



BEA WebLogic Integration™

Release Notes

Version 8.1
July 2003

Copyright

Copyright © 2003 BEA Systems, Inc. All Rights Reserved.

Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks or Service Marks

BEA, Jolt, Tuxedo, and WebLogic are registered trademarks of BEA Systems, Inc. BEA Builder, BEA Campaign Manager for WebLogic, BEA eLink, BEA Liquid Data for WebLogic, BEA Manager, BEA WebLogic Commerce Server, BEA WebLogic Enterprise, BEA WebLogic Enterprise Platform, BEA WebLogic Express, BEA WebLogic Integration, BEA WebLogic Personalization Server, BEA WebLogic Platform, BEA WebLogic Portal, BEA WebLogic Server, BEA WebLogic Workshop and How Business Becomes E-Business are trademarks of BEA Systems, Inc.

All other trademarks are the property of their respective companies.

Contents

About This Release of BEA WebLogic Integration	2
What Is New and Improved in This Release	4
New Features	4
Integrated e-docs and dev2dev	4
Platform Support and System Requirements	4
Adapters	4
Tutorials	5
Business Process and Data Transformation Tutorials.	5
Tutorials for Trading Partner Integration—Coming Soon	6
Documentation Additions	7
Known Limitations	7
Administration And Configuration8	
Running Business Processes	8
Restrictions on the Use of Message Ids.	8
Updated Process Information in the WebLogic Integration Administration	
Console, Must Build Application in WebLogic Workshop	9
WebLogic Integration Resources Require Process Projects	9
Process Authorization Policies Are Reset Upon Redeploying From Workshop or	
When A Deployment Mode WebLogic Server is Restarted.	9
Business Process Appears Active in Console Even If It Failed At Startup with	
Security Exception	10
Netscape 7.1 Not Supported For WebLogic Platform, Adobe SVG Viewer 3.0 Not	
Supported on Netscape 7.1	10

Application Integration	10
Directory Structure Change for Application View EJB	10
Dependent Application Views Not Suspended When Adapter Instance is Suspended 11	
Increasing the Transaction Timeout Set by Default	11
Auto-Suspend Fails If Calling Process Does Not Have Administrator Privileges	12
Invoking Service in Sample Application Against Oracle Database Results in Exception	12
Always Edit Application View or Service Connection Security Policy Using the WebLogic Integration Administration Console	13
Security Policy Settings Must Be Edited In Specific Order	13
Changes to Adapter Instance for Service Connection and to Connection Factory Not Applied	13
Cannot Change Auto-Suspend Settings for Adapter Instance Using Console . . .	14
Issues with DBMS LOCALTX Adapters	14
JTA XAER_PROTO Exceptions Can Be Thrown When the Service Connection Connection Pool Has No Free Connections	15
Business Process	15
How to View More Events in the Test Browser	15
Synchronous Sub-Processes Invoked From a JWS, Page Flow, or Business Control, Will Lose Tracking Data	15
The freeze on failure Property Ignored for Explicit Transactions	15
Use of the jpd:unexpected-message Annotation	16
When Archiving Business Processes Using a XA Driver, Must Set SupportsLocalTransaction Equal to True	16
Process Graph May Use Service Control Icon for Service Broker Control Nodes	16
Control Send Node in a Business Process Invoking a WebLogic Workshop Web Service Which Invokes a Method on a Stateful or Stateless EJB May Fail .	17

Cluster Configurations	17
Using the Configuration Wizard to Modify Cluster Configurations.	17
Using the Configuration Wizard to Modify Cluster Trading Partner Integration Configurations	18
In Cluster Configurations Using a Non-XA Oracle Driver, Certain JTA Exceptions May Be Thrown, Can Be Ignored	19
In Cluster Configurations, The ClusterAddress Attribute Must be Set for WebLogic Integration Clusters	19
In Cluster Configurations, Restarting a Managed Server May Fail If Admin Server Is Shutdown	19
Steps for Manual JMS Server Migration from a Failed Managed Server.	19
Using Oracle XA Driver, OracleXAException May Occur When Recovering a Failed Server	21
Controls	21
Empty Directory Name for Append Operation Results in Exception (File Control) 21	
File Override Behavior of the File Control	21
Overwrite Option Does Not Work When Suffix Type Is Set To Timestamp (File Control)	21
When Adding A Control Factory, the First Argument of a Callback Handler Must be the Same Type As the Control.	21
Generated Service Broker and Service Controls Have the Same Name.	22
Service Broker Control Cannot Be Created For a Business Process Not in the Current Application	22
Parent Process Not Notified of Failed Call to Subprocess (Process Control). ...	22
Multiple Channel Definitions For the Same Channel Name Are Allowed.	22
Need to Make Edits for deadletter Channels To Show Up in the Node Builder When Defining a Static Channel Subscription.	23
Do Not Associate a Single XA Connection Pool with Different Datasources ...	23

Data Transformation	24
Test XML Generation for XML Schemas With Choice Groups Not Supported	24
Mapper Does Not Display All the Array Inputs and Outputs of a Java Class	24
Java Inner Classes of JPD or JWS Files Cannot Be Used In Controls or Transformations	24
Validation Not Supported for XML Schemas that Contain Substitution Groups	25
Certain XQuery Keywords Are Not Allowed in XPath Expressions	25
How to Regenerate Test Data Once the Data Has Been Modified or Deleted	26
Automatic Promotion of XmlObject Types with XSLTs Not Supported	26
Mapping Differs Between xsd:anyType to xsd:any Elements	26
Deviations from the W3C XQuery August 2002 Draft Specifications	27
To Use Test View of Mapper, Must Use Default WLS Port of 7001	27
Database	27
To Change the Default Pointbase Port	28
Intermittent Pointbase Lock Time Out Reported When Message Size is Greater than the Configured Doc Store Size	28
For Oracle, Conversation State Tables Should Store BLOB Data in Tablespace Other than USERS	28
For Pointbase, If a Process Variable Exceeds 4 Mb, All Conversations Are Terminated In the Process	29
Trading Partner Integration	29
Creating a New RosettaNet Participant, Generates Callbacks Using the Long Form, Should Use Short Form	29
In Cluster Configurations, Updates to the TPM Configuration Require a Restart of the Managed Servers	30
ebXML Service Name Cannot Contain Double Byte Characters	30
For Trading Partners Representing WLI Business Connect, Only One Version of the ebXML Protocol Can Be Used At One Time	30

