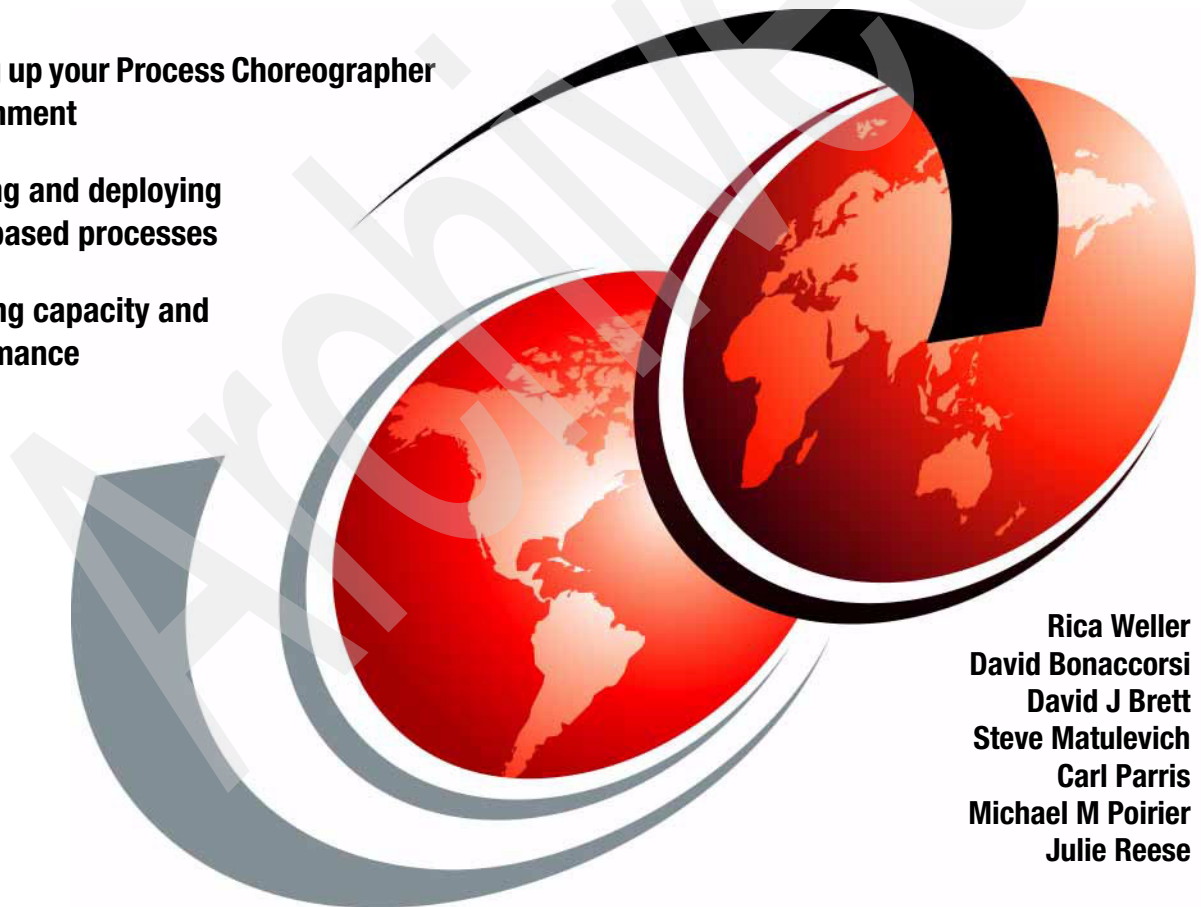


Best Practices for WebSphere Business Integration Server Foundation on z/OS

Setting up your Process Choreographer
environment

Creating and deploying
BPEL-based processes

Planning capacity and
performance



Rica Weller
David Bonaccorsi
David J Brett
Steve Matulevich
Carl Parris
Michael M Poirier
Julie Reese



International Technical Support Organization

**Best Practices for WebSphere Business Integration
Server Foundation for z/OS, V5.1**

February 2006

Archived

Note: Before using this information and the product it supports, read the information in “Notices” on page v.

First Edition (February 2006)

This edition applies to Version 5, Release 1, Modification 0 (build number W510215 PTF 15, UK05031) of WebSphere Business Integration Server Foundation for z/OS, and Version 5, Release 0, Modification 2 (build number cf50529.02 PTF 5, UK05973, UK05984, UK05986, and UK05988) of WebSphere Application Server for z/OS.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	v
Trademarks	vi
Preface	vii
The team that wrote this Redpaper	vii
Become a published author	viii
Comments welcome	ix
Chapter 1. Introducing WebSphere Business Integration Server Foundation for z/OS, V5.1	1
Chapter 2. Setting up the Process Choreographer environment	3
2.1 SMP/E issues	4
2.2 Ports and naming conventions	5
2.3 CEI and DB2 issues	6
2.4 WebSphere MQ issues	7
2.5 Security considerations before BPE install	7
2.6 BPE issues	8
2.7 CEI issues	11
2.8 Tuning considerations	12
2.9 References	14
Chapter 3. Creating and deploying business processes	15
3.1 Long-running and short-running processes	16
3.1.1 Long-running processes	16
3.1.2 Short-running processes	20
3.1.3 Long-running versus short-running implementations	21
3.1.4 Synchronous versus asynchronous behavior	22
3.1.5 When compensation is registered and when it gets triggered	22
3.1.6 Resources	23
3.2 Business Process Execution Language and Web Services	23
3.3 Versioning and binding	24
3.3.1 Versioning	24
3.3.2 Limitation of versioning	25
3.3.3 Early versus late binding	25
3.4 Audit logging	27
3.4.1 Common Event Infrastructure (CEI)	28
3.4.2 Implications of using Audit Logging	30
3.5 Staff resolution	30

Chapter 4. Summary	33
Index	35

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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 and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:


This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

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

@server®

@server®

Redbooks (logo) ™

eServer™

z/OS®

zSeries®

Cloudscape™

DB2®

IBM®

Redbooks™

RACF®

Tivoli®

WebSphere®

The following terms are trademarks of other companies:

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

Excel, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

Systems programmers and WebSphere® administrators can use this Redpaper as a guide for exploiting WebSphere Business Integration Server Foundation (WBISF) functions. The designated platform for this functional implementation is z/OS®, as there are specifics outlined here that do not correlate to details in distributed environments.

The paper provides the following guidance:

- ▶ Information about how to install and configure WBISF on z/OS, V5.1 for high performance applications
- ▶ Best practices and guidelines for creating and deploying business processes

While this paper cannot cover all the options of an install, the overall best practices should be apparent. Because WebSphere is a program product undergoing rapid enhancements, efforts are taken not to limit its function or scalability.

The paper is presented in a cookbook style that is designed to provide a working solution, not a nebulous “what if” discussion. It is not intended to replace or override education such as “WebSphere V5 Implementation Workshop” (ES685) or “WBISF Installation and Customization: WebSphere V5.1 for z/OS” (WBIZ5). The Redpaper is presented as an aid to the successful conversion or installation of WBISF.

The team that wrote this Redpaper

This Redpaper was produced by a team of specialists from around the world:

Rica Weller

ITSO Project Manager, IBM New Zealand

David Bonaccorsi

Program Director Business Process Integration, IBM USA

David J Brett

Software Group AIM Development, IBM USA

Steve Matulevich

Certified IT Specialist, Advanced Technical Support WebSphere and WBISF for z/OS

Carl Parris

STSM - zSeries performance, IBM USA

Michael M Poirier

Product Architect, IBM USA

Julie Reese

WebSphere Business Integration Services, IBM USA

Ken Rogers

Systems Programmer, Support z/OS & Tivoli®, IBM USA

Thanks to the following people for their contributions to this project:

Michael Scheible, Business Process Solutions Development lab manager, IBM Germany

Dominik Goetz, WBISF Consultant, IBM Germany

Jonas Grundler, WebSphere Solution Center, IBM Germany

John Hutchinson, Technical Support Marketing - eServer™, zSeries, IBM USA

Dieter Koenig, Architect, WBI Development Lab, IBM Germany

Richard Metzger, WBI performance, IBM Germany

Gary Puchkoff, WebSphere for z/OS Design and Development, IBM USA

Bertrand Portier, Frank Budinsky

Software Development, IBM Canada

Rich Conway

International Technical Support Organization, Poughkeepsie Center

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You team with IBM technical professionals, Business Partners, and clients.

Your efforts help increase product acceptance and client satisfaction. As a bonus, you develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at the following Web address:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this Redpaper or other Redbooks™ in one of the following ways:

- ▶ Use the online **Contact us** review redbook form found at:

ibm.com/redbooks

- ▶ Send your comments in an email to:

redbook@us.ibm.com

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. ITSO WebSphere Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Archived

Introducing WebSphere Business Integration Server Foundation for z/OS, V5.1

WebSphere Business Integration Server Foundation (WBISF) for z/OS, V5.1 expands IBM WebSphere Application Server V5.1 to provide a powerful standards-based integration platform for building and deploying Business Process Execution Language (BPEL) for Web Services (BPEL4WS) and Web Service-based composite applications. WBISF provides the BPEL4WS to link Web Services using the Service Oriented Architecture (SOA) approach. It also provides the business process choreography.

It is easy to reuse your existing infrastructure with WebSphere Application Developer Integration Edition (WSAD-IE) V5.1 to build and deploy composite applications.

