can use Smart Update to apply patches manually. If you are unable to apply the patches using Smart Update, contact BEA support.

The Smart Update tool is described in Downloading and Installing and Patches at the following URL:

http://edocs.bea.com/common/docs92/smart_update/index.html

When you start Smart Update, you are prompted to log into BEA Customer Support using your BEA Support ID and password. How to obtain a BEA Support ID is explained in "Obtaining a BEA Account and Accessing Your Contract Entitlement" in Starting Smart Update at the following URL: http://edocs.bea.com/common/docs92/smart_update/start.html.

After logging in, you can access public and optional patches. When you complete downloading and applying the patches, you will find them under `BEA_HOME\utils\bsu\cache_dir`.

You must install the patches listed in Table 1-1 to your WebLogic Integration installation using the Smart Update tool.

> If your WebLogic Integration installation does not include WebLogic Portal and Workshop, the WebLogic Portal and Workshop patches are not applicable to your installation.

Note: If you have Workshop installed, you must install the patches listed in Table 1-1 to your BEA WebLogic Integration installation using the Smart Update tool, and then run the `<BEA_HOME>\workshop92\workshop4WP\WLWPatchInitialize.cmd` script to activate the patches for Workshop.

Table 1-1 Required Patches for WebLogic Integration

Patch ID	Passcode	Products
6MK8	7Y7F86FU	WebLogic Server 9.2
A13Q	A4KAV4WX	WebLogic Server 9.2

Table 1-1 Required Patches for WebLogic Integration

Patch ID	Passcode	Products
93HW	5V6IUZKU	WebLogic Server 9.2
6L1H	SLHPWYVJ	WebLogic Server 9.2
W17P	VQHMZ67A	WebLogic Server 9.2
BINS	J3K3II9X	WebLogic Server 9.2
KDQ7	5V4J5JJI	WebLogic Server 9.2
76ES	8UMFE4XG	WebLogic Server 9.2
XE28	QFCU24WG	WebLogic Server 9.2
JZY3	RN1HN5ZL	WebLogic Server 9.2
NSX5	8RUJSA94	WebLogic Server 9.2
YWJE	7P3F6IKV	WebLogic Server 9.2
SWV7	ZMYXA2M2	WebLogic Server 9.2
XM28	QT4XQ8JK	WebLogic Server 9.2
EJQL	FQUQNFW4	WebLogic Portal 9.2
9Z62	K2N7U4DP	WebLogic Server 9.2
T4CP	BM1MJ5BQ	WebLogic Server 9.2
P816	JGGWL8BK	BEA Workshop for WebLogic Platform 9.2
S35L	PKQ6X6B2	BEA Workshop for WebLogic Platform 9.2
81H9	T9P2NPG3	BEA Workshop for WebLogic Platform 9.2
C8KW	H4828P67	BEA Workshop for WebLogic Platform 9.2
82WQ	JE6Z2EPM	WebLogic Server 9.2
KKQT	7X38HKMX	WebLogic Server 9.2
47D5	89QCIL88	WebLogic Server 9.2
5LT3	748HL4I7	WebLogic Server 9.2

Tutorials

To get hands-on experience with WebLogic Integration 9.2, we recommend that you try out the following tutorials listed in this section. To gain experience with WebLogic Integration 9.2, work through one or more of the tutorials listed below:

- Upgrade a WebLogic Integration 8.1 Application Source to 9.2 Environment
- Build a New Worklist Application
- Build Your First Business Process (non-Worklist) Application

The content for each of these tutorials is described in the following sections. In addition, try out the Workshop for WebLogic Platform tutorials that are available at the following URL:

http://edocs.bea.com/workshop/docs92/ws_platform/introduction/conWorkshopTutorials.html

Upgrade a WebLogic Integration 8.1 Application Source to 9.2 Environment

This tutorial describes how to use the automated upgrade tools available in WebLogic Integration 9.2. The upgrade wizard scans your application, and alerts you of possible problems by displaying error and warning messages. During the upgrade process, it automatically converts your 8.1 application into the new Eclipse-based project structure and upgrades the 8.1-style annotations into the new Java 5-compliant annotations. The upgrade tool maintains a running log of status and messages, which are displayed in the IDE. The tutorial is available at http://edocs.bea.com/wli/docs92/upgradetutorial/index.html.

Build a New Worklist Application

This tutorial shows you how to define a Loan Approval process using the new WebLogic Integration 9.2 Worklist subsystem. You learn how to define a task plan including task steps, actions in each step, authorized users for each step, task due date and other task attributes and properties. You will also learn how to create a task instance and test the entire task plan using a system generated portal view.

When you complete this tutorial you will have learned how to:

- Set up the WebLogic Integration 9.2 Worklist environment.
- Create a task plan.

- Test the loan processing approval system using the Worklist user portal.
- Manage loan processing task instances using the Worklist Console.
- Create task instances using a JPD.
- Accept or reject a loan using a JPD.
- Develop custom business-specific user interface elements and integrate them into the user portal.

The tutorial is available at
`http://edocs.bea.com/wli/docs92/worklisttutorial/index.html`.

Build Your First Business Process (non-Worklist) Application

This tutorial guides you through creating and testing a new business process (JPD) using the new Eclipse-based design environment available in BEA Workshop and WebLogic Integration 9.2.

When you complete this tutorial you will have learned how to:

- Design communication nodes in a business process—in other words, create the interface between your business process and its clients and resources. Clients of business processes can be any other resources or services that invoke business processes to perform one or more operations.
- Design the interactions with clients, including creating the methods that expose your business process's functionality.
- Design the interactions with resources using controls. WebLogic Platform controls make it easy to access enterprise resources, such as databases, Enterprise Java Beans (EJBs), Web services, and other business processes from within your application.
- Handle XML, non-XML, and Java data types in the business process—includes working with XML schemas and transforming data between disparate data types using the Transformation tool.
- Design business processes to publish and subscribe to message broker channels.

The tutorial is available at http://edocs.bea.com/wli/docs92/jpdtutorial/index.html.

CHAPTER 2

Known Limitations

This section describes known limitations in the BEA WebLogic Integration 9.2 release software. The known limitations are grouped by the following topics:

- Administration, Configuration, and Worklist Application
- Application Integration
- Business Process
- Cluster Configurations
- Controls
- Event Generators
- Data Transformation
- Trading Partner Integration
- Upgrading WebLogic Integration

Administration, Configuration, and Worklist Application

Problem ID	Description
CR302006	**If you installed WebLogic Integration and opted to apply patches after successful completion of the installation, the installer automatically applies patches to WebLogic Server, Workshop for WebLogic Platform, and WebLogic Portal.** The installer does not check whether WebLogic Portal or Workshop for WebLogic Platform are installed or not. So, patches to these two products are applied even if they are not installed If WebLogic Portal or Workshop for WebLogic Platform are installed at a later point, the installer overwrites the patches but Smart Update shows the patches as applied. **Workaround:** Perform the following steps. 1. Uninstall the WebLogic Portal or Workshop for WebLogic Platform patches. 2. Install WebLogic Portal or Workshop for WebLogic Platform. 3. Apply the WebLogic Portal or Workshop for WebLogic Platform patches using Smart Update.
CR302929	**FileNotFoundException when you install WebLogic Integration without Workshop.** If you do not have Workshop for WebLogic Platform installed and opt to apply patch at the end of your WebLogic Integration installation, you will encounter FileNotFoundException. **Operating System**: all UNIX platforms **Workaround**: Include Workshop as part of your custom installation or do not select **Apply Patch**, then apply WebLogic Server and Portal (if installed) patches manually via Smart Update. If the exception already occurs, you can use Smart Update to apply the remaining WebLogic Server patches.
CR301122	**When deploying a Process application containing a JWS having a web method parameter of type XBean, with the XSD for the type/element defined in the `<web project>/schemas` folder EAR deployment fails.** **Workaround**: Make a copy of the XSD to the `<utility project>/schemas` folder as well.

Problem ID	Description
CR300337 CR287765	**When JWS invokes JPD, the callback to JWS from a JPD fails in a cluster.** **Workaround:** The `com.bea.control.annotations.MessageBuffer` must not be used on the callback operations of a Process control that is used to invoke a JPD from a JWS. Using the `com.bea.control.annotations.MessageBuffer` annotation will not work reliably in a cluster environment.
CR300140	**While editing a newly created event subscriptions of a worklist instance, two Edits interfere each other.** **Workaround**: Delete the event subscription created, and recreate it.
CR300060	**During the WebLogic Integration upgrade, in the WebLogic Server Console, when you access the deployment section, navigate to a particular application and click on the + sign to expand, the navigation section changes to "Showing 1 - 10 of 10" and previous/next links are disabled.** **Workaround**: Click on deployment link from left navigation and navigate to same page again.
CR299266	**After application source upgrade, MFL Transformation failed with XQuery Exception (XQRLUserException)**. Unlike in 8.x, the 9.2 MFL-derived XMLBeans belong to a namespace. So, the upgraded XQuery transformations that use MFL-derived XMLBeans as output types must be manually updated to yield XML with elements in the appropriate namespace. The namespace used is determined by the path of the MFL file, relative to the schema source directory. For example, an MFL file located at `project/schemas/dir/purchase.mfl` will yield XMLBeans that belong to the XML namespace: dir/purchase.
CR296394	**The user portal displays default values that are actually not in the database. If you accept these values, they are inserted into the database.** At runtime, an API call for a property value does not return the value by default. If the user properties can be: • Edited, the User Portal shows the default value • Viewed (read only mode), the User Portal shows the current value If the user description can be edited, the User Portal shows the value at the closest level.