Persist Duration Attribute For ebXML	30
For Cluster Configurations, in the WebLogic Integration Console, the Trading Partner Service Information is Not Always Displayed	31
While Modifying Trading Partner Management Information, Wait For the Transaction To Complete On the Server, Before Modifying Additional TPM Information.	31
ebXML Service Name and RosettaNet PIP Name Must be Unique Among Participant Processes	31
Trading Partner Integration Messages Must Not Have A Null Payload	31
Do Not Start Participant Processes From the Test Browser	31
Do Not Use the Attributes: to-selector and from-selector at the Control Level in ebXML and RosettaNet JCX Files.	32
Cannot Specify Values Over 24 Days in ebXML or RosettaNet Bindings.	32
Generated Demo Certificate on Solaris May Have Invalid Characters and May Fail to Reimport	32
Run Time	32
XML Document Field Size Limitation	33

BEA WebLogic Integration Release Notes

BEA WebLogic Integration 8.1: Release Notes **Date: July 2003**

This document includes the following topics:

- [About This Release of BEA WebLogic Integration](#)
- [What Is New and Improved in This Release](#)
- [Platform Support and System Requirements](#)
- [Adapters](#)
- [Documentation Additions](#)
- [Known Limitations](#)

For updated release note information, go to the online BEA Platform 8.1 Release Notes available at the following URL:

<http://edocs.bea.com/platform/docs81/interm/relnotes.html>

About This Release of BEA WebLogic Integration

With WebLogic Integration 8.1, BEA Systems is driving the convergence of application integration and development platforms to a new level. WebLogic Integration is a single, unified platform for addressing the complete spectrum of business integration projects, including enterprise application integration, trading partner integration, business process management, and custom application development. This new release delivers dramatic gains in productivity and flexibility over traditional integration products, and fully leverages the powerful, enterprise-class scalability, reliability, and performance that BEA customers have come to expect from WebLogic Server. By converging all development activities across the WebLogic Platform into the WebLogic Workshop IDE, BEA provides one integrated, consistent, easy-to-use environment for building, testing, and deploying integration solutions and custom applications.

WebLogic Integration 8.1 supports end-to-end business integration through the following capabilities:

- **Business Process Management**—WebLogic Integration business processes can now be developed in the WebLogic Workshop Integrated Development Environment (IDE) using the two-way visual/source development environment to create business processes using a simplified process language.
- **Worklist**—WebLogic Integration delivers rapid access to integration with business users through the Worklist system. The Worklist system supports capabilities for managing users, groups, and roles, and for managing the routing of tasks to the people in an enterprise. It enables people to collaborate in the various tasks associated with business processes, including assigning tasks, tracking the status of tasks, handling approvals, and so on. Integral to the flow of work are actions such as receiving, approving, modifying, and routing documents. The documents that often accompany work activities provide the background required for people to complete tasks, which are the central component of all Worklist systems.
- **Trading Partner Integration** (Formerly business-to-business or B2B)
 - **Protocol support**—Support for ebXML Message Service 1.0 and 2.0, and RosettaNet 1.1 and 2.0.
 - **Message Tracking**—New ability to track on-the-wire messages to and from trading partners, link the messages to the BPM process instance, and archive the messages in the archive database. All these tasks can be performed through a Web-based administrative console that is integrated with the WebLogic Integration Administration Console.

- **Improved security**—Support for XML digital signatures, authentication (of messages and trading partners), authorization (of transport and business process), security-specific identity, secure audit log Service Provider Interface (SPI) (for all supported protocols), secure timestamp service and SPI (for all supported protocols), certificate verification SPI, non-repudiation (for all trading partner integration protocols), and key management.
- **Trading Partner Management (TPM)**—Improvements include support for extensible trading partner profiles, simplified access to TPM information, certification management, the ability to import and export TPM information, programmatically, on a running server, and integration of the TPM in existing profile stores, such as LDAP or RDBMS.
- **Data transformation**—A new data transformation mapper within the WebLogic Workshop IDE enables the definition of XQuery-based transformations using XML, non-XML, and Java data to provide an any-to-any mapping capability. XSL-based transformations continue to be supported in the run time.
- **Message Broker**—A new channel-based publish and subscribe message broker enables anonymous publish/subscribe communication between business processes and from external sources via the use of Event Generators.
- **Web Services** —Native Web Services support is provided for WebLogic Integration, including exposing business processes as Web Services, and invocation of Web Services from business processes using the Service control.
- **Application Integration**—Full integration of third-party applications via adapters configured through the Application Integration Design Console, and exposed to processes via the application view control and the Message Broker.
- **Administration**—A new Web-based management console for all WebLogic Integration management tasks: process, message broker, application integration, trading partner integration, and worklist.
- **Security**—WebLogic Integration supports the new WebLogic Server security model supported by WebLogic Server 7.0 and 8.1.

What Is New and Improved in This Release

- [New Features](#)
- [Integrated e-docs and dev2dev](#)

New Features

BEA Platform merges all of the features of the existing BEA WebLogic products—application server, Web service development, portal, and integration—into a highly integrated solution. This integrated solution combines the benefits of a common application infrastructure with an easy-to-use, robust, framework.

For a comprehensive list of new and enhanced features for the WebLogic Integration 8.1 Release, see the following URL:

<http://e-docs.bea.com/platform/docs81/interm/whatsnew.html>

Integrated e-docs and dev2dev

BEA e-docs and dev2dev converge! Now your favorite BEA documentation Web site is integrated with your favorite BEA developer portal. The same great documentation and the same great developer resources converge to present you a comprehensive technical resource center. We hope you like it!

Platform Support and System Requirements

For information on platform support, including hardware and software requirements, see the Supported Platforms page at the following location:

<http://e-docs.bea.com/platform/docs81/support/index.html>

For detailed information about platforms supported by BEA WebLogic Server, see the Supported Configurations page at:

<http://e-docs.bea.com/wls/certifications/index.html>

Adapters

The application integration framework provides the following key features to enable the integration of diverse enterprise systems:

- Standards-based architecture for hosting J2EE Connector Architecture (J2EE CA)-based adapters that connect enterprise applications to WebLogic Server.
- Application views for both event and service adapters.

For more information on adapters supported in this release, see:

<http://e-docs.bea.com/wl.adapters/docs81/index.html>

To download the adapters for WebLogic Integration 8.1, use the following URL:

http://commerce.bea.com/products/weblogicadapters/wl_adapter_home.jsp

Tutorials

To get hands-on experience with WebLogic Integration 8.1, we recommend that you try out the tutorials listed in this section. In addition, you may want to try out the WebLogic Workshop tutorials which are provided at the following URL:

<http://e-docs.bea.com/workshop/docs81/doc/en/workshop/guide/navTutorials.html>

Business Process and Data Transformation Tutorials

WebLogic Workshop 8.1 provides the following tutorials on the WebLogic Workshop Integration Extensions:

- Tutorial: Building Your First Business Process
- Tutorial: Building Your First Data Transformation

Information about running them is available at:

<http://e-docs.bea.com/workshop/docs81/doc/en/workshop/guide/navTutorials.html>

Tutorials for Trading Partner Integration—Coming Soon

To learn more about ebXML and RosettaNet business protocols, look for *Tutorials for Trading Partner Integration* which will be available soon at BEA's dev2dev site available at the following URL:

<http://dev2dev.bea.com/code/wli.jsp>

Note: We recommend that you use the RosettaNet example in the *Tutorials for Trading Partner Integration* when constructing your own RosettaNet PIPs using WebLogic Integration. The ebXML example in the *Tutorials for Trading Partner Integration* is a good reference for constructing your own ebXML business processes.

