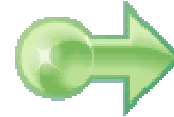




IBM Software Group

WebSphere® Process Server V6

Migration overview



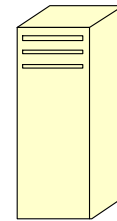
@business on demand.

© 2007 IBM Corporation
Updated September 19, 2007

This presentation will cover migrating to WebSphere Process Server V6 from previous releases of WebSphere products.

Goals

- High level overview of the migration utilities for
 - ▶ WebSphere InterChange Server 4.2 and 4.3
 - WebSphere InterChange Server
 - ▶ WebSphere MQ Workflow 3.5
 - WebSphere MQ Workflow
 - ▶ WebSphere Business Integration Server Foundation 5.1
 - WebSphere Business Integration Server Foundation



WebSphere
Process Server V6

WebSphere Process Server is the merger of 3 existing product lines, the WebSphere InterChange Server, the WebSphere MQ Workflow and the WebSphere Business Integration Server Foundation.

Section

Migration or upgrade?




The agenda for this presentation is to focus on the difference between migration and product upgrade.

Migrate or upgrade?

- Migration
 - ▶ Source artifact migration
 - Take the artifacts that compose an application and convert them into V6 artifacts.
 - Authoring activity (make adjustments for performance and clarity)
- Upgrade (not available in WebSphere Process Server V6)
 - ▶ Take an existing WebSphere Business Integration Server Foundation runtime environment and upgrade it, to WebSphere Process Server V6.
 - This uses the WebSphere migration model with pre and post upgrade command line files.
 - Runtime activity


Terminology is important and terms are often interchanged, leading to confusion. The terms Migration and Upgrade are defined here for clarity.

Note the emphasis on authoring as opposed to runtime environments. Source artifact migration is an activity that occurs during development using the WebSphere Integration Developer V6, while Upgrade is an activity specific to existing WebSphere Business Integration Server Foundation runtime installations. Product Upgrade from WebSphere Integration Server Foundation is NOT available in WebSphere Process Server V6.0.

IBM Software Group 

Section

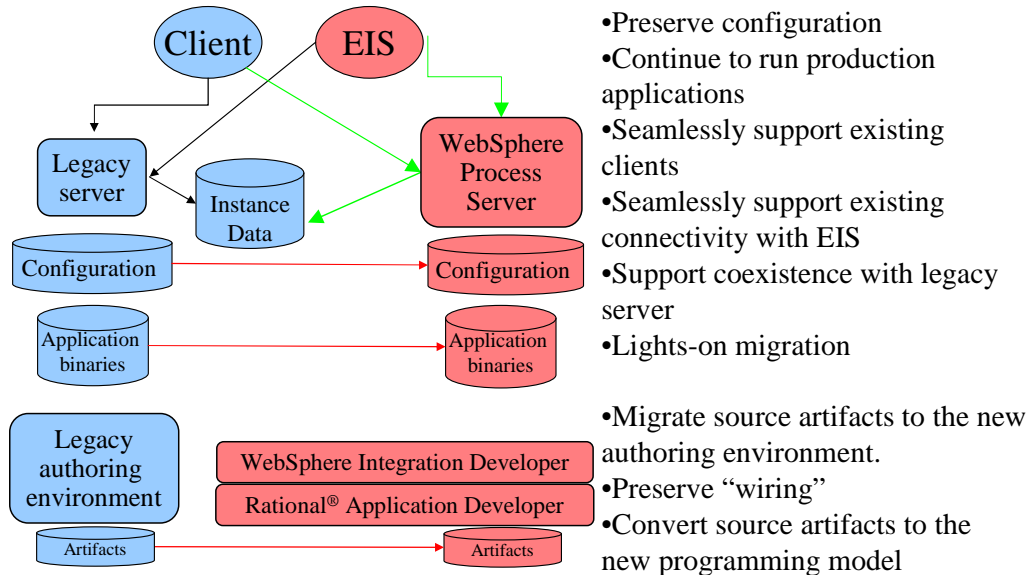
Optimal view



Migration overview © 2007 IBM Corporation 5

This section will provide an example of the optimal migration scenario.