Problem ID	Description
CR296376	**The User Portal does not display the request XML when running an 8.1x upgraded application.** **Workaround**: None.
CR295076	**You might encounter a database deadlock with Worklist applications if you have configured your domain with DB2**. **Workaround:** There is no known workaround at this point.
CR292434	**Potential database deadlock when using Microsoft SQL.** You might encounter a database deadlock if you have configured your domain with Microsoft SQL. **Workaround**: After you have created your domain, modify the datasource scripts to add the "`SendStringParametersAsUnicode=false`" property.

Problem ID	Description
CR290980	**Issues due to WebLogic Portal 9.2 template patches**.
CR295210	When you create a Platform-like by adding `wlp.jar` template onto a WLI domain, with express mode or user explicitly selecting run DB script, loadDB fails when extending `wlp.jar` and the domain does not start. **Workaround**: Using the **Configuration Wizard** tool, create a Platform-like domain by adding WebLogic Integration templates to a WebLogic Portal domain, or by selecting the WebLogic Portal and WebLogic Integration check-boxes. --- When you create a Platform-like by adding the WebLogic Portal component onto a WebLogic Integration domain, with express mode or user explicitly selecting run DB script, loadDB fails when extending `wlp.jar` and the domain does not start. **Workaround**: There is no workaround for this problem. --- When you create a WebLogic Portal domain by extending `workshop_wl.jar`, `p13n.jar` and `wlp.jar` templates, one template at a time, with express mode or user explicitly selecting run DB script, loadDB fails when extending `wlp.jar` and the domain does not start. **Workaround:** Select the WebLogic Portal checkbox to create the Portal domain. --- When you create a WebLogic Portal domain by extending `workshop_wl.jar`, `p13n.jar` and `wlp.jar` templates, one template at a time, with loadDB() invoked each time extending a template, loadDB fails when extending `wlp.jar` and the domain does not start. **Workaround:** Extend the required templates together and invoke loadDB() explicitly in the WLST script. --- When you create a WebLogic Portal domain by extending `workshop_wl.jar`, `p13n.jar` and `wlp.jar` templates together, without explicitly invoking loadDB(), the resulting domain fails to start because of missing tables. **Workaround**: You need to explicitly invoke loadDB().

Problem ID	Description
CR293795	**WebLogic Integration 9.x domains take longer to start up when compared with WebLogic Integration 8.x domains.** The problem has been mitigated by removing extra backward compatibility resources from newly created WebLogic Integration domains. Newly created domains will not deploy backward compatibility resources (applications, etc.) and will thus boot more rapidly than if those resources were present in the domain. **Note:** Even with this change, the boot time for 9.2 domains is somewhat longer than a comparable 8.x domain. This is due to more sophisticated application structure and deployment needs in 9.2. If you wish to create a new domain and also add backward compatibility support to that domain, you must create the WebLogic Integration domain, and then extend that domain with the wli_worklist81x extension template (using the configuration wizard). WebLogic Integration 8.1.x domains that are upgraded to 9.2 will *automatically* have backward compatibility support added to them. **No further customer action is required.**
CR298397	**If your application enables message buffering on controls or web services using the annotations `com.bea.control.annotations.MessageBuffer` or `weblogic.jws.MessageBuffer`, the following error is displayed when you try to deploy it on a cluster:** "While attempting to create destination MSG_BUFFER_QUEUE in module <name_of_your_ear>!WlwRuntimeAppScopedJMS the JMSServer of name <name_of_your_cluster> could not be found". **Workaround**: In this case, you must target the MSG_BUFFER_QUEUE to a specific JMS Server with the following submoduletargets option to your weblogic.Deployer command: `-submoduletargets cgJMSServer@WlwRuntimeAppScopedJMS@WseeJmsServer_auto_1`.
CR295076	**You might encounter a database deadlock with Worklist applications if you have configured your domain with DB2**. There is no known workaround at this point.

Problem ID	Description
CR290839	**An error is displayed from the server console while accessing User Management from the console**. **Workaround**: This is a known issue, and the exception is not critical. The IllegalStateException that is displayed can be ignored.
CR291534	**Any new task plan, by default does not have a create policy for any user: this has to be done explicitly through the worklist console. If this policy is not configured, then the JPDs' that create tasks and are initiated through the JPD test console fail: This is because the JPD test console invokes the JPDs' without any authentication and the JPD is run without any principal, thus as 'Anonymous'.** **Workaround**: You need to change the create policy and remove the <Anonymous>. By default <Anonymous> is added to the Global Task Plan Create policy. Thus, a task can be created Out of the Box by anyone.
CR289425	**Warning and Debug messages are displayed while starting the Out-of-the-box Integration domain.** **Workaround**: The administrator, in a production mode, must set a test table depending on the chosen database - for the databases supported by WLI.
CR288712 CR288827	**You will get an exception when you try to extend any portal-enabled domain with the WSRP template (`wsrp-simple-producer.jar`) either using the Configuration Wizard or WLST offline.**
CR288966	**Exception message displayed in the purging tasks of Worklist application.** Based on the permissions you have, you can perform the purge action. An exception message is displayed when the tasks are of type "Compat 8.1.x". <Anonymous> users can purge 8.x Compat Tasks.
CR283676	**When you enter a custom User Interface page for the first time in a browser session, a blank page is displayed.** **Workaround**: You need to refresh your browser to view the custom User Interface page.The usage of an incorrect case in the `task-ui-registry.xml` entries can cause the related custom task page to appear blank for the first time, and is then displayed after the page is refreshed.
CR283190	**The Managed Server does not start unless the Administration Server already running.** **Workaround**: None.

Problem ID	Description
CR287304	**Worklist database scripts for Sybase need to be ported to WebLogic Integration 9.x.** The problem has been resolved. The SQLs have been updated.
CR285577	**In the Worklist console screen, the Home link does not point to the correct screen after creating a user in UserManagement.** **Workaround**: The user management links from the JPD console point into the worklist console. The problem is fixed and the **Home** button works correctly.

CR277602	**JVM Memory Errors**
	While working on the Worklist Console, you may encounter the java.lang.OutOfMemoryError: PermGen space memory error.
	Workaround: You must set the domain JVM maximum memory as `-Xmx1024m` and the maximum PermSize as `-XX:MaxPermSize=256m` in the respective Domain environment setting files.
	For Windows:
	`domain/bin/setDomainEnv.cmd`
	For UNIX Platforms:
	`domain/bin/setDomainEnv.sh`

CR272488	**Exceptions are encountered in the Server Console when using the Worklist Sample Application.** While working on the Worklist Sample Application through the Worklist User Portal, you may encounter `org.apache.beehive.netui.util` type exceptions in the Server Console. You can ignore the exceptions displayed in the Server Console as they are not very critical.
CR268925	**Unauthorized users (users who do not have Administration/Update/Query rights, or claimant, assignee rights) can see the built-in and user-defined properties of a task. This can happen when a control callback or TaskEventListener delivers a TaskEvent containing this information.** **Workaround**: Asynchronous events still have user properties as `ChangedProperty` instances (each containing a PropertyInstance with a name, type, and so on.) on them, but the property values equals to null. For system properties, they receive a `ChangedProperty` instance with a name, but the value is equal to null.

Problem ID	Description
CR267653	**Worklist cursor pagination does not work when a separate transaction deletes tasks selected by the cursor.** Works as designed.
CR203350	**Worklist Administration Module may throw exception traces** The Worklist Administration Module may throw exception traces instead of clean error messages after simultaneous Worklist operations on the console; the exceptions are valid.
CR180096	**Security Configuration Dependency on web.xml** Do not use the WebLogic Integration Administration Console to access your business process to secure SOAP-HTTP access to your business process if you have already secured your business process using the `security-constraint` element in the web.xml deployment descriptor and the `@common:security` annotation. You will receive a security violation at run time. For more information, see the following: • "`security constraint`" in *Securing Web Applications* at http://e-docs.bea.com/wls/docs92/security/thin_client.html • "`@common:security`" annotation in the *Annotations Reference* at http://e-docs.bea.com/workshop/docs81/doc/en/workshop/javadoc-tag/common/security.html

Problem ID	Description

CR156713	**Result of trackdata() Call Not Recorded For Large Documents on Transaction Rollback**
	To avoid a possible problem in subsequent archiving, if both of the following conditions are met, the result of a `trackdata()` call will not be recorded in the WebLogic Integration process events table.
	The transaction that calls `JpdContext.trackData(XmlObject value)` or `JpdContext.trackData(RawData value)` is rolled back.
	The document passed in is large enough to be stored in the SQL document store.
	Note: A document is persisted to the document store if it exceeds the size specified by the `weblogic.wli.DocumentMaxInlineSize` property in the `wli-config.properties` file. For example, the property is set as follows in the sample integration domain:
	`# Minimum size for documents stored in the SQL Document Store:`
	`weblogic.wli.DocumentMaxInlineSize=524288`
	Workaround: To minimize the risk of trackData calls being affected by a process rollback, call `trackData()` in its own transaction (in a perform node encapsulated within an explicit transaction boundary).

Running Business Processes

You must build applications that use WebLogic Integration functionality in a WebLogic Integration or WebLogic Platform domain. Running business processes and data transformations in a WebLogic Workshop domain is not supported.