Documentation Additions

This section provides additions, changes, and corrections to the WebLogic Integration documentation available on the BEA documentation site at the following URL:

<http://e-docs.bea.com/wli/docs81/index.html>

Known Limitations

This section describes known limitations in the BEA WebLogic Integration 8.1 Release software. The known limitations are grouped by the following topics:

- [Administration And Configuration](#)

Note: This section contains limitations of the WebLogic Integration Administration Console in addition to other administration and configuration limitations.

- [Application Integration](#)
- [Business Process](#)
- [Cluster Configurations](#)
- [Controls](#)
- [Run Time](#)
- [Database](#)
- [Trading Partner Integration](#)
- [Run Time](#)

Administration And Configuration

In addition to the specific administration and configuration limitations listed below, see the following related limitations:

- [To Use Test View of Mapper, Must Use Default WLS Port of 7001](#)
- [Cannot Specify Values Over 24 Days in ebXML or RosettaNet Bindings](#)
- [For Cluster Configurations, in the WebLogic Integration Console, the Trading Partner Service Information is Not Always Displayed](#)
- [While Modifying Trading Partner Management Information, Wait For the Transaction To Complete On the Server, Before Modifying Additional TPM Information](#)
- [Dependent Application Views Not Suspended When Adapter Instance is Suspended](#)
- [Increasing the Transaction Timeout Set by Default](#)
- [Always Edit Application View or Service Connection Security Policy Using the WebLogic Integration Administration Console](#)
- [Security Policy Settings Must Be Edited In Specific Order](#)
- [Changes to Adapter Instance for Service Connection and to Connection Factory Not Applied](#)
- [Cannot Change Auto-Suspend Settings for Adapter Instance Using Console](#)
- [How to View More Events in the Test Browser](#)

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.

Restrictions on the Use of Message Ids

Several WebLogic Integration components are logging cataloged message ids in the range 510000-518999, 61000-610999, 800000-800999, 900000-924999. These ranges overlay the range reserved for user message ids. These numbers will be renumbered into the BEA reserved range in a future release or service pack. Your code should not use these message ids until this limitation fixed in a future service pack or release.

Updated Process Information in the WebLogic Integration Administration Console, Must Build Application in WebLogic Workshop

The process information displayed in the WebLogic Integration Administration console reflects the state of the application when it was last built in WebLogic Workshop using the **Build→Build Application** option from the **WebLogic Workshop** menu bar. For example, if you created two business processes in your application, built your application, and deleted one of the business processes, in the WebLogic Integration Administration Console two business processes would be reported. If you then built the application using the **Build→Build Application** option from the **WebLogic Workshop** menu bar and viewed the business processes in the Console, one business process would be reported.

WebLogic Integration Resources Require Process Projects

Many WebLogic Integration resources (for example: message broker subscriptions and versioning information) require a WLI app listener to be defined in the `WEB-INF/web.xml` file for the current project. When a process project is created, this WLI app listener will be defined by default in the `WEB-INF/web.xml` file, but for some projects which are not specific to WebLogic Integration—such as default web app projects, the WLI app listener will not be defined. During run time, these project may appear to work in some instances, but will fail when the required resource is accessed. For a process project, the following XML elements will be defined in the `WEB-INF/web.xml` file:

```
<listener>
<listener-class>
com.bea.wli.management.WliWebAppListener
</listener-class>
</listener>
```

Process Authorization Policies Are Reset Upon Redeploying From Workshop or When A Deployment Mode WebLogic Server is Restarted

With the WebLogic Integration Administration Console, you can set what roles can call a process, methods in that process, or control callbacks in that process. When you redeploy your application from Weblogic Workshop, these role settings will be reset to the default policy of no constraints. Similarly, when you restart your development mode WebLogic Server, your process settings will also be erased. This does not happen for a production mode server.

Workaround: Reenter your security settings after redeploying your application from WebLogic Workshop or restarting the development server. You can also refrain from setting these policies until you are testing in production mode.

Alternatively, use `@common:security` annotations in the JPD **Source View** during the development phase of the project. As you near the production phase, remove these annotations and then use the WebLogic Integration Administration Console to configure security.

Business Process Appears Active in Console Even If It Failed At Startup with Security Exception

In the instance page of the WebLogic Integration Administration Console may falsely report that a business process is still running even if that business process failed to run because of a security exception in a start **Client Receive** node.

Netscape 7.1 Not Supported For WebLogic Platform, Adobe SVG Viewer 3.0 Not Supported on Netscape 7.1

The interactive process graph of the WebLogic Integration Administration Console uses Adobe SVG Viewer Version 3.0. Adobe SVG Viewer Version 3.0 is not supported on the Netscape 7.1 browser. To learn more, see *Browser Requirements for the Interactive Graph* available at the following URL:

<http://edocs.bea.com/wli/docs81/manage/processmonitoring.html>

Netscape 7.1 is also not a supported browser for WebLogic Platform 8.1. Detailed information about the operating systems and browsers WebLogic Platform supports is provided at the following URL:

http://edocs.bea.com/platform/docs81/support/supp_plat.html

Application Integration

Directory Structure Change for Application View EJB

The application integration directory structure for application view EJBs has changed since the WebLogic Integration 8.1 Beta release.

Workaround: To reuse an application view that was created with the WebLogic Integration 8.1 Beta release, move the `schema/wlai-repository` directory and all its contents to the top level of the WebLogic Workshop application directory. For example, in the

`samples/integration/sampleApp` application, move `sampleApp/Schema/wlai-repository` to `sampleApp/wlai-repository`.

Republish the associated application views. The application views do not need to be changed, just republished. Remove any `wlai-repository.jar` files located in the `APP-INF/lib` directory.

Republishing the application views copies the proper schema files (XSD) from the `app/wlai-repository` directory to the `app/Schemas/wlai` directory, causing them to be compiled by the schema compiler and stored in `APP-INF/lib/Schemas.jar`. The XBean types remain the same as before, so no code changes are required. The republish operation also rebuilds the per-application view EJB to include all the resources needed to run the application view.

Dependent Application Views Not Suspended When Adapter Instance is Suspended

When an adapter instance is suspended using the WebLogic Integration Administration Console, dependent application views are not automatically suspended. The behavior of application views when an adapter instance is suspended depends on whether they invoke synchronous or asynchronous services:

- When a synchronous service is invoked, a check is performed to see if the adapter is suspended. If so, an `ApplicationViewSuspendedException` is thrown, and the application view is suspended.
- When an asynchronous services is invoked, the request can not be completed and the asynchronous processor puts the request back on the request queue.