The optimal migration story



Shown here is the ideal migration scenario. The existing systems are depicted on the left in blue. Application binaries and authoring artifacts are all moved seamlessly to the new environment. Everything runs as-is, and the existing client applications continue to function with the existing applications, the new applications and the new client applications are able to work with the previously existing applications. The existing application binaries are moved to WebSphere Process Server V6, along with the appropriate configuration information, which is converted if necessary.

Section

Migration: WebSphere InterChange Server

This section will cover the migration of WebSphere InterChange Server.

Source artifact migration

- Support is available beginning with WebSphere Process Server V6
- Start by exporting the WebSphere InterChange Server repository to a jar.
 - ▶ Use the WebSphere InterChange Server system manager.
 - ▶ Include all artifacts needed for a complete solution
 - Business objects and maps
- Three options for proceeding:
 - ▶ First steps
 - Part of the WebSphere Process Server installation (on distributed platforms)
 - Uses **reposMigrate.bat**
 - ▶ Command Line
 - Uses **reposMigrate.bat** (on distributed platforms)
 - ▶ WebSphere Integration Developer
 - Welcome (uses the import)
 - Import the WebSphere InterChange Server jar into WebSphere Integration Developer

Before beginning, you should review the SCA programming model.

The process of source artifact migration for WebSphere InterChange Server begins by exporting the artifacts from the WebSphere InterChange Server system to a jar file. Using the WebSphere InterChange Server System Manager, export all the artifacts that comprise a complete solution. That is to say, get everything that is referenced so that there will be no unresolved references when importing to WebSphere Process Server V6.

The next step is to import the jar into WebSphere Integration Developer V6. There is a special import type that will recognize the artifacts and make the necessary conversions, creating new SCA artifacts.

It is recommended that WebSphere Integration Developer be used initially. This will provide the opportunity to understand how the conversions are made and to become familiar with the SCA components that are generated. Once a thorough understanding is achieved, the command line approach can be used to automate the process.

The migration can also be done from the “First Steps” application or from the command line using the reposMigrate.bat.

reposMigrate.bat is located in the bin directory of the WebSphere runtime installation and is described in the WebSphere Integration Developer ‘Help’ under the Migration topic

Additional migration capabilities



- Access EJB EAR
 - ▶ JService calls route to business object
 - ▶ Must be installed
- Event sequencing
- Failed events - documentation
- Security - documentation
- WebSphere InterChange Server heritage APIs
 - ▶ Supported as deprecated
 - ▶ Maps to WebSphere Process Server adapters
- WebSphere InterChange Server application installation

9

Migration overview

© 2007 IBM Corporation

WebSphere InterChange Server allowed you to not only invoke a collaboration as the result of input from an adapter, but also from standard J2EE EJB calls through the Access EJB hosted in a J2EE environment. The Access EJB will be replicated in WebSphere Process Server as an EJB that uses the migrated module naming schema to allow JService calls to route the input Business Object to the appropriate migrated module. All external clients can maintain the same EJB calls, but instead of invoking the target WebSphere InterChange Server collaboration, it will invoke the BPEL in the migrated module. The Access EJB EAR must be manually installed.

WebSphere InterChange Server provided server support for event sequencing for Message Driven Beans. The migrated modules will take advantage of WebSphere Process Server event sequencing to retain functional parity.

WebSphere InterChange Server collaboration templates and maps can contain custom Java code that references a set of supported APIs that were delivered with WebSphere InterChange Server. Many of these supported APIs are implemented in a deprecated fashion for the WebSphere Business Integration Server Foundation runtime. These APIs will take advantage of the adapter usage pattern in WebSphere Process Server.

The WebSphere InterChange Server Application installation will read the administrative artifacts data created by the migration and create WebSphere Business Integration Server Administrative Objects that match those that were found in WebSphere InterChange Server.