Application Integration

Problem ID	Description
CR266987	**The container-managed sign-on portion of the Application Integration Security test fails.** **Workaround**: You need to set the desired principal map after the application redeployment.
CR266949	**AI Workshop EAR project facet is required to enable AI Design-time hookup.** **Workaround**: You need to manually configure and maintain `ai-publish.properties`. The AI design-time works correctly with the Workshop 9.0 IDE assuming the `ai-publish.properties` file is located in `EarContent/META-INF/ai-publish.properties`, and is updated regularly.
CR171722	**In Iterative Development Mode, Applications Deployed as EAR, JAR, WAR, or RAR files Are Not Available to Clients Such as the Application View Design Console** When the server is running in iterative development mode, only applications that are in exploded form in the Workshop Application pane may be accessed by clients such as the Application View Design Console. Applications that are packaged into a deployed EAR file are not accessible. In other words, when you are running in iterative development mode, the following conditions must be met for an application to be available for selection in the Application View Design Console: • The application must be in exploded form (it cannot be an EAR, JAR, WAR or RAR file) • The application directory must contain a file with the extension .work
CR157033	**Event Generator Target Changes for a Suspended Application View Only Apply to New Events (DBMS Sample Adapter)** For the DBMS sample adapter, changes to the event generator target for a suspended application view do not apply to events already in the system. Only new events (those triggered after the change) are assigned to the new event generator target. Events already in the system are processed by the previous event generator target.

Problem ID	Description
CR156862	**Synchronous Service Counts Multiplied By the Number of Managed Servers After a Rollback** When synchronous services are invoked and a transaction rollback occurs before the services are performed, the WebLogic Integration Administration Console displays an incorrect synchronous service count. The count displayed is the number of synchronous services in process multiplied by the number of managed servers in the cluster.
CR155711	**Performance Issues When Posting Event Messages to Remote Application View Clients** Performance issues have been noted due to the time required to post event messages to JMS topics for remote application view clients. **Workaround**: A new option has been added to enable or disable event delivery to remote application view clients. There are two ways to disable event delivery to remote application view clients: 1. Set `-DApplicationViewClientEnabled=false` on the server's Java command line. 2. Call the `AppViewDeploymentMBean.setApplicationViewClientEnabled()` method and pass false as the sole argument to the method. The first method turns off event delivery to remote clients for all application views deployed in the server. The second method turns off event delivery to remote clients only for the application view for which the MBean method is called. With remote event delivery disabled, JMS resources are not consumed for the event topic.
CR155627	**Republish Application Views to Incorporate New EJB Descriptors** The design-time EJB descriptors and the descriptors generated by the application view compiler (invoked when an application view is published) have changed for WebLogic Integration 8.1 SP2. **Workaround**: You must republish your application views to ensure that the modified EJB descriptor is included in your application view EJBs.

Problem ID	Description
CR155471	**Set SupportsLocalTransaction to True on XA Pool for Event Connection** The `SupportsLocalTransaction` option must be set to true on the XA pool used for an event connection. (Otherwise, you must use a non-XA pool for the event connection.) If the option is not set correctly for an XA pool, an exception similar to the following is thrown: `java.sql.SQLException: Logical handle no longer valid` If the DBMS sample adapter uses the same database instance on which the WebLogic Integration tables reside, it is recommended that all other XA connection pools also have the `SupportsLocalTransaction` option set to true. If the adapter uses a separate database instance, then only the XA pool for an event connection should have the `SupportsLocalTransaction` option set to true. **Workaround**: To set this option, use the WebLogic Server Administration Console and navigate to **Services** > **JDBC** > **Connection Pools**.

Problem ID	Description

CR155462	**Multiple Primary Event Generator Instances Result in Database Conflicts**
	The event generator target field is blank by default when the application is first deployed on the cluster. This implies that the event generator should not be activated on any node in the cluster. With the DBMS sample adapter in a clustered environment, multiple instances of a primary generator instance result in database conflicts and errors. Care should be taken to specify an appropriate generator instance specification for each server in the targets list, and ensure that only one server in the list contains the generator instance id '1'. For example, in a three node cluster, the following targets would properly distribute event generator instances on the nodes, with the instance on server1 being the primary instance.
	`Event Generator Targets: server1=[1/3],server2=[2/3],server3=[3/3]`
	However, the following targets would all mistakenly create duplicate primary generator instances:
	`Event Generator Targets: server1,server2,server3`
	`Event Generator Targets: server1=[1/3],server2=[1/3 2/3], server3=[3/3]`
	In the first case, no generator instances are specified, and each server creates an instance with an ID of 1, making it the primary instance. In the second case, two servers explicitly define a primary generator instance.
	If you inadvertently specify event generator targets improperly, you may see database conflicts and errors during event generation at runtime. In this case, see the workaround below.
	Workaround: To reset the event generator instances:
	1. Set the event generator target to an invalid value (anything but an actual server name). This stops all the event generators.
	2. Then set the event generator target to the desired value. This restarts only those generator instances that should be started.

Problem ID	Description
CR138792	**Asynchronous Service Counts and Suspended Async Service Counts Can Be Incorrect in Cases Where Database or EIS Failures Occur** If an XA-capable EIS being used for async service invocations fails, or if the database being used for JMS persistence fails, the transaction enclosing the work being done by an async service will be rolled back. This rolls the async service request itself back onto the AI async request queue, and allows the request to be retried. However, the transaction rollback affects only the async service request and any work it may have done against the EIS, and does not affect the suspended async service counts displayed in the WebLogic Integration Administration Console for an application view or adapter instance. Thus, in cases where the EIS or JMS database fails after the async service count has been incremented to account for a successful async service invocation, and before the containing transaction in the async service processor MDB is committed, the async service count will be higher than the actual effective service count against the EIS (since the work done will be rolled back, but count is not decremented). In cases where the EIS or JMS database fails after a suspended async service completes, and the suspended async service count has been decremented, and before the containing transaction in the async service processor MDB is committed, the suspended async service count will be lower than the actual effective suspended service count in the application integration async request queue (since the request was rolled back, but suspended count is not incremented). Administrators should be watchful for failures in either a transactional EIS or database (generally they will see service failures and a suspended adapter instance and application view related to the EIS failure). After the EIS has recovered from such a failure and suspended async services have been given sufficient time to complete, the administrator should check the suspended async service count. If the count is non-zero (or even negative), the count is incorrect. To allow administrators to force the suspended async service count back to zero in those cases that the displayed count is incorrect, the resetSuspendedAsyncServiceCounts() method has been added to the AppViewSummaryMBean interface. See the Javadoc at the following URL: http://edocs.bea.com/wli/docs92/wli.javadoc/index.html

Problem ID	Description
CR138630	**Asynchronous Service Error Counts Multiplied by the Number of Servers in the Cluster** In a clustered environment, the asynchronous service error counts displayed on the WebLogic Integration Administration Console are multiplied by the number of nodes in the cluster. For example, if 5 asynchronous service invocations fail on a three-node cluster, the asynchronous service error count is shown as 15.

Business Process

Problem ID	Description
CR326895	**A PersistenceException occurs when passing ebXML messages** A `PersistenceException` occurs when passing ebXML messages with binary attachments between two B2B applications, if the `documentmaxinlinesize` parameter in the `wliconfig.properties` file is set to the default value. **Workaround**: Set the `documentmaxinlinesize` value to 2.5 times higher than the data size.
CR302072	**There is a problem displaying the contents of a node editor (in-place editor) in the Process editor canvas when using large fonts on windows.** To view the problem, double click the node Double Click to Select Start Event in a newly created process file. The node editor contents appears cut-off at the bottom. **Workaround**: Try setting a smaller font. You should be able to see the complete contents of the editor.

Problem ID	Description

CR297730	**In WLI 9.2, if a synchronous request is made to a frozen or suspended JPD, a null response is returned.**
	This response is confusing to the user, who may be led to believe the request was processed successfully.
	Workaround: This feature works fine as expected for suspended and frozen processes, the following error is displayed:
	`<Nov 1, 2006 5:08:35 PM IST> <Error> <WLI> <000000>`
	`<Failure=com.bea.wli.bpm.runtime.ProcessInstanceUnavailableException: Process instance unavailable due to status: 'Frozen'. Service URI: '/WebProject1/test/Test1.jpd'. InstanceId: '1162381060435'.>`

CR298911	**Endless timeout events getting displayed in Test console.** **Workaround**: The feature works as designed. This is a general expected behavior of all non WLI controls. The external controls are not frozen or suspended with the JPD.
CR298097	**The browsing feature does not work for JPDs.** **Workaround**: In the EAR deployment mode, the test browser allows you to browse inside the war file. However, you cannot open or read the actual files. In the split source mode, no files are displayed as the war is empty. It is not possible to reset the status of a process from within the test browser.
CR296136	**In WLI9.2, any redelivery limit greater than zero is lost when a message is either frozen, received while a process is frozen or suspended, or simply received out-of-order. This may defeat attempts by a process author to handle.** Under certain circumstances the message may be retried more than original redelivery limit due to the limited information available from the JMS subsystem about JMS messages being processed.

Problem ID	Description
CR292400	**In WLI9.2, while generating a Process Control and Fronting JWS, empty conversation finish method is generated.** **Workaround**: Works as designed. The reason, the conversation is seen when the Process Control is generated is due to the fact that there is additional node (perform) after the `clientRequestWithReturn` node. It is recommended to create other client visible nodes (conversation continue) after the `clientRequestWithReturn` node and then start the conversation annotation on the process control. Remove the perform node after the `clientRequestWithReturn` and the conversation start annotation on the process control is not seen. When the process control has a conversation annotation, the generated JWS will have a place holder conversationFinish method giving the client ability to finish the conversation. In this case, the client must call the conversationFinish to finish the conversation. This restriction will not be present once the JWS front end feature is enhanced so that the conversation on the JWS is finished implicitly when the JPD conversation completes.
CR287671	**The package names of MFL xbean types are derived from MFL file paths, relative to the schemas directory.** For example, types for an MFL file `foo/BAR.mfl` would live in a package foo.bar. But, if the MFL file contains an alphabet which occurs after a number, case of the alphabet changes to upper case. Example: a file `/foo/foo2bar.mfl` will result in a package name of `foo.foo2Bar` This affects upgrade scenarios where the user has: • MFL files named in such a manner and • files (e.g. processes) which reference xbean types generated from the MFL files. **Workaround**: You need to update the references to use the 9.2 style package names.
CR284937	**Test console Displays Comments in the JPD** When you use the test console to test an application that you have upgraded from 8.1 to 9.2, you may see comments annotated as follows in the Test console: `"/** * @jpd:process process:: ....."` You can ignore these comments—the upgraded JPD contains the original (8.1) process annotations as comments and these are displayed in 9.2.