BPEL4WS provides a means to link together Web service interactions to represent the flow of an organization's business process. This ultimately connects an enterprise with its clients and suppliers in real time and forms the basis for the on demand Operating Environment. In this on demand Operating Environment, the choreography of new, re-usable business functions via Web Services is the foundation of a service oriented architecture. The IBM vision of the on demand Operating Environment is to enable clients to succeed in an

environment with an unprecedented rate of change. The fundamental component of on demand infrastructure is integration, based on open standards. Open standards affects the on demand Operating Environment across the levels of automation, integration, and virtualization. Each of these elements leverage open standards specifications in order to achieve their objectives. Open standards are the key element of obtaining the flexibility and interoperability across heterogeneous systems.

SOA is an approach to architecture that must be applied to systems and integrations that specifies a set of principles and techniques that encourage the encapsulation and modeling of reusable functions and processes. BPEL4WS has emerged among other trends in Web Services and increasingly supports the modeling concepts of SOA. Recurring elements (security, service level, monitoring, and so on) can be shared across applications to provide horizontal services to decouple these reusable application components.

Process Choreography is a way of modeling business activities that consist of several steps as a service flow. It brings a service-oriented approach to both application development and workflow modeling. The enormous potential of SOA comes from the synergy made possible by exposing application components as services and the ability to model business processes and workflows and as service flows. Given an abstraction, an understanding of how activities and information flow through an organization, you can model aspects of the work separately as services and model the business process by choreographing the flow of control and data flow among the services. The process choreographer is a feature of Integration Edition that helps you design activities involving J2EE™ components, Web Services, existing applications and systems, and even human activities in dynamic workflows. The key concept of process choreography is to separate the flow of business functions from the actual implementation of these functions. In SOA, each function implementation is a service and a building block to be used within any number of processes.

Setting up the Process Choreographer environment

This chapter contains guidelines and best practices to set up the Process Choreographer environment. It provides details about the following:

- ▶ SMP/E issues
- ▶ Ports and naming conventions
- ▶ CEI and DB2 issues
- ▶ WebSphere MQ issues
- ▶ Security considerations before BPE install
- ▶ BPE issues
- ▶ CEI issues
- ▶ Tuning considerations

The Information presented in this chapter is based upon the build number W510215 PTF 15 (UK05031) of WebSphere Application Server for z/OS V5.1 (Program number 5655-I35) and build number cf50529.02 PTF 5 (UK05973, UK05984, UK05986, and UK05988) of WebSphere Business Integration Server Foundation for z/OS (Product number 5655-L85, FMID HWBI510).

The APAR/PTF table for WebSphere Application Server for z/OS is available at the support tab of the following Web address:

http://www.ibm.com/software/webservers/appserv/zos_os390/

The APAR/PTF table for WBISF for z/OS is available at the following Web address:

<http://www-1.ibm.com/support/docview.wss?rs=404&uid=swg21170385>

2.1 SMP/E issues

Consult the *Program Directory* and *Getting Started* from the WBISF library, available at the following Web address:

<http://www-306.ibm.com/software/integration/wbisf/editions/zos/library/documentation.html>

Before you attempt to install the product see the categories *Product Overview*, *Planning*, and *Installing* of WBISF for z/OS in the Information Center (InfoCenter) at the following Web address:

<http://publib.boulder.ibm.com/infocenter/wasinfo/v5r1/index.jsp>

Then follow the usual steps of RECEIVE, APPLY and ACCEPT, and MIGRATE into the driving system according to the description in the InfoCenter and the publications. Additionally consider the following guidelines:

- ▶ The WBISF files have an identifier of SBBZ*, much like the WebSphere Application Server files have SBBO* as High Level Qualifier (HLQ). That makes it easier to identify them. For the installation and configuration, it is necessary to have both families of files have matching high-level qualifiers.
- ▶ Keep the output from the jobs run, as a result of the dialogs, for convenience until the new layer of WBISF is running to expectations. This can be seen as unnecessary; however, referring back to specifics can be a huge help as configuration gets finalized. Even if the workbook design pages are tracked, keeping the output until after the new configuration is declared a success has advantages. Saving the install instructions as well is a good way to make sure all the tasks are run in the correct order.
- ▶ In the SMP/E installation steps, the APPLY goes against some directory paths. The migration and install processes need a starting directory, for instance **/Service** that is identified in the dynamic data definitions (DDDEFs) for SMP/E. If the functioning directory **/usr/lpp/zWBISF/V5R1M0/** is defined off the **/Service** directory, it allows an HFS to be mounted for SMP/E work and to be easily migrated into use in a controlled manner. Auditors seem to favor this method.

An alternative method that auditors are less pleased with is mounting the HFS at unique mount points and changing symbolic links to allow correct traverse of the tree for files. So the DDDEFs in SMP/E do not need to be changed.

This philosophy can be extended to datasets used for the driving system as well.

2.2 Ports and naming conventions

Most of the installation and configuration planning has to do with TCP/IP port definitions. Make sure that the desired ports are not protected or used by more than one server. Some of that is guarded against in the installation dialogs.

Most of the pre-install workbook pages (worksheets/customization panels) provide great quick information, as well as prevent subtle but major problems. They also provide information to help you document your environment, and set up a spreadsheet or table depicting ports assigned to various port types for each address space.

White papers and Excel® spreadsheets that can help you with port definitions and naming conventions and provide installation and customization tips and examples are available at the following TechDoc Web address:

<http://www.ibm.com/support/techdocs>

See WebSphere for z/OS - Configuration Planning Spreadsheet, PRS1331 and *WBISF for z/OS V5.1 Installation & Customization Tips and Examples*, WP100612.

Table 2-1 shows an example spreadsheet for port definitions.

Table 2-1 Port definition example

Port type	Daemons	Deployment Manager	Node agent	AppServer 1	AppServer 2
JMX™		9810	9820	9840	9850
DRS		9811	9821	9841	9851
Bootstrap		9812	9822	9842	9852
ORB IIOP	9800	9812	9822	9842	9852
ORB SSL	9801	9813	9823	9843	9853
Discovery		9814	9824		
Multicast			9825		
HTTP		9818		9848	9858
HTTP SSL		9819		9849	9859

2.3 CEI and DB2 issues

When Installing CEI with DB2® V7 you must use the type 4 XA driver and run the DB2T4XAIndoubtUtil utility to create the “indoubt” tables. This utility is packaged in the db2jcc.jar file, and it is located in the DB2 install directory under jcc/classes. Make sure this file is in your CLASSPATH environment variable. The user of this utility must have DB2 SYSADM authority. Remember to install all stored procedures.

Read the README text file located in the DB2 install directory under jcc. This file contains valuable information including instructions on how to create a JDBC™ trace. Many problems are averted by reviewing its contents.

After installing the JDBC driver, ensure that you bind the packages with DB2. This is a common pitfall for those who are not familiar with installing DB2.

From the AdminConsole go to **Resources** → **JDBC Providers**, making sure the scope is **Deployment Manager node**. Then click **Apply**.

Particulars with DB2 V7 are valid for DB2 V8 as well; Table 2-2 lists the specifics.

Table 2-2 DB2 variables to be defined

Variables and parameter	Examples
Subsystem name	DSN8
Location name	ZOSLOC
DASD volumes for table spaces	WBI001
HLQs for the tablespace dataset names	DB2V801
database name	BPEDB
storage group	BPESG

For some DB2 commands and definitions you need to have SYSADM authority. To grant SYSADM authority to the userid assigned to servant regions issue the following command, assuming that the userid is ZWBISF1:

GRANT SYSADM TO ZWBISF1

2.4 WebSphere MQ issues

Following is the preferred method for WebSphere MQ installation, especially if you plan on using it as an external JMS provider:

1. Install WBISF in the Deployment Manager.
2. Install WBISF to an Application Server within a WebSphere cell.

When you install WebSphere Application Server for z/OS you have the option to install embedded WebSphere MQ, also known as Integrated Java™ Messaging Service Provider (IJP). Use the following instructions to install embedded WebSphere MQ:

1. Install WBISF to an Application Server.
2. Create a Deployment Manager.
3. Add WBISF to the Deployment Manager.
4. Federate the Application Server into the Deployment Manager.

For a decision about the way you choose to install WBISF, also consider Global Security described in 2.5, “Security considerations before BPE install” on page 7.

For a speedier install, we recommend that you not install the provided sample programs. If desired, install them later but to do so in the beginning can prolong the install time extensively.

You must perform the server configurations through the JMS set up, which is usually option 4 of the BBOWSTRT exec's main panel.

2.5 Security considerations before BPE install

Prior to running the install, you have the choice of turning off Global Security. If Global Security is left on, you can resolve all security issues at this time. However, you can turn Global Security off and proceed with the installation and then turn it on after running the bpeconfig.jacl script to resolve your security challenges. Refer to the IBM Redbook *WebSphere Application Server for z/OS V5 and J2EE 1.3 Security Handbook*, SG24-6086, for more information about global security.

Another reason for choosing the alternative configuration (Unfederated AppServer) is that when you install WBISF on the AppServer that was federated into the Deployment Manager the AppServer needs to communicate with the Deployment Manager over a SOAP connection/port. If you have difficulty with this as a result of incorrect security setup, then the Deployment Manager will not be updated and synchronized correctly. At this point you should turn off Global

Security and unfederate the AppServer from the Deployment Manager. Then continue with the installation.