Migration - reposMigrate

- Every source WebSphere InterChange Server artifact is represented as an XML document and will be converted into one of the following types of WebSphere Process Server artifacts.
 - ▶ SCA component
 - ▶ Shared artifact
 - ▶ Administrative artifact
- Two main options
 - ▶ Create J2EE™ EARs to be deployed manually
 - Uses *ServiceDeploy* to create the EARs
 - ▶ Create J2EE EARs and have them automatically deployed using **wsadmin**.
 - Uses *ServiceDeploy* to create the EARs
 - then *wsadmin* to deploy the EARs to the runtime.

ServiceDeploy is the command utility for the WebSphere Process Server V6 used to compile and package the SCA and J2EE artifacts programmatically. The reposMigrate utility receives the WebSphere InterChange Server artifact jar and creates the SCA components and then invokes the ServiceDeploy utility to create the J2EE Ear. The J2EE EAR can then be deployed manually using the WebSphere Process Server administrative console, or programmatically using the WebSphere command line utility, wsadmin. Additionally the reposMigrate utility includes an option to invoke the wsadmin utility after the ServiceDeploy utility has created the EAR.

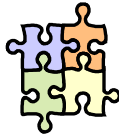
All of these command line utilities can be further automated and managed using ANT in combination with a source code control system.

Before the EARs are deployed to the runtime, whether you do it automatically or manually, the WebSphere InterChange Server environment must be prepared by quiescing the system as described in the WebSphere Integrated Developer help.

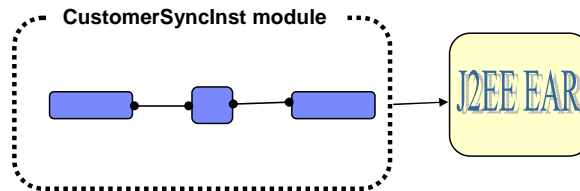
Artifacts and reposMigrate

- Converts each artifact to its corresponding WebSphere Integration Developer artifact.

WebSphere InterChange Server collaboration



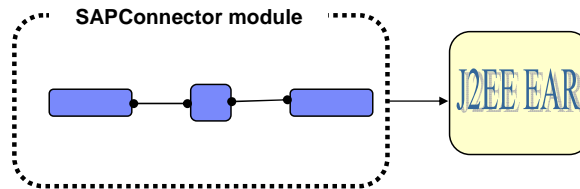
CustomerSynclnst.cwc



WebSphere InterChange Server connector



SAPConnector.con



It is important to note that the WebSphere InterChange Server artifacts get mapped to SCA Modules and that SCA Modules get deployed as J2EE EARs. That is to say, that there will be an EAR for each of the WebSphere InterChange Server artifacts.

Migration using WebSphere Integration Developer

- WebSphere Integration Developer will let you **import** the WebSphere InterChange Server repository jar and will create all the modules for you.
 - ▶ You can then inspect the results and adjust if you need to.
 - ▶ You can then test resulting modules and then
 - ▶ Deploy the EAR to the runtime when you're satisfied.
 - ▶ Your Java™ code used to manipulate data will be preserved and will run unchanged.
- Import can be invoked by either the
 - ▶ Welcome page – using the Migration Wizard
 - ▶ File -> Import -> WebSphere InterChange Server jar file
 - ▶ You supply
 - Location of the source jar
 - The name of the module, which is really a module prefix, which will be used with all the modules created from the various WebSphere InterChange Server artifacts.

Importing the WebSphere InterChange Server repository jar into WebSphere Integration Developer is the recommended approach in the beginning. This will provide the opportunity to easily inspect the results and make any necessary changes.

The WebSphere InterChange Server APIs have been deprecated, but to facilitate the migration an interface library is provided which will map the WebSphere InterChange Server API calls to WebSphere Process Server/SCA API calls. This will allow written Java code to run unchanged.

For a given WebSphere InterChange Server repository jar there will be many SCA modules generated. The module name supplied to the wizard will be pre-pended to the name of the artifact. For this reason it is recommended that the module name be short and distinctive.

Section

Migration: WebSphere MQ Workflow

The next item on the agenda is migrating from WebSphere MQ Workflow.

WebSphere MQ Workflow migration

- Support is available beginning with WebSphere Process Server V6
- Start by exporting the Flow Definition Language (FDL) from WebSphere MQ Workflow buildtime
 - ▶ Requires semantically complete FDL
 - export DEEP from WebSphere MQ Workflow buildtime
- Use WebSphere Integration Developer to invoke the FDL2BPEL translation utility
 - Welcome (uses the import)
 - Import
- Optimize the generated output based on knowledge of WebSphere Process Server and BPEL.

The recommended migration strategy is to use the migration tool to generate the first cut of the BPEL implementation. This will capture the flow; then use this as the starting point for refinement and optimization of the BPEL implementation.

Migration – FDL2BPEL mappings

- Mapping FDL to BPEL 2.0 (overview)
 - ▶ Mapping FDL data containers to XMLSchema definitions
 - ▶ Mapping FDL source / sink to BPEL “Receive / Reply” activities
 - ▶ Mapping FDL control flow to BPEL “Flow”
 - ▶ Mapping FDL data connector to BPEL “Assign activity”
 - ▶ Mapping FDL process activity to BPEL “invoke activity”
 - ▶ Mapping FDL block to BPEL “Scope”
 - ▶ Mapping FDL empty activity to BPEL “empty activity”
 - ▶ Mapping FDL User-defined program execution server activity to BPEL “service invocation activity”
 - ▶ Mapping FDL staff assignment to “Human task”

In FDL, the invocation types are distinguished using the properties associated with the activities. In BPEL there are explicit kind-of activities such as ‘empty’, ‘Human Task’, and ‘Service’.

Many but not all of the MQ Workflow constructs have BPEL analogs. The most notable difference is in the area of the invocation types, as noted here.

Control flow

- BPEL link element connects activities

WebSphere MQ Workflow Process Model Construct	BPEL with Extensions Construct
Transition condition	Transition condition (of <i>link</i> element)
Start condition of activity	Join condition (synchronizes <i>link targets</i>)
Exit condition of activity	Condition (controls a <i>While</i> activity)

In BPEL, the Link element is represented graphically as the line joining two activities. The BPEL link has the 'transition condition' and the 'join condition' modifiers that control the flow to provide the same functionality as the 'transition condition' and the 'start condition' in WebSphere MQ Workflow.

WebSphere MQ Workflow uses an implied loop to keep the activity executing until the exit condition is met. BPEL uses a while construct that must be explicitly specified.

Data flow

- The FDL concept of data flow does not exist in BPEL
 - ▶ BPEL uses global variables and messages to manage the data flow from one activity to another.
 - ▶ Assign activity
 - Used to assign values to the variables used by the messages.
 - ▶ Typed messages used on input and output of activities.
- FDL data containers are mapped to XML schema definitions

The area of Data Flow is where WebSphere MQ Workflow and BPEL differ greatly. With BPEL there is no explicit data flow and data is stored in global and local variables and can be passed from one activity to another as a parameter when invoking an activity. The data is implemented as SCA business objects in the WebSphere Process Server V6 BPEL business processes.

With FDL the data is defined with the data containers, and in SCA BPEL, data is defined using XML schema definitions.

FDL2BPEL limitations

- User-defined program execution server
 - ▶ Support is not functional
- Program execution agent
 - ▶ Not available
 - ▶ No equivalent in the WebSphere process integration programming model.
- Program execution server
 - ▶ Not available
- Predefined members do not exist in BPEL
 - ▶ Variables and messages must be explicitly set in BPEL

The FDL2BPEL tool has several limitations. Some best practices to use when building WebSphere MQ Workflow-based workflow applications can help the migration process. For runtime clients, use the browser-based Web client of WebSphere MQ Workflow. Do not use the standard, ActiveX-based Windows runtime client and do not implement a custom runtime client by using the C, C++ or ActiveX APIs. For human-facing activity implementations, use or customize the JSP-based Web client to implement JSPs for activities related to users. Do not use the program execution agent for such activities. For Automatic activity implementations these best practices vary based on the platform. Use user-defined program execution server-based implementations for all platforms. Do not use the program execution agent on distributed platforms. On z/OS invoke legacy IMS WebSphere InterChange Server applications using the program execution server invocation mechanism and data mapping. Do not use the program execution server container API for such activities. When migrating APIs, only use the Java API.

Section

Migration: WebSphere Business Integration Server Foundation



This section will cover WebSphere Business Integration Server Foundation V5.1.1

WebSphere Business Integration Server Foundation migration

- Support is available beginning with WebSphere Integration Developer V6.0
- Prepare your environment
 - Details are available in the WebSphere Integration Developer 'help' topic.

Migrating source artifacts to WebSphere Integration Developer from WebSphere Studio Application Developer Integration Edition

- Two steps
 - Migration Wizard
 - Part of WebSphere Integration Developer
 - Based on WebSphere Business Integration Server Foundation 5.1 service projects and **not workspaces**
 - Designed to migrate one service project at a time.
 - Migrate BPEL service with different protocols/bindings through SCA (inbound)
 - Create SCA components for BPEL partner services (outbound)
 - Manual Fix-up
 - Complete the artifact migration and rewire the migrated artifacts using the tools available in WebSphere Integration Developer (such as assembly editor) for non-BPEL service project migration.



Migrating from WebSphere Business Integration Server Foundation V5.1 requires a preparation step, which involves manually moving any dependent utility applications or jars, meaning the non-service projects. The steps for this are listed on the next slide. The migration wizard primarily migrates business process artifacts, but it may be run for any V5.1 service project. It migrates the business process artifacts, including the BPEL Java™ snippets where possible, from the WebSphere Business Integration Server Foundation V5.1, to the WebSphere Process Server V6 supported BPEL specification. It also creates a SCA component for each migrated business process. Since it is only operating against the service projects, the non-service projects must be imported into the workspace manually before the wizard is run.

Migrated BPEL services are now fully re-wired in version V6.0.2. All non-BPEL services still require manual re-wiring of the partner services and protocols.

Keeping the SCA module the same name as the V5.1 service project will reduce the amount of post migration fix-up due to the class path and project dependencies.

The Migration Wizard only handles source artifacts and not application binaries. See the description of the limitations in the WebSphere Integration Developer on-line help.

Migration – prepare, migrate and fix-up

- Turn off '*automatic build*'
- Use a new workspace
- Copy all *non-service* projects to the new workspace
- Open WebSphere Integration Developer on the new workspace
- Import the *non-service* project
 - ▶ File -> Import -> Existing Project into Workspace
- Fix the classpath to add the JRE and WebSphere Process Server libraries
- Import all the service projects from a location outside the new workspace, using the migration wizard.
- Ensure that all the .wsdl and .xsd files referenced by the .bpel files are accessible in the new workspace

This is a high level outline of what must be done to migrate existing WebSphere Business Integration Server Foundation V5.1.1 artifacts to WebSphere Process Server V6.

Details for completing these steps can be found in the WebSphere Integration Developer help, in the preparation step for the WebSphere Business Integration Server Foundation migration.

Migration wizard

- The migration wizard does these steps:
 - ▶ Creates a new business integration module
 - ▶ Migrates the service project's class path entries to the new module.
 - ▶ Copies all the WebSphere Business Integration Server Foundation source artifacts to the new module
 - ▶ Migrates the BPEL extensions in the WSDL files
 - ▶ Creates an SCA component for each .bpel process
 - ▶ Generates a monitoring *.Mon* file for each BPEL process to preserve the default monitoring behavior from WebSphere Business Integration Server Foundation
- **Details available in article:**
http://www.ibm.com/developerworks/websphere/library/techarticles/0509_ityengar/0509_ityengar.html

The steps performed by the WebSphere Business Integration Server Foundation migration wizard are listed here. The link provides detailed information related to migrating from WebSphere Business Integration Server Foundation V5.1 to WebSphere Process Server V6.