Problem ID	Description
CR284905	**In the WLI 9.2 JPD, wrong icons are used for Service Control nodes.** **Workaround**: None.
CR279454	**Dynamically Propagate Changes in a JPD to the Control and WSDL files** Unlike in 8.1, changes made to a JPD in 9.2 are not automatically reflected in its corresponding Control or WSDL files. **Workaround**: If any changes are made in a JPD, you must regenerate the Control and WSDL files to reflect these changes.
CR275878	**Renaming the default Web Content folder (WebContent) in a Web Project.** Manually renaming only the default "WebContent" folder in a Web Project would cause problems when the project is compiled or run. This is because the new name does not reflect in all the settings files that reference it. **Workaround**: To successfully change the Web Content folder name, you have to ensure the new name is also recorded in the following documents: • {webProject}/.settings/.component • {webProject}/.settings/org.eclipse.jdt.pref
CR269354	**Warning message is displayed when JPDs are redployed.** A warning is issued instead of aborting the deployment of the web application. Error messages are displayed when the deployment is aborted. **Warning**: Undeploy the application and republish the JPD.
CR245676	**Eclipse problems viewer which lists all the diagnostics in a JPD is displayed in the Design view.** You can use the **Design** view to find the source of the error.
CR206655	**ProcessRuntimeMBean Failing to Retrieve Stateless Business Process Instance Information in Development Mode** The `ProcessRuntimeMBean` is not normally used for stateless business processes. It is usually used for stateful processes. In development mode, not production mode, on some processes, it may return the wrong information.
CR200072	**Uncommitted Local Transaction when Connection Returned to Connection Pool** The local transaction is left open. To close it, in the JDBCConnectionPoolMBean, set the property `RollbackLocalTxUponConnClose="true"`.

Problem ID	Description
CR155813	**Timeout Attribute not Supported on Transaction Blocks** You cannot specify a timeout property on an explicit transaction block in the WebLogic Workshop graphical design environment. This feature is deprecated in Service Pack 2. If a business process you created in an older version of WebLogic Integration contains a timeout property on a transaction block, a compiler warning is issued to indicate that the timeout value is ignored.
CR155294	**The Freeze on Failure Property is Ignored for Explicit Transactions** The `freeze on failure` property for explicit transactions is ignored. That is, if you set the `freeze on failure` property on an explicit transaction block, it is ignored at run time. When this property is set on the start nodes in a business process, it operates correctly and as documented. This feature is deprecated in Service Pack 2. If a business process you created in an older version of WebLogic Integration contains a `freeze on failure` property on a transaction block, a compiler warning is issued to indicate that the `freeze on failure` value is ignored.

CR154487	**Use of @jpd:unexpected-message Annotation**
	Business processes often include nodes such as **Control Receive** or **Client Request**, at which the process waits for delivery of an expected message before continuing. By default, messages that arrive before they are expected—that is, before the process encounters the **Control Receive** or **Client Request** node in question—are automatically buffered and are delivered later when the process is ready to receive them. In some cases the process designer may wish to discard any such early, *unexpected* messages; this enables the process to ignore messages that arrived earlier yet are no longer relevant to the process. The `jpd:unexpected-message` annotation gives process designers the ability to control this behavior on a node-by-node basis. The annotation is available for **Control Receive** nodes and **Client Request** nodes in positions other than the Start node. The annotation can be set by switching to the Source view, clicking on the corresponding node's method header, consulting the unexpected-message header in the **Property Editor**, and setting the `action` property from `save` (the default) to `discard`. The annotation can also be set using the annotation editor just like all the method annotations. The annotation is placed into the JPD source code, as shown in the following code segment:
	`/**`
	`* @jpd:unexpected-message action="discard"`
	`*/`

Problem ID	Description
CR145540	**Behavior of rename-old Attribute for the File Control** If you use a File control for which the suffix-name or suffix-type attributes are not specified, but for which the create-mode attribute specifies rename-old, the create-mode attribute specification is not honored. In other words, the older file is not renamed; instead it is overwritten. **Workaround**: If the create-mode attribute specifies rename-old, you must specify a value for the suffix-name.
CR138164	**How to View More Events in the Test Browser** If you want to see more than 30 events (the default) in the Test Browser, before running the business process, select the **Console** tab of the Test Browser and enter a larger number in the **Keep last** *number* **message** field, where *number* represents the number of messages to be displayed.

Cluster Configurations

Problem ID	Description

CR182788	**After performing JTA and JMS migrations from a failed managed server, a process instance may remain in running mode.**
	A repeating set of error messages similar to the following appears in `db2diag.log`:
	`Instance:DB2 Node:000`
	`PID:3888(db2syscs.exe) TID:5400 Appid:/1721616.3366.00090031001F data protection sqlpxTEntrySwitchIn Probe:300 Database:DBNAME DIA8036C XA error with request type of "". Transaction was not found. ZRC=0x80100024.`

CR156912	**Deploying an EAR in Cluster Configurations Causes Certain Warnings That Can Be Ignored** Specifying a cluster name as the target when you deploy an EAR file causes BEA-149055 warnings to appear in the WebLogic Server console window for the WebLogic Server that hosts the WebLogic Server Administration console (WebLogic Administration Server). These warnings include the following text: `Having multiple individual servers of a cluster as targets instead of having the entire cluster as the target can result in non-optimal load balancing and scalability.` You can ignore these messages.

Controls

Problem ID	Description
CR3000356	**The change in the Workshop Time control behavior from 8.1 to 9.2 causes problem when the control is used from JPD.** **Workaround**: WliTimerControl is the default WebLogic Integration timer control for JPDs. There is limited support for the WLW Timer Control when used from a JPD.
CR295021	**9.2 Workshop Web Service Control does not support WSDL with a fault message and without any parts specified.** **Workaround**: Either remove the fault message or add atleast one part to the fault message.
CR292434	**Potential database deadlock when using Microsoft SQL.** You may encounter a database deadlock if you have configured your domain with Microsoft SQL. **Workaround**: After you have created your domain, add the `SendStringParametersAsUnicode=false` property to the data source scripts.
CR288927	**Issues using sendRequest in TIBCO Rendezvous™ Control with Certified Messaging** When using TIBCO Rendezvous Control with Certified Messaging, the sendRequest function can return a null response. This may also cause the server process to consume close to 100% CPU. **Workaround**: These issues are intermittent and do not have a solution or workaround. You can check the response message for a null value, and in the event of a 100% CPU utilization, restart the WebLogic Integration server.
CR287071	**No foreign JMS destinations visible while creating JMS Controls using the Wizard.** JMS Control and Workshop JMS Control creation wizards in Workshop the Destination fields do not show foreign JMS destinations. **Workaround**: Foreign JMS destinations can be added directly in the source code annotations.
CR272670	**Tuxedo Control is not supported in 9.2**

Problem ID	Description
CR181948	**http-xml Protocol Not Supported For Service Broker Control** If you use the Service Broker control to call processes, do not use `http-xml` attribute.
CR145542	**I\Overwrite Option Does Not Work When Suffix Type Is Set To Timestamp (File Control)** If you set the suffix attribute timestamp in the File control, the `create-mode=overwrite` attribute is not honored. Instead, the file is renamed.
CR145540	**File Override Behavior of the File Control** When using the file control, if no suffix is specified, the `create-mode="rename-old"` attribute is not honored. Instead the file is overwritten.
CR138481	**Specified Scheme, Server Name, or Port Number in Process Control Target is Ignored** The Process control target can be specified in several ways: • through the `jc:location` annotation on the jcx file • by invoking the `setTargetURI` or `setProperties` methods on the control • with dynamic properties This target location is relative to the application. It must not have a scheme, server name, or port number. If a scheme, server name, or port number is specified, it will be ignored.