Increasing the Transaction Timeout Set by Default

The current default JTA transaction timeout for domains generated by the Configuration Wizard is 30 seconds. Typically, many adapter calls can exceed 30 seconds.

Workaround: Increase the server's default transaction timeout using the WebLogic Server Administration Console:

1. Open the WebLogic Server Administration Console and log in.
2. In the left pane, select **Services->JTA**. The JTA configuration page is displayed.
3. In the right pane, edit the **Timeout Seconds** field to reflect a longer timeout value. The suggested value is 500 seconds.

Auto-Suspend Fails If Calling Process Does Not Have Administrator Privileges

The application integration auto-suspend feature requires that the calling process or service have administrator privileges. In most cases, users in production will not have administrator privileges when running business processes. Processes without administrator privileges encounter exceptions that result in the failure of the auto-suspend operation. An `access not allowed for subject` exception is thrown to the WebLogic Integration server log while the invoking client receives an `ApplicationViewSuspendedException`, which indicates the correct state.

Workaround: Contact BEA Support to obtain a patch for this problem. We do not recommend providing users in production with administrator privileges when running business processes.

Invoking Service in Sample Application Against Oracle Database Results in Exception

In the application integration sample (sampleApp), invoking the `AddCustomer` service in the `InsertBasedEventDemo` application view against an Oracle database results in a `ResourceException`. The exception is thrown because the date of birth field (DOB) is defined as type `varchar` rather than type `Date`.

Workaround: To use the `InsertBasedEventDemo` service against an Oracle database:

1. Start the Application Integration Design Console, log in, and select the sample application for the design session context.
2. Click on `InsertBasedEvents` to display the Summary page for the application view.
3. Click the Edit link to display the Administration page for the application view.
4. Click the Edit link for the `AddCustomer` service. The Edit Service page is displayed.
5. Change the SQL Statement field from:

```
insert into {myTableQualifiers}CUSTOMER_TABLE(FIRSTNAME, LASTNAME, DOB)
values([FirstName varchar], [LastName varchar], [DOB varchar])
```

to:

```
insert into {myTableQualifiers}CUSTOMER_TABLE(FIRSTNAME, LASTNAME, DOB)
values([FirstName varchar], [LastName varchar], [DOB Date])
```

6. Deploy the application view and republish it to WebLogic Workshop.
7. In WebLogic Workshop, build the Schemas project for sampleApp application.

Always Edit Application View or Service Connection Security Policy Using the WebLogic Integration Administration Console

Do not edit security policies for ApplicationView EJBs or service connection Connector modules using the WebLogic Server Administration Console. Doing so causes you to be unable to edit the security policy for the effected application views and service connections in the WebLogic Integration Administration Console.

Workaround: Edit all security policy for application views and service connections using the WebLogic Integration Administration Console.

Security Policy Settings Must Be Edited In Specific Order

When using the WebLogic Integration Administration Console to enable or disable container managed sign-on for an application view, you must redeploy the adaptor instance to make this change take effect. If security policy settings are not edited and deployed in the correct order, application view security policy settings may be lost when the application is redeployed.

Workaround: Please follow the suggested sequence when editing security policy settings:

1. Edit the container manager sign-on settings for application view. Here you set your WebLogic server to EIS role maps.
2. Redeploy the adapter instance associated with the application view.
3. Edit the security policy for the application view.

Changes to Adapter Instance for Service Connection and to Connection Factory Not Applied

When using the WebLogic Integration Administration Console, changes to the adapter instance for a service connection appear to be accepted by the console but are not applied. The application view continues to use the original adapter instance. Redeploying the application view does not resolve the problem.

Similarly, when using the Weblogic Server Administration Console to change the connection factory, the changes are not applied.

Workaround: Do one of the following options:

- Use the application integration design console to switch the service connection being used by the ApplicationView, then republish it.

- For the adapter instance problem, you can also use the AdapterDeploymentMBean API or access the MBean using `weblogic.Admin`. (The JMX utility called `weblogic.Admin` allows you to list/manipulate MBeans in the server. For more information on this utility, see “[weblogic.Admin Command-Line Reference](#)” in *WebLogic Server Command Reference* at the following URL: http://e-docs.bea.com/wls/docs81/admin_ref/cli.html.) The method to call is `setServiceAdapterInstanceID()` and `setServiceFactoryName()`. Make sure to end with a call to `applyServiceConnectionChanges()` or none of the changes you made will be applied to the MBean's state.

Cannot Change Auto-Suspend Settings for Adapter Instance Using Console

In both clustered and non-clustered environments, the WebLogic Integration Administration Console only allows you to change auto-suspend settings for an application view, not for the associated adapter instance. This occurs for the WebLogic Integration sample adapters and with the BEA WebLogic Adapters.

Workaround: To enable, disable, or otherwise modify auto suspend settings for a particular adapter instance:

1. Shut down the server.
2. Delete the back up file `AdapterInstanceConfiguration.xml.backup` if the file exists.
3. Modify `AdapterInstanceConfiguration.xml`, by setting `AutoSuspendEnabled` to `true` to enable or `false` to disable the auto-suspend feature. Edit `AutoSuspendTimeout` to set the timeout in seconds (the default timeout value is 1800 seconds).
4. Reboot the server.

You can also use the AdapterDeploymentMBean API or access the MBean using `weblogic.Admin`. (The JMX utility called `weblogic.Admin` allows you to list/manipulate MBeans in the server. For more information on this utility, see “[weblogic.Admin Command-Line Reference](#)” in *WebLogic Server Command Reference* at the following URL:

http://e-docs.bea.com/wls/docs81/admin_ref/cli.html.)

Issues with DBMS LOCALTX Adapters

The implementation of the SPI LocalTransaction interface in the ADK improperly sends local transaction demarcation events to connection event listeners. This causes warnings from the WLS Connector about local resources being freed that have already been freed, and in high-load cases can cause JTA Invalid Transaction exceptions. There is a patch for this issue available from WebLogic Integration customer support.

JTA XAER_PROTO Exceptions Can Be Thrown When the Service Connection Connection Pool Has No Free Connections

When a service connection's connection pool (for adapter's using LocalTransaction transaction support setting) is drained of free connections, it can cause JTA XAER_PROTO exceptions. This happens because the WLS Connector container mistakenly does not enlist the connection from the pool in the global transaction, leaving any work done on the connection under an implicit local (not XA) transaction.

Workaround: Increase the max pool size for your service connections, so you don't run out of free connections in the pool.

Business Process

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.