You may now install the CEI Feature on the Deployment Manager. DB2 V8.1 is supported on z/OS; Otherwise, you may defer this until later.

Do not start the initial installation and configuration unless all AppServers, all Node Agents, Daemons, and the Deployment Manager are stopped.

For ease of security and internal processing, allow RACF® IDs assigned to WebSphere and WBISF to have read access to the BPX.SUPEUSER profile in the Facility Class.

To prevent some security issues, make the octal permission bits for the program HFS to be 775 throughout. This sounds like an exposure, but to prevent any intrusion, the file system should be mounted read-only. This allows sharing of the HFS across z/OS systems as well.

If you receive an SQLCODE=-553, SQLSTATE=42503, error tokens=Server name, then you have an authorization problem. In most cases the user ID assigned to the server or Deployment Manager is not authorized to set sqldid function. Grant SYSADM to the user ID in RACF.

Edit the JCL for the xxxASR and xxxASRS procedures adding the parameter: HEAP(,,,FREE) to the PARM field, like in Example 2-1.

Example 2-1 JCL example with HEAP parameter

```
//BBOASR2S PROC IWMSSNM='BBOASR2A',PARMS='-ORBsrvname '  
// SET CBCONFIG='/WebSphere390/CB390'  
// SET RELPATH='controlinfo/envfile'  
//BBOASR2S EXEC PGM=BBOSR,REGION=0M,TIME=NOLIMIT,  
// PARM='HEAP(,,,FREE)/ &PARMS &IWMSSNM'  
//* PARM='/ &PARMS &IWMSSNM'
```

2.6 BPE issues

When installing WBISF on the AppServer, ensure that the Deployment Manager is running. If you have not installed CEI on the Deployment Manager, then do not select this feature on the AppServer. Again, you may defer installing this feature.

Make sure the Deployment Manager is up and running and the AppServer and NodeAgent are stopped before submitting the job to add WBISF to the AppServer. This is often forgotten and leads to an error.

Following are some things that can go wrong during installation:

- ▶ Authorization problems.
Both z/OS and USS files.
- ▶ Most of these jobs are to be run with the cell's administrator ID, not super user.
- ▶ AppServer up when it should not be.
You will get a failure with a clear error message.
- ▶ Deployment Manager up when it should not be.
You will get a failure with a clear error message.
- ▶ Jobs not run in the correct order.
- ▶ Features not selected correctly.
- ▶ Space exhausted in config HFS or /tmp
Clean out /tmp/ directory frequently.
- ▶ Same files or directories owned by another administrator.
This can occur if you ran the job with a different user ID, or if there are multiple cells on the same system.
- ▶ Incorrect values entered in ISPF dialogs.
- ▶ Jobs interrupted.
Usually you can re-submit.

The bpeconfig.jacl script is interactive. Take all defaults and follow the hints and tips for the BPE Container installation as noted in the Whitepaper *WBISF for z/OS Installation & Configuration Tips and Examples*, WP100612.

Use a telnet session before using OMVS to navigate around the HFS and for submitting jacl and shell scripts. OMVS has a two line limit and is cumbersome to use when submitting jacl scripts that require several arguments to be passed on the command line that may exceed the two line limit.

When installing the BPEWebClient, there is a known problem with the bpelib.jacl script that is called from the bpeconfig.jacl. It limits the selection of installation nodes. This is addressed in APAR PQ89316. One way to circumvent this issue, until the APAR is published, is to install under the alternate configuration (Unfederated AppServer).

If you receive the error SQLCODE =-204, SQLSTATE = 42704

The object identified by name is not defined in the DB2 subsystem. This SQLCODE can be generated for any type of DB2 object. This is caused by running the bpeconfig.jacl script without specifying the servant region's user ID in the Schema qualifier field. You must correct this value and rerun the script.

A common problem with bpeconfig.jacl script in a system with multiple cells:

WASX7017E: Exception received while running file sample/bpeconfig.jacl
exception information: com.ibm.bsf.BSFException: error while eval'ing Jacl
expression (no expression displayed).

This is corrected by deleting the /tmp/bpeconfig/ directory. The directory
/tmp/bpeconfig/ was "owned" by a different user ID.

Remove extraneous jar files from the Deployment Manager's CLASSPATH:

This is a common error, and it is often overlooked. A jar file left in this
CLASSPATH has often caused the error Unable to load T2 Native
Libraries to occur, and failed Test Connection to the data source.

Set JDBC driver variables: Go to **Environment** → **Manage WebSphere
Variables** (Set the scope to the AppServer's node but leave the **Server** box
blank.):

DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH= /usr/lpp/db2810/jcc/lib

DB2UNIVERSAL_JDBC_DRIVER_PATH = /usr/lpp/db2810/jcc/classes

Note: The Deployment Manager must have access to all the Process
Choreographer databases that are used by business process containers in the
cell.

Ensure that the symbolic links of the DB2 load libraries are included in either the
link list or steplib of server jobs. If using steplib all members of the steplib must
be APF authorized. If using link list, the preferred method, only the required DB2
members need to be APF authorized.

To display the link list use the following command from the SDSF LOG:

/D PROG,LNKLST This displays the link list.

/D PROG,APF This displays the APF authorized members in the link
list or steplib.

For the same information through the console, the commands are the same
without the initial forward slash (/).

Run the bpeconfig.jacl script in order to get an understanding of the WebSphere
resources used by Process Choreographer. The utility walks you through the
Installation steps documented in the InfoCenter. The utility can create a complete
configuration when using Cloudscape™ and Embedded Messaging. When using
external subsystems the utility only creates the definitions required. It may
require additional tailoring that would be provided by your DB2 and WebSphere
MQ subsystem administrators.

Note: jacl scripts are run in ASCII encoding and cannot directly be edited using vi. Special considerations must be made when scripts alter z/OS files to ensure that they are updated in the correct encoding.

If the bpeconfig.jacl script did not finish completely and a Signal 29 error was encountered, correct the problem by changing the CPU Time and rlimit environment variables. The SIGNAL 29 should be managed on a permanent basis using the TIME= JCL parameter. This goes in your logon proc or the EXEC statement if this is a batch run. The exec below takes the current time setting and replaces it with the maximum setting.

If the max setting is not enough, then change the max setting. Max setting is either the TIME= JCL parameter or CPUTIMEMAX setting in RACF for the user profile. Use the ALTUSER command to alter. See the following Web site:

http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/bpxzb251/5.3.12.3?ACTION=MATCHES&REQUEST=RLIMIT_CPU&TYPE=FUZZY&SHELF=&DT=20041214152236&CASE=&searchTopic=TOPIC&searchText=TEXT&searchIndex=INDEX&rank=RANK&ScrollTOP=FIRSTHIT#FIRSTHIT

Additional information about setting the CPU Time and max memory on a user ID can be found at the following Web site:

http://64.233.187.104/search?q=cache:_WfJYbs42_gJ:www.schunk-associates.com/presentations/Share095/S2562pea.PDF+ALTUSER+%2BUNIX+%2BCPU&hl=en&client=firefox-a

Set MaxThreads to at least 10000.

2.7 CEI issues

Special considerations must be made for enabling messaging connections pools. While increasing the amount of application work in an AppServer you need to increase the number of messaging connections for your CEI connection factories.

When creating own events ensure you do not overuse them in all activities as this can lead to performance and debugging issues.

You find extensive information about CEI on the developerWorks web site and in various Redpapers and Redbooks. For example:

- ▶ Andy Brodie and Amanda Watkinson, *The Common Event Infrastructure: From technical preview to production*, 27 April 2005, at:

http://www-128.ibm.com/developerworks/websphere/library/techarticles/0504_brodie/0504_brodie.html

- Redpaper *WebSphere Business Integration Server Foundation Using the programming API and the Common Event Infrastructure*, REDP-3915, at:
<http://www.redbooks.ibm.com/abstracts/REDP3915.html?Open>

Also see Chapter 3.4.1, “Common Event Infrastructure (CEI)” on page 28 for more information.

2.8 Tuning considerations

Due to design and implementation the biggest potential for performance tuning can be found within the applications. Use WSAM for the fastest methods of application and design improvements. Performance documentation for WebSphere Application Server for z/OS is just as valid for WBISF for z/OS. Search for the following reference number, 1172361, at the following Web address to get to the documentation, hints and tips, and Technotes that can assist you in tuning your system for the best performance:

http://www-306.ibm.com/software/webservers/appserv/zos_os390/support/

z/AAP

z/OS V1.6 is the minimum level for z/AAP utilization. WebSphere for z/OS V5.1 and V6 both use Java V1.4.2 which can utilize z/AAPs.

Messaging throughput

Messaging throughput can be influenced in several areas. Following are some of those areas:

- Database connections
- WebSphere MQ configuration
- Application usage of CEI feature

A console dump may be useful in order to obtain a view of the resources, threading, and bottlenecks for both the controller and servant address spaces.

To review the number of worker threads in use issue the following SDSF LOG Dump Commands:

```
/dump comm=(WebSphereStackTrace) (dump consolecommand)
/05,ASID=(3e),SDATA=(RGN,TRT,CSA,NUC,PSA,GRSQ,LPA,SQA,SUM)
```

After the dump completes, bring it into IPCS using the following:


```
IP SETDEF ASID(X'3E')
IP SUMM FORMAT ASID(x'3E')
IP VERBX LEDATA 'NTHREADS(*)'
```

or use dump reading tools as described in the Redbook *WebSphere for z/OS V5 Problem Determination*, SG24-6880.