Event Generators

Problem ID	Description
CR289678	**A SQL Exception can occur when you create an RDBMS Event Generator on a table that has LONG column(s) (Oracle Database).** **Workaround**: Change the table to use LOB instead of LONG.
CR288927	**When using TIBCO Rendezvous Control with Certified Messaging, at times the sendRequest function returns a null response. This may also cause the server process to consume close to 100% CPU.** **Workaround**: These issues are intermittent and do not have a real solution or workaround. You can check the response message for a null value, and in the event of a 100% CPU utilization, restart the WebLogic Integration server.
CR276975	**Creating a New TIBCORV Event Generator with a blank space through WebLogic Integration Administration Console throws javax.servlet.ServletException: jndi lookup failure for TIBCO Event Generator.** **Workaround**: Event Generator names should not have any spaces in between.
CR206549 CR206332	**Slow Down of Sybase Database Table or Microsoft SQL Table When Using RDBMS Event Generator** When a Trigger Type event is created on a Sybase Table or Microsoft SQL Table, the RDBMS Event Generator creates a Shadow table that is a replica or subset of the User's Table. The Trigger copies the Inserted, Deleted, and Updated rows from the User Table into the Shadow Table. As the rate at which rows are inserted, deleted, or updated into the User table increase beyond the rate at which the event generator can poll and process rows, the new rows become backlogged and cause the Poll query to slow down **Workaround**: The Shadow table gets created with the same name as the Channel Rule Definition name with an `_BEA_SDW` suffix. To speed up polling and hence processing, a Unique Index must be created on a particular column. For example, if the Trigger Event name (Channel Rule Definition name) is `hello123`, the Shadow Table gets created as `hello123_BEA_SDW`. To workaround this problem, create a Unique Index as follows: `CREATE UNIQUE INDEX [idx_hello123] ON schema_name.hello123_BEA_SDW(BEA_SEQ_ID)`

Problem ID	Description
CR206528	**View All Timer Event Generators Page Does Not Refresh Status** When using a Timer Event Generator, the **View All Timer Event Generators** page does not refresh the status from Suspend to Running after clicking Resume. **Workaround**: You must refresh the browser page to show the updated status.
CR206519	**Timer Event Generator May Not Reflect Changes in Business Calendar** After making a change to the Business Calendar, the Timer Event Generator may not pick up the change in the calendar. **Workaround**: In the **WebLogic Integration Administration Console**> **Event Generators** > **View All File Event Generators** window, select the appropriate Timer Event Generator, click **Suspend**, and then click **Resume**.
CR205996	**RDBMS Event Generator Trigger Not Deleted For Certain Trigger Type Events created on Microsoft SQL Server** When creating a Trigger Type Event on a schema other than with the DBO username, the SQL Server creates the Trigger in the same schema as the User Table on which it was created. If there is an attempt to insert rows after the Event has been deleted, errors are thrown because the Trigger still exists. Creating the Trigger does not require a schema name prefix but dropping the Trigger requires a schema to be prefixed to the statement. **Workaround**: After the Event is deleted, the Trigger must be dropped manually using a simple command `DROP TRIGGER <user_table_schema_name>.<trigger_name>`.
CR204272	**Enabling Auto Commit for RDBMS Event Generators when Working with Informix Databases** If you are working with an Informix database, you should always enable the auto commit mode during an Insert, Update, or Delete event.

Problem ID	Description
CR202902	**Automatic Delete for Query or Post Query Events in RDBMS Events** In the case of an Query or Post Query Events query with automatic delete (nothing specified in Post Query), the Max Rows Per Poll value is ignored. This was tried with values of 1 and 10 for Max Rows per poll. But in each case, all the rows were published and deleted in one single poll instead of 1 or 10 per poll as specified in the Channel rule definition. This case is specific to the Sybase database when using the Data Direct driver. In Oracle 9i DB with Oracle's 9i Thin Driver, the behavior was as expected and only 1 or 10 rows were published as specified in the channel rule definition. In DB2 and Informix, the automatic delete option is not supported.
CR201132	**Creating Triggers on a Sybase Database Table** If you are working with a Sybase database, when a Trigger Event is created on a Sybase table that already has one Trigger Event of the same type, such as Insert, configured, the older Trigger is replaced. This means that the older Trigger Event stops working. You can have one Insert, one Delete and one Update Trigger Event per Sybase Table.
CR200743	**Correct Data Not Published for Informix Database Triggers with Different Data Types** Due to the fact that Informix databases have peculiar ways of retrieving the Precision and Scale of `DECIMAL` and `MONEY` data types, it is recommended that you specify the Precision and Scale in the User Table - `DECIMAL(p,s)`.
CR200681	**Restrictions on Trigger Type Events Created on Informix Databases** Trigger type events created on Informix databases have some restrictions. The value specified in the **No. of Threads** field and **Max Rows Per Poll** field must always be 1 and the value specified in the **Polling Interval** field must be greater than the total time it takes to publish the rows picked up in every poll.
CR200495	**Creating New RDBMS Event Generator Channel Rule Definitions in Informix Databases** If you are working with an Informix database and you want to delete an existing channel rule and create a new channel rule with the same name as the one you just deleted, you must set the Statement Cache size to zero.

Problem ID	Description
CR196414	**File Event Generator FTP Supports Only Windows and UNIX Type FTP Servers** Only Windows and UNIX type FTP servers are supported by the File Event Generator FTP. VMS is not supported.
CR196088	**Using the RDBMS Event Generator in a Cluster** To use the RDBMS Event Generator in a cluster, the managed nodes must be created with specific IP addresses and port numbers. By default, the IP addresses of the managed nodes are null. If the default settings are used, events cannot be created in RDBMS Event Generator. IP Addresses and Port Numbers of the Managed Nodes must be provided. Additionally, the Administration server sends messages to the distributed queue, Because the distributed queue is available only on the managed nodes, the JNDI-scoped security for `wli.internal.egrdbms.queue` must have lookup access to the default group: `Everyone`. BEA recommends always using the default security policies for administrative and server resources.
CR186350	**Setting Event Generator Polling Interval for Configuration Changes** The polling interval of the Event Generators may be set to a user-defined interval by using the following Java system properties: • Email: `wli.eventgen.email.checkConfigDelay` • File: `wli.eventgen.file.checkConfigDelay` • Timer: `wli.eventgen.timer.checkConfigDelay` These properties may be set on the Java command that starts the server. Each property sets the number of milliseconds between polls. For example: `-Dwli.eventgen.timer.checkConfigDelay=30000` This sets the Timer Event Generator polling for configuration changes every 30 seconds. If not specified, the default polling interval of 10 seconds is used.
CR138802	**Retrieving or Filtering on Timer Event Generator Metadata with Channel Type of XML is Not Supported** Events published by the Timer event generator with channel type of XML do not contain metadata headers, consequently, filtering or retrieving events based on metadata is not supported at this time.

Problem ID	Description

Using the RDBMS Event Generator to Publish String or XML Type Channels

If you configure events to publish to a String or XML type channel, the published results will contain a RowState attribute similar to the following: `<TableRow wId:RowId="2" wld:RowState="Inserted">`. The RowState attribute will always be "Inserted", even if the event type is "Deleted" or "Update". You should ignore this attribute as it does not indicate that the Row was inserted.

RDBMS Event Generator Data Loss

If the WebLogic Workshop application containing the channels to which the RDBMS Event Generator is publishing events is undeployed while the event is publishing data, or if the JDBC Store/Database used for JMS fails, but the database on which the event is configured is still running, there is a possibility of some data being lost. To minimize this possibility, the Rows allocated Per Poll to each Processing Thread should be a small number (around 50 or lesser). This number can be calculated as follows; Max Rows Per Poll/No. Of Processing Threads = Rows Per Processing Thread.

RDBMS Event Generator Channel Rule Definition

When you are creating channel rule definitions in the WebLogic Integration Administration Console, it is recommended that you do not use the **Back** button if you want to resubmit the details on a page. You should always use the navigation links provided and create a new channel rule definition.

Data Transformation

Problem ID	Description
CR297754	**DND of Expression functions do not work on Linux** If you open an XQ file, select a target element in the Design view, select a function in the Expression Functions view, and attempt to DND a function to a valid location, the function is not applied. **Workaround**: You can drag and drop the function only when the mouse pointer icon changes to indicate a droppable mouse pointer.
CR297025	**Intermittent problems with mapping a source element to a target element.** Drag and Drop in the Linux IDE sometimes does not function when you try to map a source element to a target element. **Workaround**: Drag and drop the source element to the target element only when the mouse pointer icon changes to indicate a droppable mouse pointer.

CR287396	**Strings that include spaces are not allowed in the XQuery Condition Editors**
	In 8.1, you could create a condition in an XQuery Condition Editor that included a space in string values for the condition. However in 9.2, you must enclose such strings in single or double quotes.
	For example, in 8.1, a Decision Node with the following variable and string selected on the left and right hand sides of the builder, respectively, created a valid condition:
	(left hand side) variable: `'$requestXML/ns0:shipAddress/@state'`
	(right-hand side) string: **`New Jersey`**
	Condition Generated:
	`data($requestXML/ns0:shipAddress/@state) = “New Jersey”`
	Workaround: For 9.2, the XQuery Condition Editor requires that you enclose such strings in single or double quotes—use the following syntax for a string such as the one in the right-hand side of the condition builder example above.
	`‘New Jersey’` or `“New Jersey”`

Problem ID	Description
CR206671	**Transformation Exception for In-Flight Processes when Shutting Down and Restarting Server** If the server is shutdown while a business process transformation is running and restarted, a transformation exception is thrown. **Workaround**: Restart the business process.
CR206148	**Message Format Error While Using MFL Non-XML to XML Transformation Method in Format Builder** The delimiter of Group level causes the exception in Format Builder or run time. **Workaround**: Remove this delimiter.
CR182658 CR138588	**Casting is Limited or Unsupported Between Some XML Schema Types and Java Types in Transformations** During run time, the casting in transformations between the following types can be limited or unsupported: • Between different XML Schema types • Between Java types and XML Schema types For example, the casting is limited between the XML Schema type `xs:double` and XML Schema type `xs:integer`. The casting from a source `xs:double` to a target `xs:integer` in a transformation will be successful during run time if the source double value is equal to 8 but will fail if the source double value is equal to 8.5 or even 8.0. The casting between these XML Schema types is unlike the casting done between types in the Java or C language. The casting between the XML Schema type `xs:date` and the `java.util.Date` is unsupported and will fail during runtime because these two types are not equivalent. The XML Schema type `xs:date` contains only a date component and does not contain a time component while the `java.util.Date` Java class contains both a time and date component.