Synchronous Sub-Processes Invoked From a JWS, Page Flow, or Business Control, Will Lose Tracking Data

If a JWS, Page Flow, or Business Control, invokes a synchronous business process (sub-process) and the JWS, Page Flow, or Business Control rolls back the transaction, the tracking data of the sub-process will be lost. The term *Business Control* refers to custom control and not a trading partner integration or business-to-business control.

Note: A business process invoking synchronous sub-process will not have this limitation—the synchronous sub-process's tracking data will not be lost.

Workaround: Add an intermediate business process to your application—have the JWS, Page Flow, or the Business Control invoke an intermediate business process which then invokes the synchronous sub-process.

The freeze on failure Property Ignored for Explicit Transactions

The `freeze on failure` property for explicit transactions is ignored. When set on JPD start nodes, this property operates correctly and as documented.

Use of the `jpd:unexpected-message` Annotation

Business processes often include nodes such as **Control Receive** or **Client Request**, upon 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 **Properties** palette, and setting the `action` property from the default `save` to `discard`. The annotation will be placed into the JPD source code, as shown in the following code segment:

```
/**
 * @jpd:unexpected-message action="discard"
 */
```

When Archiving Business Processes Using a XA Driver, Must Set `SupportsLocalTransaction` Equal to `True`

If you are archiving business processes and using a XA driver for the business process archiver data pool (`bpmArchPool`), you will need to add the `SupportsLocalTransaction="true"` setting to the XA connection pool definition in your `config.xml`, to enable the data source to participate in a local transaction.

Process Graph May Use Service Control Icon for Service Broker Control Nodes

For domains created from the WebLogic Integration domain template, SVG- and PDF-based process graphs may show the **Service Control** icon (machine in front of a blue globe) for process nodes that actually use the **Service Broker Control** (yellow diamond linked to two blue squares). Specifically, this display anomaly appears in the following places:

- The **Process Graph** tab of the Workshop Test Browser
- The **Graphical View** of the **Process Instance Details** pane in the WebLogic Integration Administration Console

- The **Printable Graph** of the **Process Instance** pane of the WebLogic Integration Administration Console

Workaround: Refer to the **Design View** of your business process in Weblogic Workshop for the definitive graph of what controls are used by your process. Alternatively, name your process nodes to indicate if you are using the **Service Broker Control** or the **Service Control** nodes.

Control Send Node in a Business Process Invoking a WebLogic Workshop Web Service Which Invokes a Method on a Stateful or Stateless EJB May Fail

If a **Control Send** node in WebLogic Integration business process invokes a WebLogic Workshop Web service and the Web service then invokes a method on a stateful or stateless EJB, an exception may be thrown. The exception listed in the console, will be similar to the following exception:

```
<Jun 26, 2003 3:49:56 PM EDT> <Error> <WLW> <000000> <Unable to remove bean
instance: weblogic.ejb20.locks.LockTimedOutException:
[EJB:010107]The lock request from EJB:SimpleSS with primary
key:145008051647152128 timed-out after waiting 0 ms. The transaction or
thread requesting the lock was:
Thread[ExecuteThread: '11' for queue: 'weblogic.kernel.Default',5,Thread
Group for Queue: 'weblogic.kernel.Default'].
```

Cluster Configurations

In addition to the specific cluster configuration limitations listed below, see the following related limitation:

- [In Cluster Configurations, Updates to the TPM Configuration Require a Restart of the Managed Servers](#)
- [For Cluster Configurations, in the WebLogic Integration Console, the Trading Partner Service Information is Not Always Displayed](#)

Using the Configuration Wizard to Modify Cluster Configurations

While creating a WebLogic Integration cluster domain using the Configuration Wizard, you will need to make some additional changes using the Configuration Wizard as described in the following steps:

1. Go forward to the **JMS Distributed Destinations to Servers** pane. (Keep clicking Next until you get to the **JMS Distributed Destinations to Servers** pane.) Delete the following attributes on the pane:

```
-wli.internal.egmail.queue_auto  
-wli.internal.egfile.queue_auto
```

2. Backup to the previous **Configuring JMS Queues** pane. (Keep clicking **Previous** until you get back to the **Configuring JMS Queues** pane.) For each managed server, delete the following attributes from the pane:

```
-wli.internal.egmail.queue_auto_N  
-wli.internal.egfile.queue_auto_N
```

Where *N* is the number of the managed server that has been added by the configuration wizard.

3. Add the following JMS queue to the single managed server that hosts the event generators:

```
-wli.internal.egmail.queue  
-wli.internal.egfile.queue
```

To learn more see, *Configuring a Clustered Deployment* in [Deploying WebLogic Integration Solutions](#) and [Configuring Managed Servers, Clusters, and Machines](#).

If you are using the Trading Partner functionality of WebLogic Integration, also follow the steps in [“Using the Configuration Wizard to Modify Cluster Trading Partner Integration Configurations”](#) on page -18.

Using the Configuration Wizard to Modify Cluster Trading Partner Integration Configurations

For a WebLogic Integration cluster involving trading partners, after creating a WebLogic Integration cluster domain `config.xml` file using the Configuration Wizard, you will need to make some additional changes in WebLogic Server console as described in the following steps:

1. Login to WebLogic Server console with Administrator authority.
2. In left panel, find *Domain_Name* → Deployments → Web Application Modules → B2BDefaultWebAppApplication. (Where *Domain_Name* is the name of the current domain.)
3. Add Admin Server to its target and deploy.

The Admin server should now be in the target list in deploy panel with `Active` as the **Module Status** and `Success` as **Status of Last Action**.

In Cluster Configurations Using a Non-XA Oracle Driver, Certain JTA Exceptions May Be Thrown, Can Be Ignored

In cluster configurations using a non-XA Oracle driver, the JTA

`javax.transaction.xa.XAException: No connection associated with xid` exception may appear in the WebLogic Server console window for the WebLogic Server that hosts the WebLogic Server Administration console (WebLogic Admin Server). You can ignore these messages.

In Cluster Configurations, The ClusterAddress Attribute Must be Set for WebLogic Integration Clusters

In cluster configurations, while running business processes or using the WebLogic Integration Administration console, the following error message may appear in the WebLogic Server console window for the WebLogic Server that hosts the WebLogic Server Administration console (WebLogic Admin Server):

```
Failed to initialize clustered process configuration backend
```

If you encounter this problem, you must set the `ClusterAddress` attribute for the cluster. To learn how, see “Cluster Address” in [Setting up WebLogic Clusters](#) in *Using WebLogic Server Clusters*, which is available at the following URL:

<http://e-docs.bea.com/wls/docs81/cluster/setup.html>

In Cluster Configurations, Restarting a Managed Server May Fail If Admin Server Is Shutdown

In cluster configurations, the restart of a Managed Server may fail, if the Administration Server is shutdown. The Administration Server must be started before Managed Server can be restarted.