JVM tuning

This is not so much a tuning issue as a compatibility one. With levels of WebSphere at W510200, and JAVA at the 1.4.2 level, there was a demonstrated performance problem with the provided Trade3 application. After some 20 clients ran for half an hour or so, there were system abends (S538) where the number of concurrent RESERVE requests exceeded an address. No tuning or performance adjustments could be found to prevent this from occurring. More insight and clarification is needed.

The 64-bit Java SDK on z/OS is not supported by WebSphere for z/OS V6.

Operating system tuning

Define 1000 cylinder page dataset per defined server. So if Server 1, 2, A, and B are defined, make sure that either through PAGEADD command or PARMLIB definitions at IPL, the equivalent of 4000 cylinders of page spaces are available.

The best might be that each server has a corresponding PAGESPACE: three datasets are available at 1350 cylinders each for 4 servers.

Make the following parameter adjustments in the BPXPRMxx member of z/OS for the USS processing use for this systems' IPL:

```
MAXASSIZE(1073741824)
SUPERUSER(???????) /* Any system defined USER with UID=0 */
MAXSHAREPAGES(524288)
MAXPROCSYS(800)
MAXCORESIZE(16777216)
MAXPROCUSER(600)
MAXFILEPROC(10000)
MAXTHREADS(10000)
MAXTHREADTASKS(5000)
MAXQUEUEDSIGS(10000)
```

2.9 References

- ▶ Review the Whitepaper *WBISF for z/OS Installation & Configuration Tips and Examples*, WP100612, published by the Washington Systems Center at the following Web address:

<http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100612>


- ▶ Review Technote *Performance Monitoring and Tuning*, reference number 1172361, at the following Web address:

http://www-306.ibm.com/software/webservers/appservers/appserv/zos_os390/support/

- ▶ Review related IBM Redbooks at the following Web address:

<http://www.redbooks.ibm.com>

- *WebSphere Business Integration Server Foundation V5.1 for z/OS*, SG24-6382
- *WebSphere Business Integration Server Foundation V5.1 Handbook*, SG24-6318
- *WebSphere for z/OS V5 Problem Determination*, SG24-6880
- *WebSphere Application Server for z/OS V5 and J2EE 1.3 Security Handbook*, SG24-6086



Creating and deploying business processes

This chapter introduces business process concepts and a number of ways to construct them in a comprehensive and efficient manner. The discussion includes the business process itself, compensation, audit logging (the use of the Common Event Infrastructure), versioning, the underlying process language, and Web Service invocation.

3.1 Long-running and short-running processes

A business process is a collection of activities or tasks that takes place in an organized manner. The execution of a given activity may depend on the result of one or more previous activities, or it may just depend on the completion of one or more activities. We sometimes need to impose a time limit on an activity and perhaps take corrective action when the expiration is reached. Other common needs in a business process include the following:

- ▶ Human interaction or intervention
- ▶ Notification or alert to a human or other application that something has or has not occurred
- ▶ Notification from a human or other application that something occurred (process is event driven)
- ▶ Interaction with external or legacy applications or services
- ▶ Compensation to undo the effects of prior actions when a process activity fails

This section introduces the types of processes that the Business Process Choreographer supports. Business processes can be one of two types, either long-running or short-running. This section describes these types, and then discusses how to determine which type to implement based on the requirements. We also discuss the impact on performance, transactional behavior, and compensation.

3.1.1 Long-running processes

Long-running business processes may include human interaction (staff activities), mid-process event triggering, and compensation support as well as automated activities. The life of the process may span long periods of time, even months or years. When a process instance is started or the state of an activity in the process changes, the information is persisted in a database.

Note: Long-running processes are often referred to as interruptible, macro flows, or stateful.

Short-running processes have a “state”, although not a persistent state.

Use the following steps to mark a business process as long-running in the WebSphere Studio Application Developer Integration Edition (WSAD-IE) Eclipse tool:

1. Select the start activity of the process.

2. Check the **Process is long-running** box in the Server properties of the process, as shown in Figure 3-1.

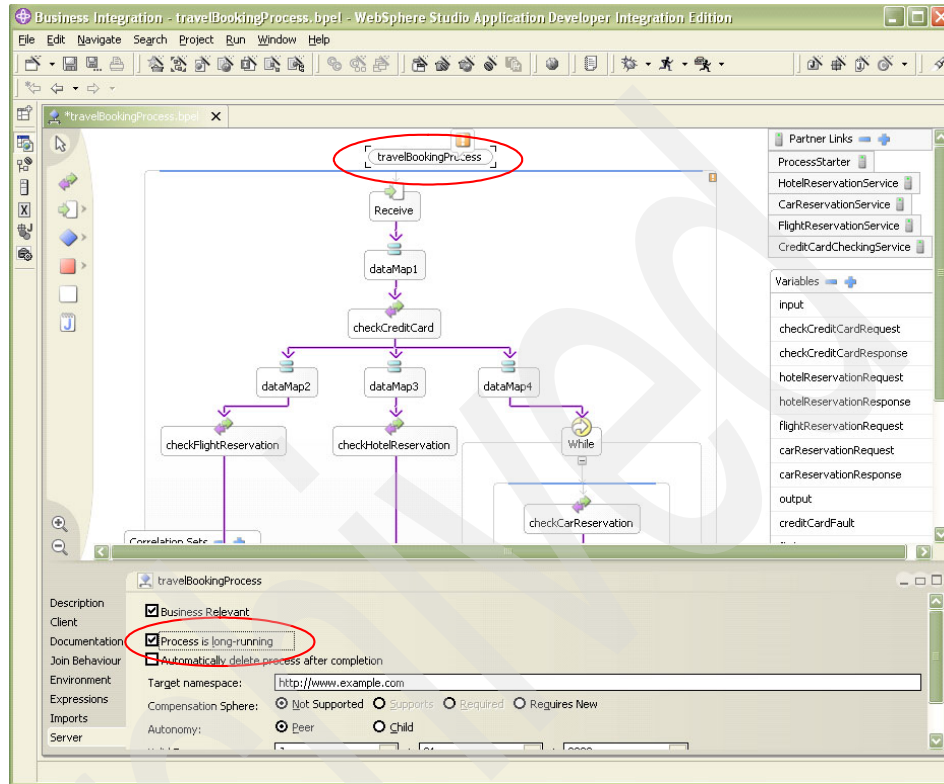


Figure 3-1 Marking a business process as long-running

Human Interaction

Some business processes require activities that are performed by a human staff member in an organization. The most common reasons for this are data collection activities, approval activities, and manual intervention activities needed to correct an error situation.

Event Triggering

A business process may be started when an external event occurs, or the process may be paused in order to wait for an external event to occur. Typically when a pause activity is included, it is defined with a time limit so that non-responses may be tracked or cause an alert to be issued. The alert can be an e-mail, a human interaction activity, or an alternate automated activity.

Transactional Behavior

Since a long-running process must be aware of the state of its activities in the instance, it saves the state information to a database. This means that the process will not run as a single transaction and that actions associated with the activities are committed along the way. Generally speaking, each activity is performed as its own transaction. You can request to adjust the transactional behavior in the WSAD-IE tool for some activities by selecting the activity and choosing a behavior in the Server properties.

Transactional Behavior: ☐ Commit Before ☐ Commit After ☒ Participates ☐ Requires Own

- ▶ **Commit Before** is guaranteed to cause a commit to occur immediately before the activity starts.
- ▶ **Commit After** is guaranteed to cause a commit to occur immediately after the activity completes.
- ▶ Mark an activity as **Participates** if it does not require a commit to occur either before or after it. **Participates** is a means to group one or more activities into a single transaction. However, it is not guaranteed that the Process Choreographer will execute the participating activities within a single transaction. It is treated as a suggestion to the Process Choreographer. Additional commits may occur even though you requested a group of activities to coexist within the same transaction.
- ▶ Use **Requires Own** when you want an activity to run within its own transaction. It also is treated as a suggestion—the Business Process Choreographer does not guarantee that it will run in its own transaction.

Parallel paths may execute simultaneously in a long-running process. To ensure this, specify **Commit After** for the activity that precedes the split. Alternately, specify **Commit Before** for the activities at the beginning of each of the split paths, as shown in Figure 3-2 on page 19.