Trading Partner Integration

Problem ID	Description
CR276691	**The iterativeDevDisabled() call has been deprecated in WebLogic Integration 9.2.** **Workaround**: The `iterativeDevDisabled()` call has been replaced with the `productionModeEnabled()` call for Trade Partner Integration (Utils.java).
CR182302	**Bulk Loader Utility is Not Compatible with XA Database Drivers** Attempting to load data in the TPM repository with the Bulk Loader configured to use an XA database driver fails with the following error: `No suitable driver`. **Workaround**: Configure the Bulk Loader to use a non-XA driver, or load the data interactively using the WebLogic Integration Administration Console. For information about how to configure the Bulk Loader, see "Configuring the Bulk Loader Configuration File" in Using the Trading Partner Bulk Loader in *Managing WebLogic Integration Solutions* at the following URL: http://edocs.bea.com/wli/docs92/manage/bulkloader.html For information about how to load TPM data using the WebLogic Integration Administration Console, see "Importing Management Data" in Trading Partner Management in *Managing WebLogic Integration Solutions* at the following URL: `http://edocs.bea.com/wli/docs92/manage/tpm.html`
CR175845	**Extraneous Error When Deleting a Certificate** When deleting a certificate from the WebLogic Integration Administration Console, if you encounter the following error message, click Continue to dismiss it: `Certificate with name cert_name does not exist for partner partner_name.` Despite the error, when you click **Continue**, the certificate is deleted. **Workaround**: The error message can be disregarded.

Problem ID	Description
CR156555	**WebLogic Administration Console Generated Client Certificates May Not Work for Two Way SSL Testing** The "self-signed" client certificates that you can generate for testing purposes through the WebLogic Integration Console may not work for two way SSL configurations when client certificates are enforced on the server-side. **Workaround**: When you are testing two way SSL configurations, generate your test certificates by using other tools, such as OpenSSL.
CR155713	**DOCTYPE is Not Preserved in XQuery Transformations** XQuery transformation does not preserve the DOCTYPE element. **Workaround**: If you need the `DOCTYPE` element in further processing, add it back into your message by using the `obj.documentProperties().setDoctypeSystemId` in a Perform node following the transformation. An example of this is shown in the "Walkthrough of the Failure Notifier Business Process" section of the "Step 2: Open the PIP0A1: Notification of Failure Example" example under the "Tutorial Steps" heading of the Tutorial: Building RosettaNet Solutions available at the following URL: http://edocs.bea.com/wli/docs92/tptutorial/rosettanet.html
CR155685	Update Older Bulkloader XML Files when Using Signature Configurations This version of WebLogic Integration supports MD5, in addition to SHA1, as a digest algorithm option for RosettaNet. If you want to use signature configuration with older versions of bulkloader XML files, you need to add the following attribute to these files: • `signature-digest-algorithm="MD5"` valid values are `MD5, SHA-1 or NONE`. The `signature-digest-algorithm` attribute is optional. Its representation is a character string in the DBMS. If you do not specify a value when you import the older version of the XML file, the value is set to NONE.
CR155614	**Trading Partner Integration API Changes** The following APIs have changed: • `retrieveAllTradingPartner is now retrieveAllTradingPartners` • `retrieveAllAuthentication is now retrieveAllAuthentications` • `retrieveRosettaNetServiceBinding is now retrieveRosettaNetServiceDefaults`

Problem ID	Description
CR155423	**The ebXML Protocol Uses the Remote Trading Partner's Values for Retry Number, Retry Interval, and Persist Duration** When you are using the ebXML protocol for Trading Partner messaging, the values used for **Retry Number, Retry Interval,** and **Persist Duration** are always the values of the remote trading partner, not the local Trading Partner.
CR154862	**The Default Trading Partners have New Trading Partner IDs** The two default trading partners that are created when you create a new WebLogic Integration domain have new default trading partner ids, as shown in Table 2-1. If you use a new WebLogic Integration domain with any old application data, be sure to update any relevant files.
CR138262	**Using Controls to Send Messages from Participant Business Processes is Not Recommended** In WebLogic Integration, you use Trading Partner Integration controls to send messages from the *initiator* business process to the *participant* business process. However, in the *participant* business process it is recommended that you use Client Response nodes to handle outgoing business messages to the initiator. If you use controls in a *participant* business process, you may lose the message response signals, such as acknowledgments and error messages. If you need to use a control to send messages instead of using the recommended design pattern, place the control in a subprocess and invoke the subprocess from the *participant* process.

Table 2-1 Trading Partner IDs (CR154862)

Trading Partner	Old ID	New ID
`Test_TradingPartner_1`	`TP1-id`	000000001
`Test_TradingPartner_2`	`TP2-id`	000000002

Upgrading WebLogic Integration

Problem ID	Description
CR337692	**Deployment of a WLI application with Web Service controls fails** Deployment of a WLI application fails after upgrading the WLI application with Web Service controls from 8.1.x to 9.2 MP2. **Workaround**: Recreate the Web Service controls migrated to 9.2 MP2 and reference the newly created 9.2 MP2 controls in their upgraded processes to successfully deploy the upgraded application.
CR306344	If you upgrade a 9.2GA platform to 9.2 MP1 using the upgrade installer and then roll back to 9.2GA, sample domains in WebLogic Integration and WebLogic Portal will be unusable. Therefore, you must be very sure about upgrading to 9.2 MP1. In case you do upgrade and then decide to roll back, the sample domains will be unusable.
CR300337	`com.bea.control.annotations.MessageBuffer` must not be used on the callback operations of a Process control that is used to invoke a JPD from a JWS. Using the `com.bea.control.annotations.MessageBuffer` annotation will not work reliably in a cluster environment.
CR299867	**Timer control does not work in a cluster.** **Workaround**: When you upgrade an 8.1 application, the timer control is upgraded to the WebLogic Workshop timer control that does not work in cluster. You need to manually upgrade the application to the clustered version, if the application is to be used in clustered environment.
CR299266	**After application source upgrade, MFL Transformation failed with XQuery Exception (XQRLUserException).** In 9.2, MFL-derived XMLBeans belong to a namespace. In 8.x, XQuery transformations that use MFL-derived XMLBeans as output types must be manually updated to yield XML with elements in the appropriate namespace. The namespace used is determined by the path of the MFL file, relative to the schema source directory. For example, an MFL file located at `project/schemas/dir/purchase.mfl` will yield XMLBeans that belong to the XML namespace: `dir/purchase`.

Problem ID	Description
CR299154	**The Upgraded Work List Application throws a ControlException in the upgraded Integration domain.** **Workaround**: The upgrade does not set the security policy when the default authenticator is used. If you use an authorization provider other than the default LDAP provider, you need to either check or set this security policy. After you upgrade a WebLogic Integration 8.1 domain, you must set the security policies on the Compatibility 8.1.x Task Plan and allow the 'Anonymous' role in the Create Policy. Use the Worklist Administration Console (the default authorization provider) to set the Create Policy for the Compatibility 8.1.x task plan. If you are using a third-party authorizer, use the related third-party client tools to set the policy.
CR298097	**The Platform application upgrade from 8.1 SP5 to 9.2 failed while building the Schemas (XML Beans).** **Workaround**: None
CR296186	**In a COFACE Application, inline XQ upgrade does not convert XQ outside of Prolog.** **Workaround**: In the **Source** view, locate the xf: and replace with fn: for the xqueries not in prolog.
CR296260	**Process libraries like jpdpublic.jar are not added to build path when a 8.x Application with multiple web projects is upgraded.** **Workaround**: The dependent artifacts are not added in the project as there are no WLI related classes. Therefore, you need to add the libraries manually.

CR296186	**Inlined XQuery expressions that occur within process annotations are not updated after selecting the "Upgrade XQ2002 to XQ2004" option.**
	Even if the "Upgrade XQ2002 to XQ2004" option is selected in the Upgrade Wizard, some inlined XQuery expressions that occur within process annotations are not updated. Like, the usage of the xf: function prefix is supposed to be replaced by the fn: function prefix. So, for example:
	`<case name=\"Case\" value=\"xf:string($x)\"/>`
	should be:
	`<case name=\"Case\" value=\"fn:string($x)\"/>`
	Workaround: Manually change the function prefix in the source view.

Problem ID	Description
CR295456	**The upgrade process allows the synchronization receive 'method' to have a non-void return type.** The upgrade process does not check for the non-void return type, which leads to an compilation error. **Workaround**: You can detect this error by parsing the process XML and checking the return type of the message. WebLogic Integration issues a warning/error message that the 8.x source is invalid.
CR295684	**Aborting the upgrade process at the Upgrade Preview stage.** Clicking Cancel in the Upgrade Preview tab of the Upgrade Wizard will result in errors that may vary from one application to another. These are not application specific issues and can be ignored.
CR295612	**Exception after iterative development using Workshop for Eclipse, and using JWS with MessageBuffer.** A process application is created with JWS and a MessageBuffer annotation. The application is published from Workshop and tested. A few changes are made to the web application without republishing it from Workshop. Invoking either the JWS or a JPD from the test console results in an "Unable to deploy EJB..." error **Workaround**: Restart the server and publish the application after server restart to proceed.
CR294091	**Insufficient memory when archiving a large web application.** A `java.lang.OutOfMemoryException` is seen when archiving a very large web application using Workshop generated build scripts. **Workaround**: Add `memoryMaximumSize="1024m"` property to the `<apt>` task in the `build.xml` file of the web application.