Steps for Manual JMS Server Migration from a Failed Managed Server

The following steps describe how to prepare managed servers for JMS migration. This should be done prior to using the domain in production:

1. Boot the Administration Server.
2. Start the WebLogic Server Console.
3. In the left navigational tree pane, expand the current cluster, expand **Services**, expand **JMS**, and expand **Servers**.
4. For each JMS Server in the cluster, repeat the following steps:

- a. In the left navigational pane, select the JMS Server (for example: `cgJMSServer_auto_1`)
 - b. In the right pane, choose the **Target and Deploy** tab.
 - c. In the **Target** drop-down menu, change to the migratable target. For example, change `slave1` to `slave1 (migratable)`.
 - d. Click **Apply**.
5. Retarget WLI Admin to all servers in the cluster by following these steps:
- a. In the left navigational tree pane, in the current cluster, expand **Deployments**, expand **Applications**, expand **WLI System EJBs**, and select **WLI Admin**.
 - b. In the right pane, choose the **Targets** tab.
 - c. Select the **All servers in the cluster** option.
 - d. Click **Apply**.
6. If you use event generators, retarget the event generator to all servers in the cluster, by following these steps:
- a. In the left navigational tree pane, in the current cluster, expand **Deployments**, expand **Applications**, expand **EJB Modules** and select your event generator.
 - b. In the right pane, choose the **Targets** tab.
 - c. Select the **All servers in the cluster** option.
 - d. Click **Apply**.

The following steps describes how to migrate the JMS server:

1. Select the correct migration destinations for your cluster. In the left navigational tree pane, under the current cluster, expand **Servers**, and select a managed server.
2. In the right pane, choose the **Control** tab, and then **JMS Migration Config** tab.
3. In the **Available** plane, choose the all Managed Servers and click **Apply**.

To learn more see, *Configuring a Clustered Deployment* in [Deploying WebLogic Integration Solutions](#).

Using Oracle XA Driver, OracleXAException May Occur When Recovering a Failed Server

When using the Oracle XA driver, the `javax.transaction.SystemException: Failed to delist resources while suspending: oracle.jdbc.xa.OracleXAException` may occur when recovering a failed server. This may impact running business processes.

Controls

Do Not Use Underscores In Control Callback Method Names

Do not use underscores in control callback method names. Using underscores can cause business process not to be displayed correctly in the **Design View**, making it difficult to design your business process.

Empty Directory Name for Append Operation Results in Exception (File Control)

When using the File control for an append operation, specifying an empty directory (`directory-name="/"`) results in the following exception:
`java.io.FileNotFoundException.`

Workaround: Specify a directory name in the **directory-name** field.

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. However, if an archive directory is specified, the file will then be moved to the archive directory and will not be over-written.

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.

When Adding A Control Factory, the First Argument of a Callback Handler Must be the Same Type As the Control

When creating an instance of a Control factory, the first argument of the callback handler for the control factory must be of the same type as the control. For example, if you create an instance of

an **ApplicationView** Control factory by selecting the **Make this a control factory that can create multiple instances at runtime** checkbox in the **Insert Control - ApplicationView** dialog box. Then in your business process, you create a **Client Receive** node and drop the instance of the **ApplicationView Control** factory on the node. In the **Source View** of the business process, red error indicators will appear. Place the cursor over the error indicator and follow the directions in the dialog box for fixing the callback method.

Generated Service Broker and Service Controls Have the Same Name

When a Service control and Service Broker control are auto-generated for the same Web service, the second control generated will have the same name as the first. Subsequently, the first JCX file is overwritten by the second JCX file. In most cases, you will not require both types of control within the same Web service.

Workaround: If you require both a Service control and a Service Broker control in the same Web service, use the **File**→**New** menu to create the second control via WSDL.

Service Broker Control Cannot Be Created For a Business Process Not in the Current Application

When creating a Service Broker control referencing a business process (JPD), the business process must be in the current WebLogic Workshop application. Attempts to create a Service Broker control referencing a JPD file in another application result in a `Cannot parse file` error.

Parent Process Not Notified of Failed Call to Subprocess (Process Control)

When using a Process control to communicate between business processes, a buffered `clientRequest` to a subprocess may fail authorization checks. If an authorization failure occurs, the message is discarded, but the caller (sender) does not receive notification of the failure.

Multiple Channel Definitions For the Same Channel Name Are Allowed

Multiple channel definition elements can exist with the same channel name in a single channel file. Elements can specify conflicting information as to the type of the messages.

Workaround: Manually check channel files for consistency.

Need to Make Edits for deadletter Channels To Show Up in the Node Builder When Defining a Static Channel Subscription

When a message is received by a channel that has no subscription rules defined on it, the message gets routed to the corresponding deadletter channel. However, if there are no subscribers defined on the deadletter channel, this message is discarded. Currently the deadletter channels do not show up on the node builder when defining a static subscription.

Workaround: These channels need to be defined explicitly by adding the following attribute: `channelPrefix='/deadletter'` to the Channel subscription file, as shown in the following example:

```
<channels xmlns="http://www.bea.com/wli/broker/channelfile"
channelPrefix="/deadletter">
<channel name ="xml" messageType="xml"/>
<channel name ="string" messageType="string"/>
<channel name ="rawData" messageType="rawData"/>
</channels>
```

These channels will now be visible on the node builder.

Note: The channel names and the channel prefix should exactly match those defined in the preceding example. When the preceding Channel subscription file is built, the following Channel is generated:

```
/deadletter/xml
/deadletter/string
/deadletter/rawData
```

Note: The channel names are case sensitive.

Do Not Associate a Single XA Connection Pool with Different Datasources

Do not associate the same XA connection pool with two different datasources. This problem has been observed in some business processes using the Application View and Database controls. This causes problems with the way WebLogic Server enlists resources in a global transaction which includes connection requests through both datasources, as illustrated in the following scenario:

1. WebLogic Server enlists the common connection pool as a transaction resource on behalf of the first connection request through the datasource.
2. WebLogic Server fails to refrain from making a second enlist request when the first connection is requested through datasource

3. The second request to enlist this same resource results in an XAER_RMERR error.

The associated Oracle error is:

ORA-02044 transaction manager login denied: transaction in progress. Cause: A remote transaction manager tried to log in while a distributed transaction is in progress. A protocol error occurred in the remote transaction manager.

For more information, see the Note in “When to Enable Global Transactions in a Data Source” at the following URL

http://edocs.bea.com/wls/docs81/ConsoleHelp/jdbc_datasources.html

Data Transformation

In addition to the data transformation specific limitations listed below, see the related information in the following sections:

- [XML Document Field Size Limitation](#)

Test XML Generation for XML Schemas With Choice Groups Not Supported

The **Test View** of the mapper does not generate the input XML test data correctly for XML Schemas that contain choice groups.

To learn more about choice groups in XML Schemas see the following URL:

<http://www.w3.org/TR/xmlschema-0/#groups>

Workaround: Create the XML data using XML Spy, which is bundled with WebLogic Platform and save the data in an XML file. In the **Test View** of the mapper import the XML file to use as the test data by clicking **Import**.

Mapper Does Not Display All the Array Inputs and Outputs of a Java Class

In the **Test View** of the mapper, only the first item of a Java class array will be displayed in either the **Source Schema** or **Target Schema** panes.

Java Inner Classes of JPD or JWS Files Cannot Be Used In Controls or Transformations

The Java Inner classes of JPD or JWS files cannot be used in Controls or Transformations.

Workaround: Move the Java Inner Class into its own standalone .java file.

Validation Not Supported for XML Schemas that Contain Substitution Groups

The schema validating done during run time or the validate done when you click **Validate** in the **Test View** tab of the XQ file during design time cannot be done on XML data based on XML Schemas which contain substitutions groups. To learn more about validating in WebLogic Integration, see the following URL:

<http://edocs.bea.com/workshop/docs81/doc/en/integration/dtguide/dtguidemap.pervalidating.html>

To learn more about Substitution groups in XML schemas see the following URL:

<http://www.w3.org/TR/xmlschema-0/#SubsGroups>

Workaround: Do not schema validate during run time or click **Validate** in the **Test View** tab of the XQ file for input XML Schemas which contain substitution groups. Additionally, you can redesign your schemas not use substitution groups and use the original element names in your XML documents.

Certain XQuery Keywords Are Not Allowed in XPath Expressions

The XQuery engine does not allow the following XQuery keywords to be used when selecting elements in XPath expressions:

- for
- let
- some
- every
- unordered
- validate

To learn more about the reserved XQuery keywords, see the following URL:

<http://www.w3.org/TR/2002/WD-xquery-20020816/#N4021F0>

Workaround: Do not use XQuery keywords as XML elements or attribute names. Alternatively, rewrite your XPath expression to use unabbreviated syntax when the abbreviated syntax results in a keyword. For example, replace `$a/for` with the unabbreviated syntax: `$a/child::for`. To learn more about the difference, see the following URLs:

<http://www.w3.org/TR/xpath20/#unabbrev>

<http://www.w3.org/TR/xpath20/#abbrev>

How to Regenerate Test Data Once the Data Has Been Modified or Deleted

Test data is generated for the input types in the **Test View** of an XQ file. You can edit the current XML data. If you want to regenerate the test data again, follow the steps described in the *Updating the Graphical Representation Displayed in the Source View of a XQ File* section in *Selecting Input and Output Types* available at the following URL:

<http://edocs.bea.com/workshop/docs81/doc/en/integration/dtguide/dtguidemapperimportschemas.html>

For example, you might want to regenerate the XML test data, if you edited the original XML data resulting in XML data not valid to the associated schema and now want to regenerate the input XML data, again.

Automatic Promotion of XmlObject Types with XSLTs Not Supported

You can cast from an untyped XmlObject variable to the appropriate XMLBeans class if you are sure about the XML Schema type as shown in the following code segment:

```
myTypedXmlDocVar = (com.bea.customer.CustomerDocument)
myUntypedXmlObjectVar;
```

Where *myTypedXmlDocVar* is a typed XML variable of type *com.bea.customer.CustomerDocument* and *myUntypedXmlObjectVar* is an untyped XML variable (XmlObject). To learn more, see the following URL:

<http://edocs.bea.com/workshop/docs81/doc/en/integration/wfguide/wfguideDataTypesCreate.html>

However, this type of casting does not work in XSLT transformations for WebLogic Integration 8.1.

Workaround: You will need to manually edit the code as shown in the following code segment.

```
myTypedXmlDocVar = (com.bea.customer.CustomerDocument)
com.bea.customer.CustomerDocument.Factory.newInstance().set(myUntypedXmlObjectVar);
```

Mapping Differs Between xsd:anyType to xsd:any Elements

In the mapper, when you map to or from an *xsd:anyType* element, you will see a solid data link that copies the contents of the input element to the output element. In contrast, when you map to or from an *xsd:any* element, you will see a dashed structural link that only initializes an empty output element.

Workaround: To copy the contents of an input document to an `xsd:any` element, draw and select the structural link and then click on the **Functions** tab. In the **Edit Function** pane, enter the following XQuery code:

```
$_inputDoc/@* , $_inputDoc/node()
```

Where `$_inputDoc` is the desired input document variable.

To copy the entire document including its root element, in the **Edit Function** pane, enter the following XQuery code:

```
$_inputDoc
```

Deviations from the W3C XQuery August 2002 Draft Specifications

The WebLogic XQuery engine conforms to the August 16, 2002 draft of the W3C XQuery Specification which is available at the following URL:

<http://www.w3.org/TR/2002/WD-xquery-20020816/>

The following are the known deviations from that specification:

- The computed document constructor is not supported.
- The CDATA section constructor and processing instruction constructor are not supported.
- Escaped whitespace characters are not supported.
- The XQuery `xf:NOTATION` constructor is not supported.
- The XQuery `normalize-unicode` function is not supported.

To Use Test View of Mapper, Must Use Default WLS Port of 7001

In this release of WebLogic Integration, in order to use the **Test View** of the Mapper, the port number for the associated WebLogic Server must be the default value of 7001.

Database

In addition to the specific database limitations listed below, see the following related limitations:

- [Do Not Associate a Single XA Connection Pool with Different Datasources](#)
- [In Cluster Configurations Using a Non-XA Oracle Driver, Certain JTA Exceptions May Be Thrown, Can Be Ignored](#)

- [Invoking Service in Sample Application Against Oracle Database Results in Exception](#)

To Change the Default Pointbase Port

When you use Configuration Wizard in Custom mode and specify Pointbase ports other than the default (9093), the generated startup scripts do not reflect the correct port value.

Workaround: You can edit the startWebLogic script (`startWebLogic.cmd` for Windows) and add the `-port=XXXX` option in the line that starts the startPointBase script (`startPointBase.cmd` for Windows) after the `startPointBase` command. (Where `XXXX` represents the correct port value.) This workaround only applies if all database connection pools use the same port. If they use different ports, we recommend commenting out the `startPointbase` command from the scripts and starting the database manually.

Intermittent Pointbase Lock Time Out Reported When Message Size is Greater than the Configured Doc Store Size

The WebLogic Integration inline size should be set greater than the maximum message size when using the Pointbase database. The inline size is set in the `wli-config.properties` file using the `weblogic.wli.documentMaxInlineSize` property.

For trading partner integration, the inline size should be set greater than the maximum payload size when using the Pointbase database.

For Oracle, Conversation State Tables Should Store BLOB Data in Tablespace Other than USERS

When you create the conversation state tables for a Workshop web service deployed to a production environment, Oracle users should not store BLOB data to the `USERS` tablespace.

Storing BLOB data in the `USERS` tablespace can exhaust the tablespace's storage capacity.

Workaround: Create a new tablespace to store the BLOB data. You can use the following table creation script as a template.

```
CREATE TABLE <*tableName*> (
  CG_ID varchar(768) not null,
  LAST_ACCESS_TIME number(19,0),
  CG_DATA BLOB,
  PRIMARY KEY (CG_ID)
)
LOB ("CG_DATA") STORE AS ( TABLESPACE <*blobTablespace*> )
```

For Pointbase, If a Process Variable Exceeds 4 Mb, All Conversations Are Terminated In the Process

Using the Pointbase database, during run time, if the size of the data stored in a process variable exceeds 4 MBytes, all conversations will be terminated in the current process.

Workaround: Manually create all the conversation state tables in the Pointbase database with a column size larger than the size of the data stored in process variables. Alternatively, if the conversation state tables have already been created, increase the default column size of all the conversation state tables in the Pointbase database to a size larger than the size of the data stored in process variables. During development, these tables are typically autocreated by WebLogic Workshop with names based on the app/package/filename of the source (for example:

JWS_WLIPROD_BPM_ORDERPROCESSOR).

Trading Partner Integration

In addition to the trading partner integration specific limitations listed below, see the related information in the following sections:

- [Tutorials for Trading Partner Integration—Coming Soon](#)
- [Intermittent Pointbase Lock Time Out Reported When Message Size is Greater than the Configured Doc Store Size](#)
- [XML Document Field Size Limitation](#)
- [Using the Configuration Wizard to Modify Cluster Trading Partner Integration Configurations](#)

Creating a New RosettaNet Participant, Generates Callbacks Using the Long Form, Should Use Short Form

In WebLogic Workshop, when you create a new **RosettaNet Participant Process File**, the business process created uses the long form to refer to callbacks but a newer short form is the preferred way to refer to callbacks. In your RosettaNet Participant process, use the short form—for example do not use `callback.sendReceiptAcknowledgement` (the long form) but instead use `callback.sendAck` (the short form). In this release of WebLogic Integration, both the long or short forms of the callbacks will be supported but converting to the short form is recommended because the short form is easier to enter and understand, and is more consistent with the model followed on the initiator side.

In Cluster Configurations, Updates to the TPM Configuration Require a Restart of the Managed Servers

In certain cases, when running WebLogic Integration in a cluster, updates to the TPM configuration require a restart of the managed servers to take effect. Please contact BEA Support for details. Please note that the restart is *not* required in a non-clustered environment

ebXML Service Name Cannot Contain Double Byte Characters

In this version of WebLogic Integration, the ebXML service name cannot contain double byte characters.

For Trading Partners Representing WLI Business Connect, Only One Version of the ebXML Protocol Can Be Used At One Time

In this release of WebLogic Integration, when exchanging ebXML messages with a trading partner that uses WLI Business Connect, you can only use one version of ebXML Message Service protocol (either ebXML MS 1.0 or ebXML MS 2.0), because WLI Business Connect uses the same HTTP endpoint for a given trading partner regardless of the ebXML version. You cannot configure more than one protocol binding for a given partner in WebLogic Integration that uses the same HTTP endpoint.

Persist Duration Attribute For ebXML

Persist Duration Stored Incorrectly If Specified in Milliseconds

The value of the persist duration attribute in an ebXML protocol binding is saved incorrectly if it is specified in milliseconds.

Workaround: In the WebLogic Integration Administration Console, specify the `persist duration` attribute in seconds, minutes, or hours—not in milliseconds. Alternatively, specify the duration using the WebLogic Workshop string format, for example: `2 hours 3 minutes`.

How the Persist Duration Attribute Is Used

On the receive side, the `persist duration` attribute is not used to determine how long the message is persisted in the database, instead that is determined by your archiving policy.

For the outgoing message, the `persist duration` attribute will be stored as time to live in the message header.

For Cluster Configurations, in the WebLogic Integration Console, the Trading Partner Service Information is Not Always Displayed

In cluster configurations, the trading partner service information is not always displayed in the WebLogic Integration Administration Console, even if messages have been passed between trading partners and the service information has been recorded in database.

Workaround: To view the Trading Partner Service information, export the Trading Service information using the Console.

While Modifying Trading Partner Management Information, Wait For the Transaction To Complete On the Server, Before Modifying Additional TPM Information

After modifying trading partner management information using the WebLogic Integration Administrating console, wait for the transaction to complete on the server, before modifying additional information. The status bar on the browser will report when the transaction is complete. This is required to make sure that the TPM cache on the server is not in an inconsistent state—even in case of failure.

ebXML Service Name and RosettaNet PIP Name Must be Unique Among Participant Processes

If you have several participant processes with the same ebXML service name or RosettaNet PIP name in different packages, edit the name to make it unique. By default when you create a participant process, the generated name is the same as the JPD name.

Trading Partner Integration Messages Must Not Have A Null Payload

For this release of WebLogic Integration, all trading partner integration messages must not have a null payload.

Do Not Start Participant Processes From the Test Browser

If you run a trading partner integration participant business process using the WebLogic Workshop test browser, an exception might be thrown. By design, the participant business processes should be invoked by their initiator trading partner integration business process. In the WebLogic Workshop test browser, run the initiator trading partner integration business process to invoke the participant business processes.

Do Not Use the Attributes: to-selector and from-selector at the Control Level in ebXML and RosettaNet JCX Files

Do not use the attributes: `to-selector` and `from-selector` at the control level in ebXML and RosettaNet control definition files (JCXs). These attributes only apply to a send method (in ebXML control definition) or to control instance declarations in a JPD file (ebXML and RosettaNet).

Cannot Specify Values Over 24 Days in ebXML or RosettaNet Bindings

In this release of WebLogic Integration, using the WebLogic Integration Administration Console, you cannot specify duration values in bindings over 24 days.

- While adding EBXML bindings the following fields cannot exceed 24 days:
 - `Retry Interval`
 - `Persist Duration`
- While adding Rosettanet bindings the following fields cannot exceed 24 days:
 - `Retry Interval`
 - `Process Timeout`

Generated Demo Certificate on Solaris May Have Invalid Characters and May Fail to Reimport

The WebLogic Integration Administration Console allows the generation of demo certificates for testing purposes. On Solaris, these demo certificates may be generated with invalid characters. If you export these certificates using the console and then reimport them, the following exception is generated:

```
org.xml.sax.SAXParseException: An invalid XML character (Unicode: 0x0) was found in the element content of the document.
```

Note: Valid certificates generated from other sources can be imported and exported.

Run Time

In addition to the run time specific limitations listed below, see the related information in the following sections:

- [WebLogic Integration Resources Require Process Projects](#)

XML Document Field Size Limitation

During run time, the text fields in XML documents processed by WebLogic Integration cannot have a UTF-8 encoded length of more than 65535 bytes.