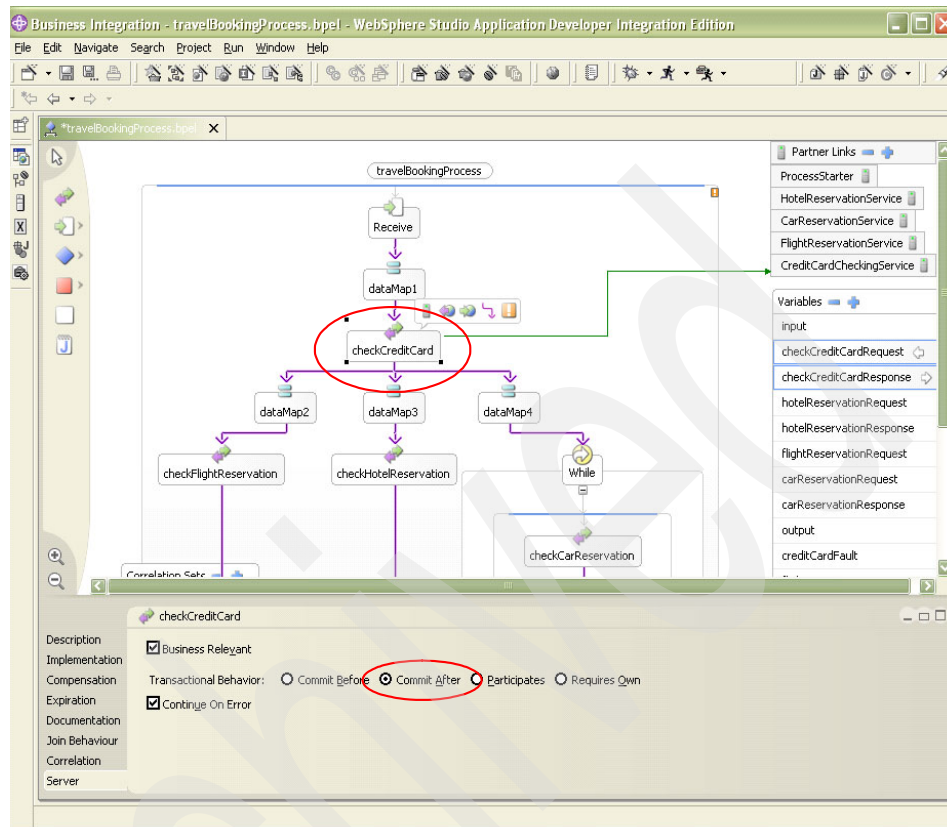


Figure 3-2 Specifying the Transactional Behavior of an activity

The Business Process Choreographer executes the paths in a random order—you cannot predict the execution order of the activities other than the guarantee that they will run in the order of the control links or as specified in the “sequence” where we do not have control links.

Compensation

Compensation provides a means to “undo” both transactional (participates in a two-phase commit) and non-transactional (does not participate in a two-phase commit) activities. A compensation activity should be implemented for every activity where an action or previous action needs to be reversed.

Note: The compensation is not necessarily an undo or a reversal of a previous action. It is any action that restores appropriate balance to the situation after a failure has occurred.

For example, a client places an order for an item and the cost of the item is billed to a credit card. During the fulfillment process, it is discovered that the product was discontinued and the order is automatically canceled. A compensation activity must exist to issue a credit to the client for the amount that was already billed.

There is currently no compensation support for asynchronous JMS activities. You can implement a work around by adding a second service activity after the asynchronous JMS activity. Let the primary service be a dummy service, and add your compensation there.

3.1.2 Short-running processes

Short-running processes consist of automated tasks only and are expected to complete in a very short period of time. They run in a single thread so that if any one activity fails, the entire process fails. Information about an instance of the process or information about the state of the activities is not persisted to the database.

Note: Short-running processes are often referred to as non-interruptible, micro flows, or stateless.

To mark a business process as short-running in the WSAD-IE tool, deselect the ☐ **Process is long-running** box in the process properties after selecting the start activity of the process.

Human Interaction

Human interaction is not supported in short-running processes.

Event Triggering

A business process may start when an external event occurs, but the process may not pause in order to wait for an external event to occur.

Transactional Behavior

Short-running processes execute in a single thread. A single thread can participate in a single transaction only, so there is no opportunity to specify the transactional behavior as there is with long-running processes.

A single thread can only execute the activities in a sequential manner. Parallel execution of parallel paths in the business process does not occur. Like long-running processes, the Business Process Choreographer executes the paths in a random order—you cannot predict the execution order of the activities other than the guarantee that they will run in the order of the control links.

Always include a time limit, expiration, for a synchronous activity. This is so that short-running processes are not left hanging for an indefinite amount of time.

Compensation

Since short-running processes run in a single thread (in one rolled back transaction or without a transaction), all activities, as a whole, are committed successfully or are rolled back. You must consider compensation in cases where asynchronous fire-and-forget activities are implemented or the short-running process is participating in the compensation of a long-running process.

For example, as part of the process, an e-mail is sent to the client to notify that the order was accepted. If the order is canceled due to a process failure, you might want to send an update e-mail to the client.

If the activity represents an external service that is transactional, be sure to check the ☒ **Activity Takes Part in Transaction** option in the Compensation properties of the invoke activity within the WSAD-IE tool.

Compensation in a short-running process, within a short-running process, is called if you set the `setRollbackOnly` flag. Compensation in a short-running process, called from a long-running process using the process binding, can be triggered by either the short-running or the long-running process. There is currently no compensation support for asynchronous JMS activities—see the work around listed in Long-running processes “Compensation” on page 19.

3.1.3 Long-running versus short-running implementations

The execution of short-running processes is quick and efficient due to the lack of performance overhead. As you may recall, neither the process instance data nor the state data is persisted in a short-running process. The process runs in memory and in a single thread.

Because of the performance benefits, always make a short-running process the default choice. Only choose a long-running process when one or more of the following is true:

- ▶ Human Interaction is required, either to perform an activity in the process or to correct an error condition.
- ▶ During the lifetime of the process, the process must be paused to wait for notification from an external activity or event.
- ▶ The business process is expected to be long in duration or it is deemed critical; for example, the business requires that the process instance not be lost should the system fail or disaster strike.

- ▶ An asynchronous request/response situation is required with an external application or service. This situation arises when the target application needs some time to process the request—it cannot respond immediately.
- ▶ The audit log is needed to obtain statistics about started processes. Only successfully completed short-running processes are audited, so you cannot use the audit log to analyze a failed activity within a short-running process.

3.1.4 Synchronous versus asynchronous behavior

The synchronous versus asynchronous invocation behavior determines whether the process returns a response to the caller in the same thread of execution (synchronous) or returns a response to the caller via a callback operation (asynchronous). By definition a synchronous process is invoked by a request/response operation, and a response is returned to the caller via this operation inside the workflow. An asynchronous process is invoked via a one-way operation, and the results and any faults are returned to the caller by the caller invoking another one-way operation or via a callback operation.

For example (see WSAD-IE documentation), you can think of a synchronous process as a telephone, and an asynchronous process as the postal system. When you are having a conversation on the phone, you send and receive messages instantaneously using the same connection. If you were to send the same message in a letter via the postal service, it is delivered in one manner, and its response returned in another.

3.1.5 When compensation is registered and when it gets triggered

Compensation reverses, or “un-does”, an operation in a process when an unrecoverable fault occurs. Each Invoke activity in a process has a primary service executes when the activity is reached. Optionally it can contain a compensation service that executes when a fault occurs in the process. The compensation service executes and reverses the primary service of the activity. You can set up more than one compensation service for a process, but if one of the compensating services fails none of the compensating services scheduled behind the failing one will execute.

When compensation is invoked, the runtime environment merges the primary activities’ input and output messages into a combined message that passes to the compensating service. The primary service and the process can fill in parts of the message as the process completes normal processing. If a fault is encountered and compensation is invoked the message will contain the data that is filled as a result of previous processing. This data can be used to carry out the compensating service. The compensation service parameters are defined on the compensation properties tab on applicable activities.

3.1.6 Resources

- ▶ IBM Redbooks
 - *WebSphere Business Integration Server Foundation V5.1 for z/OS*, SG24-6382
 - *WebSphere Business Integration Server Foundation V5.1 Handbook*, SG24-6318

<http://www.redbooks.ibm.com>
- ▶ IBM WebSphere Developer Technical Journal: *Modeling compensation in Web-Sphere Business Integration Server Foundation Process Choreographer*

http://www-128.ibm.com/developerworks/websphere/techjournal/0412_herrmann/0412_herrmann.html
- ▶ *Microflow Compensation in Business Process Choreographer*

http://www-128.ibm.com/developerworks/websphere/library/techarticles/0501_duerrstein/0501_duerrstein.html

3.2 Business Process Execution Language and Web Services

The Business Process Execution Language (BPEL) is the underlying definition of the business process and all of its rules for flow, conditions, interactions, fault handling, compensation, and other behaviors when designing business processes. The WSAD-IE tool allows you to visually build, debug, and deploy business processes that use BPEL to interface with Web Services.

Web Services are independent, isolated, reusable applications that are designed to perform a certain function within the business. These services use open-source standards for the following:

- ▶ Web Service Description Language (WSDL)
Describes the public interface to the service. The definition uses XML format.
- ▶ Discovery of the services: Universal (UDDI), (WSIL), and so on.
To find a published service that meets your needs.
- ▶ Communications: SOAP/XML over standard protocols such as HTTP, HTTPS, JMS, and so on.

The major advantages of using Web Services are the minimal exposure of the service's logic to the service requester (only the interface information need be exposed), separation of the requester and the implementation, easy reuse of the services, and platform independence.

There is extensive literature available introducing BPEL for Web Services (BPEL4WS) and the Web Services themselves. See:

- ▶ *Business processes in a Web services world*
<http://www-128.ibm.com/developerworks/webservices/library/ws-bpelwp/index.html>
- ▶ *Business Process Execution Language (BPEL) Resource Guide*
<http://www.bpelsource.com/index.html>
- ▶ Redbook *BPEL4WS Business Processes with WebSphere Business Integration: Understanding, Modeling, Migrating*, SG24-6381
<http://www.redbooks.ibm.com/abstracts/SG246381.html?open>

3.3 Versioning and binding

The Business Process Choreographer allows you to specify during what time frame the business process is valid. This section provides hints, tips, and possible consequences of using the versioning parameters.