Problem ID	Description
CR293073	**Workshop Web Service Control generated in 8.1 SP4 with types selected to Java for a complex type array with rpc/encoded mapped the complex type array to anyType[ ] instead of the concrete type array. For example:**

```
<schema targetNamespace="urn:serviciosAdminNE"
xmlns="http://www.w3.org/2001/XMLSchema";;>
<import
namespace="http://schemas.xmlsoap.org/soap/encoding/";
;/>
<complexType name="Aplicacion">
<sequence>
<element name="certificado" type="xsd:base64Binary"/>
<element name="descripcion" nillable="true"
type="xsd:string"/>
<element name="idAplicacion" nillable="true"
type="xsd:string"/>
<element name="nombre" nillable="true"
type="xsd:string"/>
<element name="organismo" nillable="true"
type="xsd:string"/>
</sequence>
</complexType>
<complexType name="AplicacionArray">
<sequence>
<element name="abonadoArray" nillable="true"
type="impl:ArrayOf_tns1_Aplicacion"/>
<element name="aplicacionArray" nillable="true"
type="impl:ArrayOf_tns1_Aplicacion"/>
</sequence>
</complexType>
</schema>
<schema
targetNamespace="http://notanot/jboss-net/services/Ser
vicioWEBAdminNE";;
xmlns="http://www.w3.org/2001/XMLSchema";;>
<import
namespace="http://schemas.xmlsoap.org/soap/encoding/";
;/>
<complexType name="ArrayOf_tns1_Aplicacion">
<complexContent>
<restriction base="soapenc:Array">
<attribute ref="soapenc:arrayType"
wsdl:arrayType="tns1:Aplicacion[]"/>
</restriction>
```

Problem ID	Description

	</complexContent>
	</complexType>
	</schema>
	The generated Workshop Web Service Control has:
	`public static class AplicacionArray implements java.io.Serializable { public anyType[] abonadoArray; public anyType[] aplicacionArray; } public static class anyType implements java.io.Serializable { private static final long serialVersionUID = 1L; public XmlObject[] t; }`
	`anyType[]` is not supported by 9.2 Workshop Web Service Control. Therefore, the above Service Control after upgrade will cause errors during deployment or application archive.
	Workaround: After upgrade change the types in the service control to point to the concrete type. For example:
	`public static class Aplicacion implements java.io.Serializable { public byte[] certificado; public java.lang.String descripcion; public java.lang.String idAplica ion; public java.lang.String nombre; public java.lang.String organismo; } public static class AplicacionArray implements java.io.Serializable { public Aplicacion[] abonadoArray; public Aplicacion[] aplicacionArray; }`

Problem ID	Description
CR292880	Upgraded applications with JPDs' subscribing to a WebLogic Integration message broker channel finish an incomplete compilation stage without reporting any errors. Subsequent execution of the application fails on the JPD subscription to a channel. This issue has been reported for test application "`e2ecm`" only. The solution to the problem is to add the missing library, in this case `p13n_app.jar`, to the application classpath.
CR292362	**JWS upgrade does not honor default values (default=0) for retryDelay and retryCount attributes of annotations common:message-buffer and jws:message-buffer.** `Workaround`: After the application has been upgraded, manually `set retryCount=0` and `retryDelay=0` to the `weblogic.jws.MessageBuffer` annotation.
CR291057	**ebXML application using ebXMLControl within java custom control results in a Null Pointer Exception when ControlHandle.sendEvent is not used.** **Workaround**: Read the note and example to *JPD and Control Callback* section of the WebLogic Integration 9.2 Upgrade Guide. The information is available at http://edocs.bea.com/wli/docs92/upgrade/component.html.
CR290904	**Upgraded XQuery files corresponding to transform functions returning XMLObject may fail to execute at runtime.** **Workaround**: Modify the XQuery function signature to specify a return value with multiple occurrence.
CR290169	**Interface WSDLs are not supported by Workshop for WebLogic Platform 9.2 Web Service Control.** This will result in an application deployment error if BPEL Import results in the generation of Workshop Web Service Control for an interface WSDL. **Workaround**: Add dummy implementation elements - <service> and <binding> - to the WSDL after BPEL Import.

Problem ID	Description
CR288918	**Upgrading an 8.x application that contains Schemas and adding a prefix in the Workshop for Weblogic Platform Upgrade Wizard.** If you add a prefix to an 8.x application with Schemas from the Upgrade Wizard, it will result in an error. This is because the upgrade utility looks for the Schemas.jar file, with the specified prefix, in the 8.x application directory. No such file exists as only the Schemas.jar file is present in the 8.x application directory. **Workaround**: Avoid adding a prefix in the Upgrade Wizard or manually delete the `<EarProject>\EarContent\APP-INF\lib\Schemas.jar` file.
CR288777	**Error upgrading B2B application as a result of incompatible package reference in a JPD and the Control file.** B2B application upgrade fails when EnvelopeDocument is being referred from two different packages in the JPD and the Control file. JPD refers to EnvelopeDocument from `org.xmlsoap.schemas.soap.envelope` while the Control file refers to `weblogic.wsee.jws.wlw.schemas.soap11` package. **Workaround**: Replace the package of EnvelopeDocument from `weblogic.wsee.jws.wlw.schemas.soap11` to `org.xmlsoap.schemas.soap.envelope` in the ebXML control files.
CR287857	**BPEL Import generates Workshop Web Service Control for external services specified as partnerlinks. Workshop Web Service Control does not successfully resolve external schemas imported using relative URIs into the WSDL (associated with the Service Control).** **Workaround**: Copy the external schemas from the utility project to the web project. Subsequently, place the external schemas relative to the location of the WSDL in the web project.
CR287681	**SOAP for a schema element defined in schema with `"elementFormDefault=unqualified"` is incorrect.** When elements from schema with `elementFormDefault="unqualified"` are used as parameters or return types to a Workshop Web Service Control, which is generated with XML Beans binding, the SOAP message generated by the Workshop Web Service Control is incorrect. **Workaround**: Use `elementFormDefault="qualified"` in the schema. When creating a new schema in Workshop add `elementFormDefault="qualified"` to the schema (default value for `elementFormDefault="unqualified"`).

Problem ID	Description
CR287361	**Suspended external event generators are not upgraded during domain upgrade.** The WebLogic Integration Domain Upgrade tool upgrades only *running* event generators. In other words, if you have any event generator in a *suspended* state, they are not upgraded. **Workaround**: Ensure that no event generators are suspended before you upgrade your application.
CR287291	**Upgraded 8.1 applications display compilation errors in JPDs' and transformation classes.** Applications which use `log4j` classes, especially in an EJB project, require the `log4j.jar` file to build. The location of the `log4j.jar` file has changed from 8.x to the 9.2 release and needs to be specified manually. **Workaround**: After upgrading the project, manually add the `log4j.jar` file to the project build path.
CR285560	**MBCS (multi-byte character set) characters in source files are not supported during upgrade, and appear as garbled text after upgrade.**
CR290852 CR285770 CR282814 CR280420	**Inter-operability issues with WebLogic Integration and other WebLogic products.** Install the following patches in a non-WebLogic Integration 9.2 installation. • Patch ID S35L • Patch ID XE28 • Patch ID 9Z62 These patch have to be installed on a WebLogic Server or Workshop for WebLogic Platform installation to ensure seamless inter-operability with WebLogic Integration 9.2.
CR282449	**XmlObject[ ] is not supported as a parameter for a JWS callback operation.**
CR282428	**In 9.2, `weblogic.jws.Types` annotation for JWS does not support array types.** 8.x JWS/Workshop Web Service Control with `jws:parameter-xml/jws:return-xml` annotation's include-java-types having an array type defined as a value will not upgrade successfully to 9.2. **Workaround**: Re-design the JWS/Workshop Web Service control before you begin to upgrade.

Problem ID	Description

CR281279	**JAX-RPC (Tylar) binding does not support schemas that reference xsd:schema element.**
	As shown in the following code sample, this lack of support effects generation of Workshop Web Service Control/JWS (using JAX-RPC binding) from a WSDL with references to `xsd:schema` element.
	`<xsd:complexType>`
	`<xsd:sequence>`
	`<xsd:element ref="xsd:schema"/>`
	`<xsd:any/>`
	`</xsd:sequence>`
	`</xsd:complexType>`
	Workaround: Assign a name to the element and use the `xsd:anyType`, as shown below.
	`<xsd:complexType>`
	`<xsd:sequence>`
	`<xsd:element name="schema" type="xsd:anyType"/>`
	`<xsd:any/>`
	`</xsd:sequence>`
	`</xsd:complexType>`

CR280929	**Problems importing zip archived Workshop for WebLogic Platform projects.** You might encounter exceptions when trying to import a Workshop for WebLogic Platform project, that is in a zip archived file. **Workaround**: Unzip the contents of the archived file, and then import the project.
CR280701	**ArrayIndexofBoundException recorded in the log file during an application upgrade.** During the 8.x application upgrade to 9.2, the ArrayIndexofBoundException might be recorded in the `WORKSPACE_ROOT/.metadata/.log` file. This is a non fatal error that could arise if `BEA_HOME/weblogic92/server/lib/webserviceclient.jar` is on the build path of the upgraded application.

Problem ID	Description
CR279416	**JWS with callback having same JMS and HTTP transport URIs, will yield a compilation error after upgrade.** **Workaround**: Before compiling the upgraded JWS, ensure the URIs are unique.
CR275039	**`weblogic.jws.Types` annotation is not supported on Callback operations of a JWS.** **Workaround**: Refactor the service to not use the `weblogic.jws.Types` annotation.
CR275002	**A JWS with w3c.dom.* (Document, Element and DocumentFragment) as parameter or return type is not supported.** **Workaround**: Operation signature and return types of JWS that used `w3c.dom` types will have to be changed to use SAAJ types. The internal implementation of the operations will have to be updated to translate to or from SAAJ and `w3c.dom` types.
CR274286	**An upgraded JWS that returned an XMLBean type will not compile.** All operations on an upgraded JWS that returned an XMLBean type will have a compile error post upgrade. The message indicates that the return type doesn't match the element name defined by the XMLBean's schema. **Workaround**: Remove the `@WebResult` annotation or change the name to match the schema type.
CR278519	**There are two known issues in wftracking/M2_ArchHelloAsync.java file and Process_Tracking_Binary.java. Remove throws Exception from helloDelay_onTimeout methods.** After the issues were resolved the issues, the following errors were seen in the Problem window. 1. The project was not built since its build path is incomplete. Cannot find the class file for `weblogic.xml.xmlnode.XMLNode`. Fix the build path then try building this project (wfTrackingWeb) 2. The type `weblogic.xml.xmlnode.XMLNode` cannot be resolved. It is indirectly referenced from required .class files. **Workaround**: Add `weblogic.jar` to the external jars list in the build path to resolve the above issues.