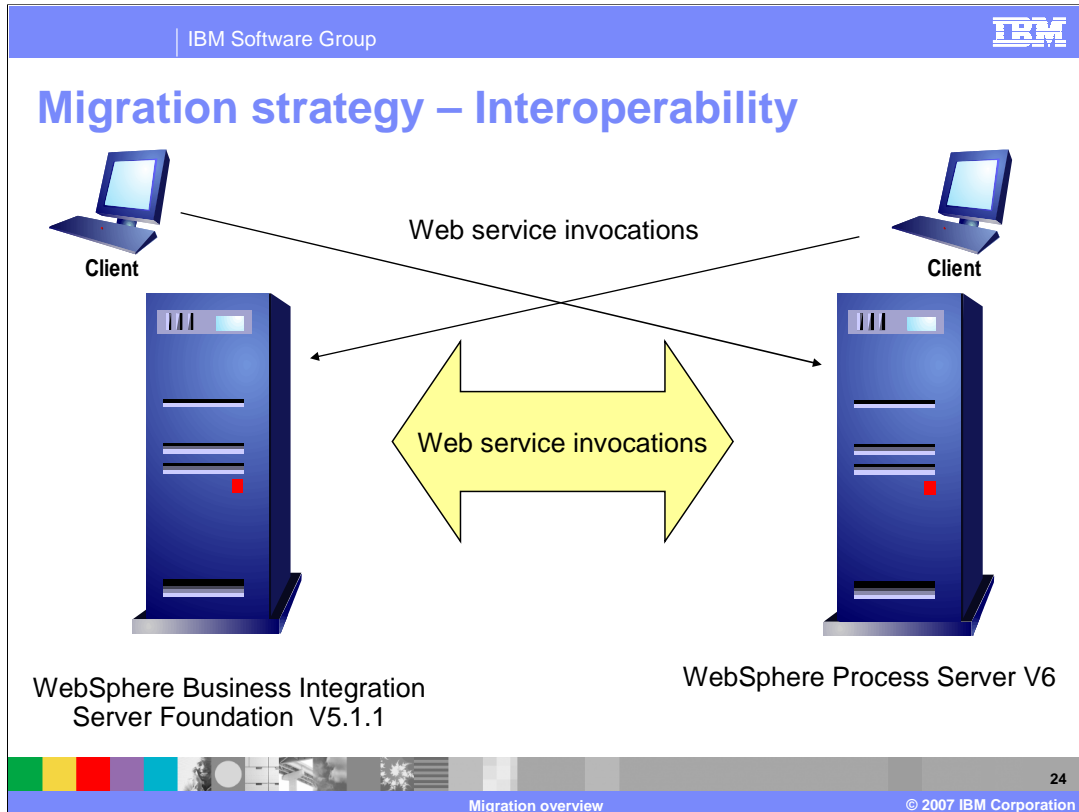
Migration wizard – Hints and tips

- Using the wizard
 - ▶ When naming the new destination module, give it the same name as the source service project.
 - This will reduce the amount of fix-up required due classpaths and project dependencies.
 - ▶ From the migration options, choose:
 - 'Preserve original BPEL Java snippet code in comments'
- Migration strategy
 - ▶ Plan on maintaining both the WebSphere Business Integration Server Foundation V5.1.1 and the WebSphere Process Server V6 systems until the V5.1.1 artifacts are completely migrated.

When using the migration wizard, it is necessary to preserve the BPEL Java snippets. This is because WebSphere Business Integration Server Foundation V5.1 uses the Web Services Invocation Framework to perform service invocations. The data sent is stored in a WSIFMessage, whereas the WebSphere Process Server programming model uses SCA to perform service invocations, where data being sent is stored in a business object. Each model has a different set of APIs for invoking a service and accessing and manipulating the data.

The BPEL Java snippets are specific to the programming model and data type used by the underlying platform. Although the migration wizard migrates the V5.1 BPEL Java snippets as much as possible, manual rework could be required for Java code that manipulates complex data types. This is why it is a good idea to preserve the existing Java snippets as comments in the migrated Java snippet. This allows you to see exactly what the V5.1 snippet did, enabling you to manually fix-up the complex snippets after migration.

Migrating existing applications can be very complex and requires detailed plans for switching over from one system to the next with considerations for dual maintenance and development



Interoperability between the two different systems is achieved using Web Services. It is possible that some WebSphere Business Integration Server Foundation V5.1.1 applications are never migrated but still require interoperability with WebSphere Process Server V6. If all WebSphere Business Integration Server Foundation V5.1.1 applications are migrated to WebSphere Process Server V6, then the WebSphere Business Integration Server Foundation V5.1.1 system can be retired.

Migration – Inbound and outbound services

- After the migration wizard is finished
 - ▶ BPEL services will be re-wired
 - ▶ Several manual steps required for non BPEL services.
- Export and import definitions depend on the transport and protocol used for inbound and outbound (partner links) services of the WebSphere Business Integration Server Foundation business process.
 - EJB (always selected)
 - IBM Web service (SOAP/JMS)
 - IBM Web service (SOAP/HTTP)
 - Apache Web service (SOAP/HTTP)
 - JMS

With WebSphere Business Integration Server Foundation there are five different ways to define the inbound and outbound partner links. The bindings are specified at the time the deployment code is generated. In the new SCA programming model, the association with the partner references and the kind of binding to use are managed in the Assembly Module with the Imports and Exports.

After running the enhanced migration utilities in WebSphere Integration Developer V6.0.2, the BPEL service is fully re-wired. The inbound SCA protocols and bindings and the SCA component for BPEL partner services for outbound are wired correctly. The WebSphere Application Developer Integration Edition V5.1 non-BPEL services that can be invoked externally still require manual migration.

All version 6.0.2 enhancements are made to the migration utilities only. There are no changes in the existing wizard.

Migration - Limitations

- Must read!
- Based on service projects and **not workspaces**
 - ▶ One service project at a time
- Does not migrate application binaries
 - ▶ Only source artifacts found in the WebSphere Business Integration Server Foundation service project
- Multiple replies for the same operation, is not supported
- BPEL java snippets
 - ▶ WSIFMessage metadata APIs
 - ▶ EndpointReference/EndpointReferenceType APIs
 - ▶ Complex types with duplicate names
 - ▶ Complex types with local names identical to java classes in the java.lang package
 - ▶ Read-only BPEL variables
 - ▶ Many-valued primitive properties in complex types
 - ▶ Instantiation of generated classes representing complex types

Be aware of the limitations before you start the migration. Migration based on service projects requires that only one project can be done at a time. The binaries are not compatible and must be rebuilt. The BPEL Java snippets do not support the coding types that are listed here. Details can be found in the WebSphere Integration Developer help, in the Limitations step for the WebSphere Business Integration Server Foundation migration.

Deprecated features

- Information center
 - ▶ There are many deprecated features to be aware of which are documented in the information center.

<http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp>

Installing and migrating
[Installing](#)
[Migrating to WebSphere Process Server](#)
[Process Server](#)

Contents

ibm.com: [About IBM](#) - [Privacy](#) - [Contact](#)

WebSphere Process Server for Multiplatforms, Version 6.0

- Release notes
- Product overview
- Installing
- Migrating to WebSphere Process Server

Select the book for WebSphere Process Server

Then on the right, select the topic migrating to WebSphere Process Server

27
Migration overview © 2007 IBM Corporation

Deprecated features can be found in the Information Center.

Section

Summary and references

This section will provide a summary and references.

Summary

- Source artifact migration is available today with WebSphere Process Server V6.0 and beyond for:
 - ▶ WebSphere InterChange Server
 - ▶ WebSphere MQ Workflow
 - ▶ WebSphere Business Integration Server Foundation
- Post migration 'fixup' will be required.

With WebSphere Process Server V6, source artifact migration is possible from all three converging products: WebSphere InterChange Server, WebSphere MQ Workflow and WebSphere Business Integration Server Foundation. The source artifacts are converted to the SCA programming model as they are imported into the WebSphere Integration Developer authoring tool. As with most migration efforts, when converting from an existing model to a new model, 100% conversion is not possible. The areas that cannot be automatically converted must be manually converted as a post migration task.

References

- Information center

<http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp>



Select the book for WebSphere Integration Developer

Then on the right, select the topic migrating applications

Installing and migrating
[Installing WebSphere Integration Developer](#)
[Migrating applications](#)

You can find details concerning all three migration paths in the Information Center.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WPSWIDv6_MigrationOverview.ppt



You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM Rational WebSphere

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, Other Countries, or both.

EJB, J2EE, Java, JRE, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.