3.3.1 Versioning

Use the following steps to set the **Valid From** information in the WSAD-IE tool:

1. Select the starting activity of the process.
2. Select the **Valid From** month, day, year, hours, minutes, and seconds in the Server properties, as shown in Figure 3-3.



Valid From:	Jan	/	12	/	2004
	12	:	00	:	00

Figure 3-3 Updating the Valid From property for the process template

Templates for the same business process may exist with different **Valid From** date specifications, but only one version of the template is valid at any given time. When an instance of the process is created, it is created using the version of the template that is currently valid. Figure 3-4 shows that we have three process templates. Supposing that today is September 18, 2005, the middle template is the valid version.

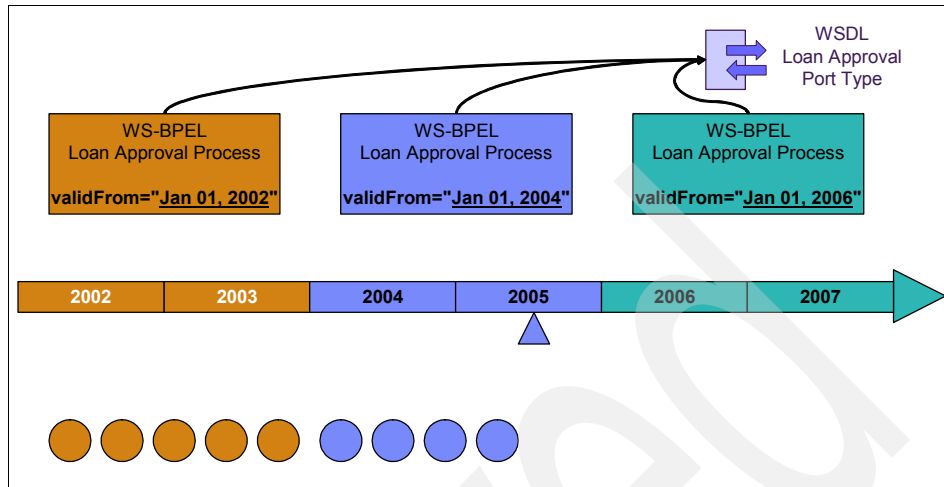


Figure 3-4 Example of multiple process templates

Process instances that were created using different template versions may coexist (illustrated at the bottom of Figure 3-4), since a given instance executes based on the version of the template that was valid at the time of its creation.

The WSAD-IE tool generates the process façade session EJB™ with a unique JNDI name (name space + process name + valid from).

3.3.2 Limitation of versioning

In order to ensure that two or more templates representing the same business process are not valid at the same time, only a “Valid From” date is supported. This prevents someone from accidentally specifying dates that overlap. If you have a requirement to specify a “Valid To” date, you can create a dummy business process that basically invalidates the previous template.

If you have an offer that is valid from January 1 through December 31 of 2005, for example, you can create the dummy process to have a “Valid From” date of January 1, 2006 00:00:00. The dummy process may contain steps to log some information or may generate a notification that the offer has expired, depending on your requirements.

3.3.3 Early versus late binding

This section explains the concepts of late binding and early binding.

Late binding

With late binding, the currently valid version of the template is determined at runtime when the process instance is created. If a new version of the template becomes valid, it is dynamically recognized without redeployment of the caller. See the top of Figure 3-5.

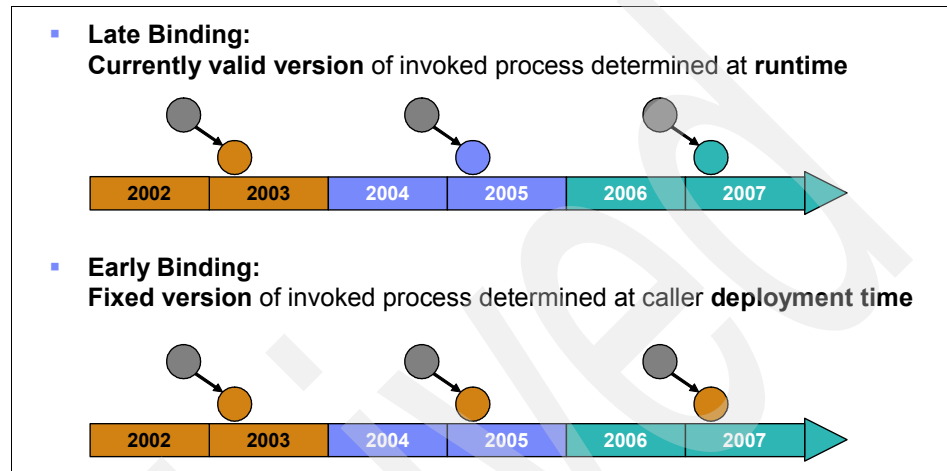


Figure 3-5 Late versus early binding sample

Early binding

With early binding, a fixed version of the template is determined at the caller's deployment time. Redeployment of the caller is required in order to pick up a new version of the template. See the bottom of Figure 3-5.

Supported Scenarios

Following are supported scenarios of late and early binding.

1. Parent process invoking a service exposed by a business process or subprocess:

BPEL partner link attribute for parent process

- Late binding with **Resolution Scope** set to Computed
- Early binding with **Resolution Scope** set to Deployment (short-running processes only)

To set the **Resolution Scope** in the WSAD-IE tool

- Select the desired Partner Link.
- Choose the scope from the pull-down list in the **Server** properties, as shown in Figure 3-6.

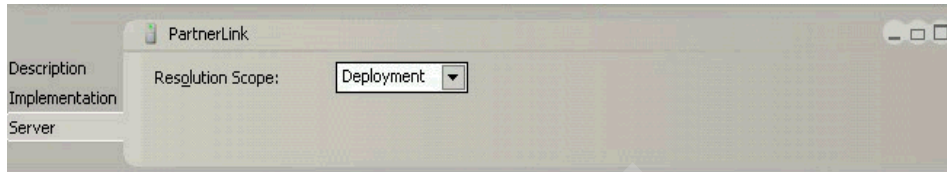


Figure 3-6 Setting the binding for a partner link in the WSAD-IE tool

Note: With respect to these late binding and instance location scenarios, the **Resolution Scope** values Deployment, Process, and Activity all have the same behavior.

For long-running processes, use different names for invoked processes to explicitly indicate different versions.

2. EJB application invoking a service exposed by a business process
 - Late binding only—current valid version is selected

The application keeps the process template ID for long-running conversations, which allows for identification of the correct “old” instance, even if meanwhile a new template becomes valid.

3. JMS application invoking a service exposed by a business process
 - Late binding only - current valid version is selected

There is only one interaction with an invoked process. It is a one-way operation, fire-and-forget, or a request-response operation.

3.4 Audit logging

The Business Process Choreographer supports automatic logging of activity that occurs during the process. Logging is used for the following:

- ▶ To produce an audit trail history of what happened, and when it happened
- ▶ To track the progress of the business process
- ▶ To coordinate work between independent business processes
- ▶ To monitor for exceptions in a business process
- ▶ To capture statistics to be used to improve the business process. To capture statistics to be used for reporting and analysis

The logging is implemented using the Common Event Infrastructure (CEI) component of the WBI Server Foundation product. The logging data is captured in an event object that is then emitted to the CEI.

In this section we discuss the efficiency of the logging process and the implications of using the logging facility. We also touch on the options that exist for viewing the emitted events.

Note: Audit logging to CEI helps you to react to process events externally. Consider using the WBISF Audit trail table for the purpose of keeping a trail of what happened, because it is more lightweight than CEI.

3.4.1 Common Event Infrastructure (CEI)

When you choose to log events during the business process, data about the event is captured in a standardized format called the Common Base Event (CBE). The CBE is an emerging standard that is an XML based format. The event includes five logical sections that contain standard elements plus extensible elements in order to allow for the following application specific data:

1. Common Base Event elements, such as creation time, priority, severity, and global instance ID.
2. Event Classification elements to describe the situation.
3. Source Component Identity elements.
4. Context data fields.
5. Application data to extend the basic format.

The WBI Server Foundation automatically fills in the details from the runtime environment such as server name and business context. The application supplies the business data. Refer to *WBISF for z/OS V5.1 Applications* for details of the attributes within each of the five sections.

The CEI takes the emitted event and stores it in a database. It can also distribute it to interested consumers for additional processing, as shown in Figure 3-7 on page 29.

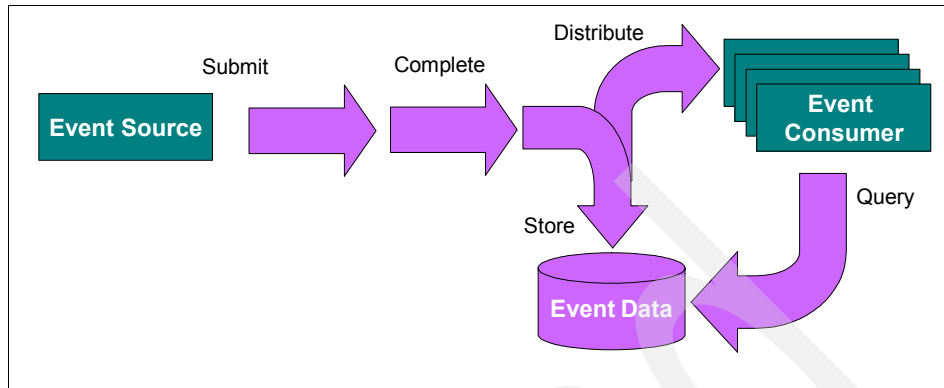


Figure 3-7 The flow of a log event emitted to the Common Event Infrastructure

In addition to the fact that the event is written to multiple tables in a database, which of course takes system resources, you need to be aware that more than one event may be written for a given activity. This depends on the number of state changes for the activity. Figure 3-8 shows some of the activities and the types of events they emit.