Problem ID	Description
CR277695	**While executing an 8.x application in WebLogic Integration, comment strings are displayed in the test browser console.** **Workaround**: Works as designed. The Upgraded JPD will have comments so they are shown on test browser. Remove the comments after you the upgrade.
CR273709	**Workshop for WebLogic Platform 9.2 Web Service Control generation is not supported from an 8.x WSDL containing callbacks.** **Workaround**: If the 8.x WSDL represents an 8.x JWS, perform the following: 1. Upgrade the 8.x JWS to 9.2. 2. Generate WSDL from the 9.2 JWS. 3. Generate the 9.2 Workshop Web Service Control. If the 8.x WSDL represents an 8.x JPD, perform the following steps: 1. Upgrade the 8.x JPD to 9.2. 2. Front end the JPD with a 9.2 JWS (generate Process Control from 9.2 JPD and then generate Fronting JWS from Process Control). 3. Generate WSDL from the 9.2 JWS. 4. Create 9.2 Workshop Web Service Control from the WSDL generated in the previous step.
CR265847	**Upgrade required for Clients that use SOAP 1.2 messaging format.** **Workaround**: The WSDL for the Web Service changes during the upgrade to WebLogic Integration 9.2. To conform to the new specification, you need to regenerate the clients of Web Services that use the SOAP 1.2 messaging format
CR262360	**Workshop for WebLogic Platform 9.2 Web Service Controls do not support receiving 8.x-style callback messages from an 8.x style end point service.** If your 8.x application contains the use case of a JWS (caller) using the Workshop Web Service Control (caller) to invoke and receive a callback from another JPD (callee), it will not function correctly after upgrade. **Workaround**: Upgrade both the callee (JPD) and the caller (JWS and Workshop Web Service and execute the following steps: 1. Front end the callee JPD with a 9.2 JWS 2. Regenerate 9.2 Workshop Web Service Control from the WSDL for the 9.2 JWS 3. Use the 9.2 Workshop Web Service Control generated in step (2) from the caller JWS

Problem ID	Description
CR262359	**Workshop for WebLogic Platform 9.2 Web Service Controls do not support receiving 8.x-style callback messages from an 8.x style end point service.** If your 8.x application contains the use case of a JPD (caller) using the Workshop Web Service Control (caller) to invoke and receive a callback from another JPD (callee), it will not function correctly after upgrade. **Workaround**: It is recommended that you upgrade both the callee (JPD) and the caller (JPD and Workshop Web Service and execute the following steps: 1. Front end the callee JPD with a 9.2 JWS 2. Regenerate 9.2 Workshop Web Service Control from the WSDL for the 9.2 JWS 3. Use the 9.2 Workshop Web Service Control generated in step (2) from the caller JPD If you have upgraded only the caller JPD and Workshop Web Service Control, execute the following steps: 1. Generate 9.2 Service Broker Control from the WSDL for the callee JPD 2. Use the 9.2 Service Broker Control instead of the upgraded Workshop Web Service Control from the caller JPD.
CR240167	**JMS control does not process incoming messages asynchronously in 9.2** In 8.x, a JPD could use a JMS Control to process incoming messages asynchronously. This is not supported in 9.2 and will cause the JPD to fail. **Workaround**: Replace the JMS Control with the WebLogic Integration JMS Control to process incoming messages asynchronously.
CR235276	**Use of non-SOAP XML message format over HTTP or JMS protocol is not supported.** In 8.x releases, the non-SOAP XML message format was supported over the HTTP or JMS protocol for JWS. In Workshop for WebLogic Platform 9.2, only the SOAP protocol is supported. So, an upgraded JWS front ending JPD will not function. **Workaround**: Remove the JWS front ending JPD and create a HTTP or a JMS event generator to receive the non-SOAP XML message. Subsequently, create a message broker subscription in the JPD to receive non-SOAP XML messages, from the recently created HTTP or JMS event generator.

CHAPTER 3

Problems Fixed in 9.2 MP3

The following table lists selected problems fixed in BEA WebLogic Integration 9.2 MP3, including a Change Request (CR) number for each problem.

For more information about the known limitations in WebLogic Integration 9.2 MP3, see "Known Limitations" on page 2-1.

Table 3-1 Problems Fixed in BEA WebLogic Integration 9.2 MP3

Change Request Number	Description
CR325707	After a graceful shutdown and restart, the parent process used to remain in a running state in the Process Monitoring Console. This problem occurred because internal response messages between parent and child processes were missing. This problem has been resolved.
CR328505	The `build.xml` file, generated by the Eclipse IDE, used to fail because all the fields were not marshalled properly at application build time. This problem has been resolved.
CR340743	Prior version was getting triggered when a message was published to a message broker channel on which the active process version was listening. This problem has been resolved.

Table 3-1 Problems Fixed in BEA WebLogic Integration 9.2 MP3

Change Request Number	Description
CR320124	In a debug mode, an AssertionError was returned when a WLI Java Message Service (JMS) Control was used through a Java Custom Control. This problem has been resolved.
CR326248	When creating an RDBMS Event Generator, an incomplete list of queue connection factories was displayed if a DNS name was used as the listen address of the domain. This problem has been resolved.
CR308667	When creating new tasks by using the worklist portal or the `RemoteWorklistTaskAdminBean` from a Java client, the task creation was not using the default settings for Description, Owner, Task Time Estimate and Task Completion Due Date specified in the Task Plan. This problem has been resolved.
CR312699	Incorrect method signature was being displayed for CustomCallback. The eventSet was referring to `TaskControl.Callback.class` instead of `TaskControl.CustomCallback.class`. This problem has been resolved.
CR321191	When publishing a WebLogic Integration application to the server, the application takes a long time to deploy. This problem has been resolved.
CR324723	WebLogic Workshop IDE was taking a long time to open JPD files and other WebLogic Integration artifacts. This problem has been resolved.
CR332638	When using WebLogic Integration 9.2 MP2 Event Generator for DB2, an invalid SQL statement used to return `COM.ibm.db2.jdbc.DB2Exception`. This problem has been resolved.
CR335705	Creation of JPD proxies by using JpdProxyTask used to fail with `FileNotFoundException` if the JPD package name contained "jpd". This problem has been resolved.

Table 3-1 Problems Fixed in BEA WebLogic Integration 9.2 MP3

Change Request Number	Description
CR353521	Timeout for HTTP control socket (HTTP Control method `setTimeout(timeInMillis)`) did not work. This problem has been resolved.
CR353593	When the same Dynamic Transformation control was called within two different transactions, the second transaction used to fail. This problem has been resolved.
CR342013	Applications were attempting to reuse an ebXML control within the same conversation and use the JPD to send several ebXML messages in a row. This problem has been resolved.
CR350770	The exception '`java.lang.RuntimeException: Protocol 'http-soap' not available on this operation`' used to be returned when a JPD was called. This used to occur in a setup with a JPD and a web service control calling the web services method. This problem has been resolved.
CR333587	`RDBMSConnMDB.markPolledRows()` was being overwritten by DB2ConnMDB and this caused `preparedStatement.setObject` to receive an incorrect parameter. This problem has been resolved.
CR341835	Length of WebLogic Integration message ID of the ebXML message was more than the permitted maximum length (128 bytes). This problem has been resolved.
CR348100	When a WebLogic Server EJB client was attempting to retrieve Task data using the Worklist API, the `getTask(<taskid>, <boolean (true)>)` method used to fail with unmarshall exception. This problem has been resolved.
CR356219	When a JMS client program sent a message to the queue, the message was received by the new process version of JPD and used to fail with an error message, even though the message had the correlation ID as the process instance ID of the older process version. This problem has been resolved.

Table 3-1 Problems Fixed in BEA WebLogic Integration 9.2 MP3

Change Request Number	Description
CR356402	The T3S protocol could not be used as the domain-wide administration protocol to secure clusters because the `DomainMbean.AdministrationProtocol` value was not being used to set the `defProtocol` value. This problem has been resolved.
CR356404	When undeploying an application in a WebLogic Integration cluster environment, the `DescriptorUpdateFailedException` used to be returned. This problem has been resolved.
CR356540	After using a JPD to create an array of Java objects and transform them into an XML document, the Java object was not being released. This problem occurred due to memory leaks during the transformation. This problem has been resolved.
CR341031	When a custom control has the same method name as in the JMS Control but a different number of arguments, `java.lang.ArrayIndexOutOfBoundsException` used to be returned. This problem occurred because the control container was retrieving the number of arguments from the base control method and not the extended method. This problem has been resolved.
CR341320	Annotation overriding was not working for the control target endpoint of the web service. This problem has been resolved.
CR349809	Worklist Tasks could not be created when the Admin Server was shut down. This problem occurred because, to post the task event to any listener, the JMX used to retrieve the Worklist event JMS destination from the Admin Server instead of using the MBeans of the local managed server. This problem has been resolved.
CR351020	The query for completed 8.1.x compatible tasks was incorrectly looking for a current **Step Name** of completed, instead of the **Administrative State** of **Completed**. This problem has been resolved.