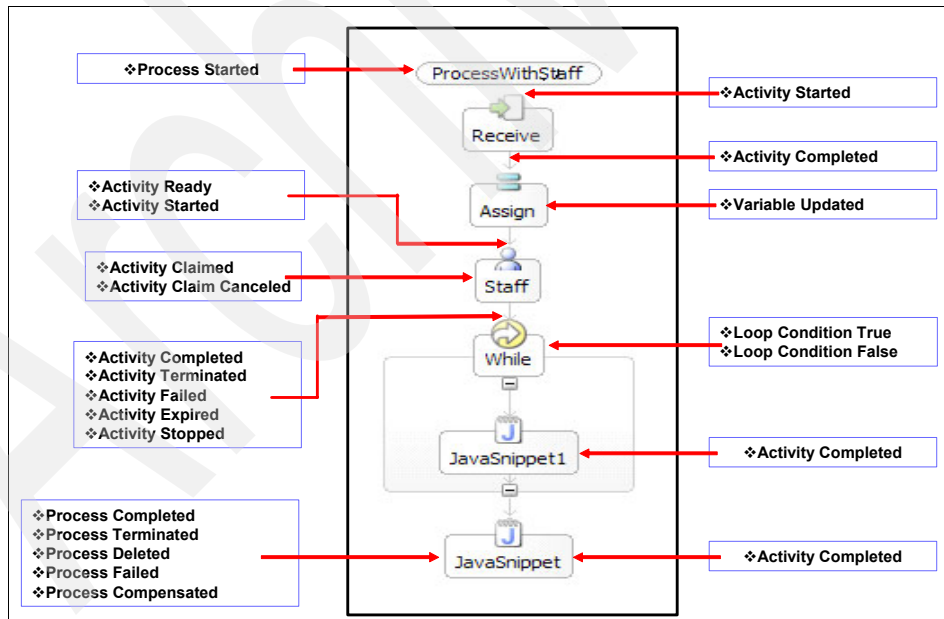


Figure 3-8 Log events emitted by some of the process activities

To set an activity to emit audit log events, in the WSAD-IE tool, check the ☒ Business Relevant box in the Server properties of the selected activity.

3.4.2 Implications of using Audit Logging

When using Audit Logging consider the following issues:

- ▶ Performance
- ▶ Debugging methods
- ▶ Roll-back in short- versus long-running processes
- ▶ How to customize a CBE to include application specific data, and the implications of carrying all of this data in the record (in other words, don't include more than you have to)
- ▶ Problems with creating own events
- ▶ Problems with overusing the audit logging in all activities
- ▶ Scalability issues
- ▶ Best practice of redirecting the events to legacy systems or data warehouses so that the CEI database can be kept as small as possible
- ▶ Emitting events from the services that the BPEL process uses

If the service is already emitting events, then maybe the process does not need to duplicate this effort. Note that the first rev of the WBI monitor only monitors events emitted by the BPEL process.

3.5 Staff resolution

If you have your own mechanisms for validating roles and varying degrees of authority and specializations, then within the Staffing model your organization's responsibility model has to be integrated with the Process Engine. This section points out considerations and sources for implementing such behavior.

Staff resolution interaction takes place with a process at the following two points:

- ▶ At process deployment, in order to create an optimized staff query.
- ▶ When a process is invoked and the process engine creates a work-list item, the scheduling of activities, and the triggering of events.

Process Choreographer supports processes with human interaction, staff, only when security is enabled in WebSphere for z/OS. This allows client authentication to occur and work items to be placed on a specific users work list.

For more information, consult the following Web site:

<http://www-128.ibm.com/developerworks/websphere/zones/was/wpc.html>

We only provide a little overview here.

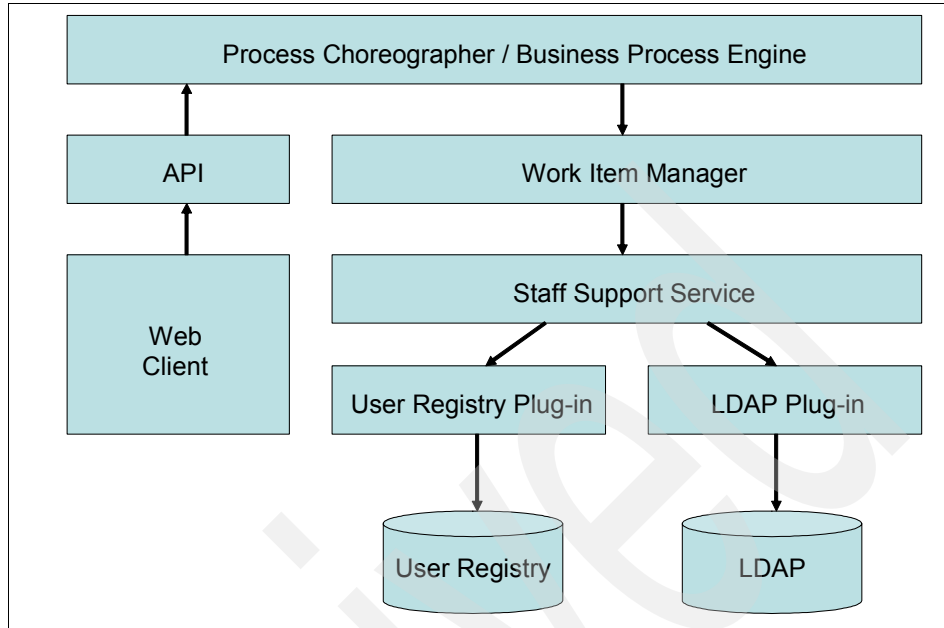


Figure 3-9 Staff resolution - main components

The main components involved in Staff resolution are depicted in Figure 3-9:

- ▶ **Web Client** - this is a .jsp-based application that provides users with a set of work list items on which they can perform queries, view details, and perform certain actions. The Web Client interacts with the process engine via the Process Choreographer APIs.
- ▶ **Process Engine** - this the main component of Process Choreographer. For each process the Process Engine performs the following:
 - Manages states
 - Manages the activities of the process and the navigation
 - Allows process data to be stored transparently or in memory
 - Manages plug-ins associated with a process and invokes them when needed by the process
- ▶ **Work Item Manager** - this is responsible for the following:
 - Creating and deleting work items
 - Resolving work item queries for process participants
 - Invoking staff query resolution. This is invoked by WIM but is performed by the Staff Support Service and the Staff resolution plug-ins
 - Enforces authorization for work items created for users

WIM uses an internal cache to resolve staff queries. The query results in the cache having a user defined expiration time. This allows for better performance and allows users to control the cache refresh based on their environment.

- **Staff Support Service** - this manages the life cycle of the Staff resolution plug-in. It translates the staff query verbs into plug-in specific queries and manages the invocation of the Staff resolution plug-ins based on the requests. The “staff query verbs” are abstract staff queries that define the staff members who will have access to a work list item. At deployment time these verbs are translated into staff resolution specific queries. See Figure 3-10 for more detail.

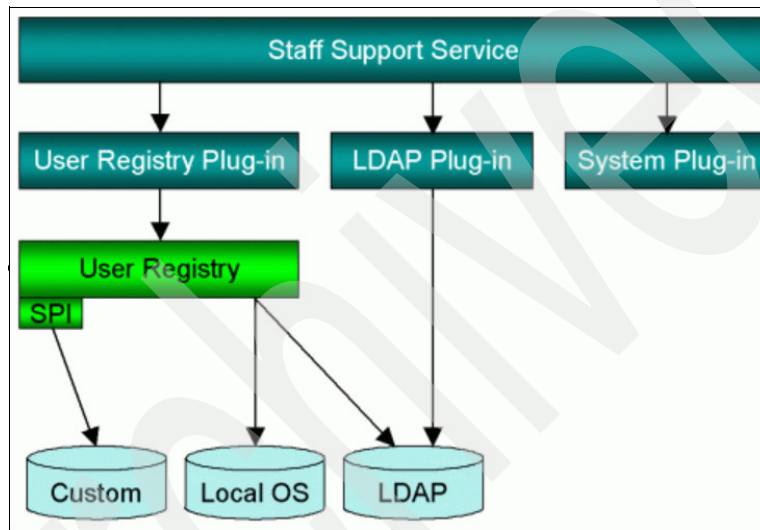


Figure 3-10 Staff Support Service - components

- **Staff resolution plug-ins** - these are responsible for the following:
 - Retrieving staff information from the repositories they are bound to, such as User Registry and LDAP
 - Deploying and executing a staff query
 - Processing queries by invoking a set of APIs specific to the repository being used
- **Staff Repository** - this is a repository that sits behind the User Registry. It is not a part of Process Choreographer and does not need to be changed for it. Process Choreographer gains access to the information in the repository by invoking a Staff resolution plug-in (read only mode) which formats data from repository format into a format that Process Choreographer can use.

Summary

The more features some tools have, the harder it can be to understand and utilize them effectively. WebSphere Business Integration Service Foundation (WBISF) is a powerful tool, and its robust nature can seem daunting.

WBISF can also seem intimidating to activate, with a tendency for errors to a certain extent. The topics we touched in earlier chapters on provide a working guideline for a successful realization of a working WebSphere in a secure Deployment Manager server configuration with Process Choreography centered around Service Orientated Architecture (SOA).

The details we offer in this Redpaper are not the only possibilities for using this tool. The options we explored here only represent working constructs that will be successful. The features we did not address were mostly omitted due to time considerations. Subsequent revisions of the product will broaden the power and flexibility of the services that WBISF has brought into focus.

Index

A

- activity 27
 - approval 17
 - asynchronous JMS 20–21
 - automated 17
 - data collection 17
 - event 29
 - human interaction 17
 - invoke 21
 - manual intervention 17
 - participates 18
 - pause 17
 - sequence 19
 - start 20
 - state 18, 20
 - synchronous 21
- alert 16–17
- analysis 27
- APAR 3–4, 9
- APF 10
- API 32
- application
 - data 28
 - specific data 30
- APPLY 4
- approval activity 17
- ASCII 11
- asynchronous
 - behavior 22
 - JMS activity 20–21
 - request 22
- audit 4
 - log 22
 - log event 29
 - logging 15, 27, 30, 42
 - trail history 27
- authority 30
- authorization 9, 31
- automated activity 16–17
- automated tasks 20
- automatic logging 27

B

- BBOWSTRT 7
- bind packages 6
- binding 24
- bottlenecks 12
- BPE installation 7
- bpeconfig.jacl 7, 9–11
- BPEL 1, 23, 30
 - partner link attribute 26
- BPEL4WS 1, 24
- bpelib.jacl script 9
- BPEWebClient 9
- BPX.SUPEUSER 8
- BPXPRMxx member 13
- build number 3
- business context 28
- business data 28
- business process concepts 15, 42
- Business Process Execution Language 23

C

- callback 22
- CBE 28, 30
- CEI 6, 8, 27–28, 30
 - connection factories 11
 - issues 11
- CLASSPATH 6, 10
- client authentication 30
- Cloudscape 10
- commit after 18
- commit before 18
- Common Base Event 28
- Common Event Infrastructure 27–28
- compensating service 22
- compensation 16, 19–23
 - properties 22
- computed 26
- conditions 23
- configuration 4
- console dump 12
- Contact us ix
- context data 28
- coordinate work 27

- correct error 21
- CPU Time 11
- creation time 28
- critical process 21
- customization panels 5
- cylinder page dataset 13

D

- data collection activity 17
- database 6, 28
- DB2 6
 - driver 6
 - load libraries 10
 - object 9
 - subsystem 9
 - V7 6
 - V8 6
 - variables 10
- db2jcc.jar 6
- DDDEF 4
- debugging method 30
- deployment 26–27
 - manager 6–8, 10
- different versions 27
- directory path 4
- disaster 21
- dummy business process 25
- dummy service 20
- duration 21
- dynamic data definitions 4

E

- early binding 25–26
- EJB application 27
- e-mail 17, 21
- Embedded Messaging 10
- emit
 - audit log event 29
 - event 28, 30
- error situation 17
- event 21
 - classification 28
 - data 28
 - emit 30
 - object 27
 - own 30
 - redirecting 30
 - triggering 16–17, 20, 30

- exceptions 27
- expiration 21
 - time 32
- extensible elements 28
- external activity 21
- external application 22
- external event 17, 20
- external service 21–22

F

- failed activity 22
- failure 19, 21
- fault 22
 - handling 23
- federation 7
- fire-and-forget 21, 27
- fixed template 26
- flow 23
- formatting data 32

G

- global instance id 28
- Global Security 7

H

- hanging process 21
- HEAP parameter 8
- HFS 4
 - sharing 8
- High Level Qualifier 4
- HLQ 4
- HTTP 23
- HTTPS 23
- human interaction 16–17, 20–21, 30
 - activity 17
- human intervention 16

I

- ID 9
- IJP 7
- indoubt tables 6
- InfoCenter 4
- install 4
- instance location 27
- interactions 23
- internal cache 32
- interruptible 16

invocation behavior 22
invoke activity 21

J

jcc/classes 6
JDBC 6
 driver variables 10
 instruction 6
 provider 6
 trace 6
JMS 23
 activity 20
 application 27
 provider 7
 sample programs 7
JNDI name 25
job output 4
JVM 13

L

late binding 25–27
LDAP 32
life cycle 32
link list 10
location 6
logging 27
long-running process 16, 21, 27, 30

M

macro flow 16
manual intervention 17
max memory 11
MAXASSIZE 13
MAXCORESIZE 13
MAXFILEPROC 13
MAXPROCSYS 13
MAXPROCUSER 13
MAXQUEUEDSIGS 13
MAXSHAREPAGES 13
MAXTHREADS 13
MaxThreads 11
MAXTHREADTASKS 13
memory 21, 31
messaging connections pools 11
Messaging throughput 12
micro flows 20
migration 4

mount point 4

N

naming conventions 5
navigation 31
ND cell 7
ND configuration 5
Network Deployment 1, 5
non-interruptible 20
non-responses 17
notification 21

O

OMVS 9
Operating System Tuning 13
own events 30

P

PAGEADD 13
PAGESPACE 13
parallel execution 20
parallel paths 18, 20
parent process 26
PARMLIB 13
Participates activity 18
partner link 26
pause activity 17
performance 21, 32
permission bits 8
persistence 21
plug-ins 31
port 5
 example 5
 spreadsheets 5
primary service 22
priority 28
process 27
 binding 21
 data 31
 deployment 30
 event 28
 façade session EJB 25
 hanging 21
 instance 26
 instance data 21
 long-running 21, 30
 participants 31

- pause 17
- properties 20
- short-running 20–21, 30
- statistics 22
- template 24
- Process Choreographer APIs 31
- Process Choreographer database 10
- Process Engine 31
- product number 3
- Program Directory 4
- program number 3
- progress track 27
- PTF 3–4

Q

- queries 31

R

- RACF ID 8
- random order 20
- read access 8
- read only 32
- read-only
 - file system 8
- Redbooks Web site ix
- redeployment 26
- redirecting events 30
- reporting 27
- repository 32
 - format 32
- request-response operation 27
- requires own 18
- resolution scope 26–27
- responsibility model 30
- reversal 19
- rlimit 11
- roll back 21, 30

S

- scheduling 30
- Schema 9
- security 7, 30
- server
 - name 28
 - properties 24, 26, 29
- setRollbackOnly 21
- severity 28

- short-running process 20–22, 30
- Signal 29 11
- single thread 20–21
- SMP/E 4
- SOAP connection 7
- SOAP/XML 23
- source component ID 28
- specializations 30
- SQL errors 8–9
- sqlid function 8
- staff 30
 - activity 16
 - information 32
 - member 32
 - query 30, 32
 - resolution 31
 - verbs 32
 - resolution 30
 - plug-in 31–32
- Staff Repository 32
- Staff Support Service 31–32
- staffing model 30
- standard elements 28
- start activity 20
- starting activity 24
- state 20, 31
 - changes 29
 - data 21
 - information 18
- stateless 20
- statistics 27
- steplib 10
- storage group 6
- subsystem 6
- super user 9
- SUPERUSER 13
- symbolic link 4, 10
- synchronous
 - activity 21
 - behavior 22
- SYSADM 6, 8
 - authority 6

T

- table spaces 6
- telnet session 9
- template 24, 26
- threading 12

time limit 17, 21
Trade3 13
transaction 18, 20
 roll back 21
transactional behavior 18, 20
tuning
 OS 13
two-phase commit 19
types of events 29

U

UDDI 23
undo 19, 22
unfederate AppServer 8
Universal 23
unrecoverable fault 22
User Registry 32
UserId 8

V

valid from 24
valid to 25
validating roles 30
versioning 24
 limits 25

W

WBISF 1
Web Client 31
Web Service 23
 Description Language 23
WebSphere MQ 7, 10, 12
 embedded 7
WIM 31
work item
 create and delete 31
 manager 31
 queries 31
workaround 21
worker threads 12
worklist item 30–31
worksheets 5
WSAD-IE 1, 16, 18, 20–22, 26, 29
WSDL 23
WSIL 23

X

XML 23, 28

Z

z/AAP 12



Redpaper

Best Practices for WebSphere Business Integration Server Foundation on z/OS

Setting up your Process Choreographer environment

This Redpaper will help systems programmers and WebSphere administrators exploit WebSphere Business Integration Server Foundation (WBISF) functions.

Creating and deploying BPEL-based processes

It covers the functional implementation on z/OS and includes information about how to install and configure WBISF for z/OS, V5.1 for high performance applications.

It contains guidelines and best practices to set up the Process Choreographer environment. It points out specific issues with SMP/E, ports and naming conventions, CEI and DB2, WebSphere MQ, security, BPE, and tuning, and provides recommendations how to tackle them.

This paper also introduces business process concepts and a number of ways to construct them in a comprehensive and efficient manner. The discussion includes the business process itself, compensation, audit logging using the Common Event Infrastructure, versioning, the underlying process language, and Web Service invocation.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks