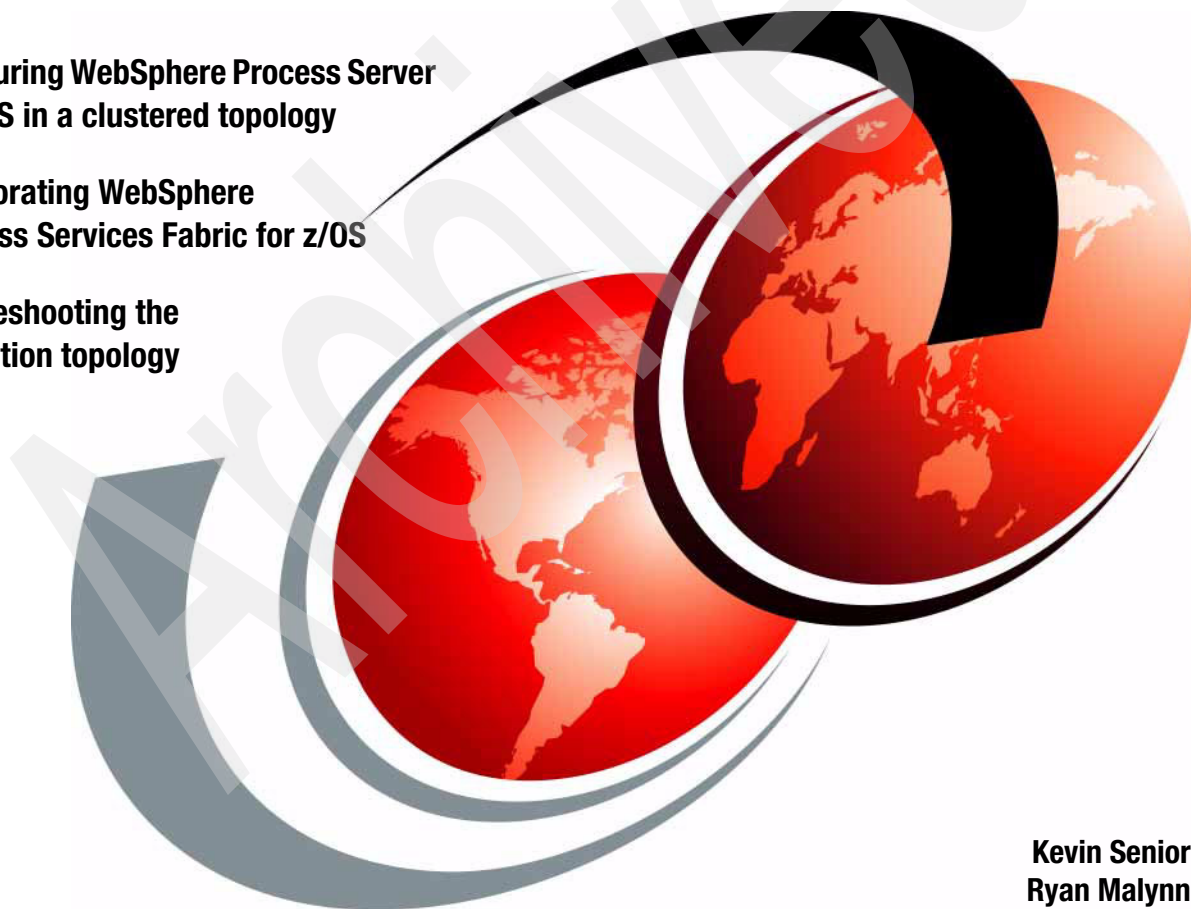


# z/OS: WebSphere Business Process Management V6.1.2 Production Topologies

Configuring WebSphere Process Server for z/OS in a clustered topology

Incorporating WebSphere Business Services Fabric for z/OS

Troubleshooting the production topology



Kevin Senior  
Ryan Malynn





International Technical Support Organization

**z/OS: WebSphere Business Process Management  
V6.1.2 Production Topologies**

January 2009

Archived

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

**First Edition (January 2009)**

This edition applies to Version 6, Release 1.2 of WebSphere Process Server for z/OS.

© Copyright International Business Machines Corporation 2009. 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

<b>Notices</b> .....	.xi
Trademarks .....	xii
 <b>Preface</b> .....	 xiii
The team that wrote this book .....	xiv
Become a published author .....	xv
Comments welcome .....	xvi
 <b>Chapter 1. Business Process Management production topologies for z/OS</b> .....	 1
1.1 Introduction .....	2
1.1.1 WebSphere Application Server for z/OS architecture .....	3
1.1.2 Hardware considerations .....	4
1.1.3 HTTP servers, LDAP servers, and external security managers .....	4
1.1.4 WebSphere Process Server for z/OS scalability and availability .....	5
1.1.5 Summary .....	5
1.2 WebSphere Process Server components .....	6
1.2.1 Databases .....	6
1.2.2 Service integration buses .....	7
1.2.3 Business Process Choreographer (BPC) .....	8
1.2.4 WebSphere Process Server applications .....	8
1.2.5 Common Event Infrastructure (CEI) .....	9
1.3 WebSphere Process Server deployment environment patterns .....	9
1.3.1 Single-cluster topology pattern .....	10
1.3.2 Two-cluster topology pattern .....	13
1.3.3 WebSphere MQ for z/OS queue sharing .....	16
1.3.4 Custom topology patterns .....	17
1.4 Selecting an appropriate topology .....	17
1.4.1 The single-cluster topology pattern .....	18
1.4.2 The two-clusters topology pattern .....	19
1.4.3 Custom topology .....	20
1.4.4 Condensed topology selection criteria .....	21
1.5 Incorporating Fabric into the two-clusters topology .....	22
1.5.1 WebSphere Business Services Fabric .....	22
 <b>Chapter 2. Configuring WebSphere Process Server for z/OS Network Deployment</b> .....	 25
2.1 Overview .....	26
2.2 Plan the Network Deployment configuration .....	29

2.2.1	Hardware and software pre-requisites	29
2.2.2	Installing WebSphere Process Server for z/OS using SMP/E	30
2.2.3	Choosing the topology	32
2.2.4	Planning dataset names and disk space	32
2.2.5	Planning using a spreadsheet	33
2.2.6	Planning security	38
2.2.7	Managing EJB roles using RACF GEJBROLE profiles	41
2.2.8	Using intermediate symbolic links and ICFCAT aliases	45
2.2.9	Planning DB2 names	46
2.2.10	Choosing currentSchema or currentSQLID	48
2.2.11	Implications for GRANT statements in DB2	54
2.2.12	Plan and create DB2 buffer pools	55
2.2.13	Decide how you will create and execute the DDL	56
2.2.14	WebSphere Application Server for z/OS 64-bit mode	59
2.3	Prepare a WebSphere cell, RACF, and DB2	59
2.3.1	Create intermediate symbolic links to products	60
2.3.2	Create ICFCAT aliases to SBBO* libraries	61
2.3.3	Create RACF user IDs, groups, and EJBROLE profiles	62
2.3.4	Create a DB2JccConfiguration.properties file	65
2.3.5	Customize a .profile	68
2.3.6	Verify Java and the DB2 Universal JDBC Driver	69
2.4	Create the deployment manager and an empty node	70
2.4.1	Task overview	71
2.5	Add WebSphere Process Server to the DMGR	72
2.5.1	Back up the deployment manager's ZFS data set	73
2.5.2	Run the zSMPInstall script	73
2.5.3	Prepare DmgrDB2.rsp	76
2.5.4	Back up the deployment manager's ZFS	77
2.5.5	Run zWPSConfig.sh	77
2.5.6	Start the deployment manager	78
2.5.7	Set the cell's JVMs to run on local time	78
2.6	Configure DB2 z/OS using createDB.sh	79
2.6.1	Copy createDB.sh to a work directory	80
2.6.2	Customize the sample createDB.sh script	81
2.6.3	Execute createDB.sh to generate the DDL	83
2.6.4	Modify the generated DDL	84
2.6.5	Create the databases	86
2.6.6	Execute the DDL using createDB.sh	86
2.6.7	Execute the DDL using DSNTEP2	87
2.6.8	Grant table privileges to the J2C auth alias user ID	88
2.7	Adding WebSphere Process Server to the empty nodes	88
2.7.1	Empty node: Run the zSMPInstall script	89
2.7.2	Empty node: Prepare ManagedDB2.rsp	90

2.7.3	Back up the WebSphere configuration HFS datasets . . . . .	92
2.7.4	Empty node: Run zWPSCfg.sh . . . . .	92
2.7.5	Set the currentSchema property on WBI_DataSource . . . . .	93
2.7.6	Clear the OSGi cache . . . . .	94
2.7.7	Back up the WebSphere configuration HFS datasets . . . . .	94
2.7.8	Federate the primary empty node into the cell . . . . .	95
2.7.9	Update the primary node's Node Agent ports . . . . .	95
2.7.10	Create a personal certificate for the servant . . . . .	98
2.7.11	Back up the WebSphere configuration HFS datasets . . . . .	98
2.8	Configure the common database in DB2 for z/OS . . . . .	98
2.8.1	Create the common database . . . . .	99
2.8.2	Copy the common DDL to z/OS sequential files . . . . .	99
2.8.3	Edit and execute the common DDL . . . . .	100
2.8.4	Grant table privileges to the J2C auth alias user ID . . . . .	100
2.9	Create a secondary node . . . . .	101
2.9.1	Empty node: Run the zSMPInstall script . . . . .	102
2.9.2	Empty node: Prepare ManagedDB2.rsp . . . . .	102
2.9.3	Back up the WebSphere configuration HFS datasets . . . . .	104
2.9.4	Empty node: Run zWPSCfg.sh . . . . .	104
2.9.5	Back up the WebSphere configuration HFS datasets . . . . .	105
2.9.6	Add DB2 libraries to the Servant and Adjunct JCL . . . . .	106
2.9.7	Federate the secondary empty node into the cell . . . . .	106
2.9.8	Update the secondary node's Node Agent ports . . . . .	107
2.9.9	Back up the WebSphere configuration HFS datasets . . . . .	107
2.10	Create WebSphere Process Server clusters . . . . .	108
2.10.1	Creating a cluster . . . . .	109
2.10.2	Define additional clusters . . . . .	115
2.10.3	Familiarize yourself with the cluster . . . . .	115
2.10.4	Update the server's ports . . . . .	117
2.10.5	Back up the WebSphere configuration HFS datasets . . . . .	119
2.11	Define HTTP servers or proxies . . . . .	120
2.11.1	IBM HTTP Server for z/OS powered by Apache . . . . .	120
2.11.2	IBM HTTP Server for z/OS (LGW) . . . . .	121
2.12	Service Component Architecture . . . . .	121
2.12.1	Generate DDL for all SIBs using sibDDLGenerator.sh . . . . .	122
2.12.2	Edit and execute the SIB DDL . . . . .	124
2.12.3	Run the SCA wizard . . . . .	125
2.12.4	Verify SCA Modules and SCA runtime . . . . .	128
2.12.5	Back up the WebSphere configuration HFS datasets . . . . .	128
2.13	Remote SCA in AppTarget cluster . . . . .	128
2.14	Configuring CEI using the Integrated Solutions Console . . . . .	129
2.14.1	Run the CEI configuration wizard . . . . .	131
2.14.2	Update the CEI bus authentication alias . . . . .	135

2.14.3	Update the CEI bus connector role . . . . .	136
2.14.4	Check the database name on the CEI data sources. . . . .	138
2.14.5	Review the CEI DDL . . . . .	139
2.14.6	Edit and execute the CEI Event database DDL . . . . .	142
2.14.7	Grant table privileges on CEI tables . . . . .	143
2.14.8	Define CEI roles to RACF . . . . .	143
2.14.9	Restart the cluster. . . . .	144
2.14.10	Verify the CEI configuration and CBE Browser. . . . .	146
2.14.11	Back up the configuration ZFS datasets . . . . .	146
2.15	Business Process and Human Task containers . . . . .	146
2.15.1	Execute the BPC configuration wizard . . . . .	147
2.15.2	Set the schema name for the BPC message engine . . . . .	152
2.15.3	Set currentSchema on the BPC datasource . . . . .	153
2.15.4	Verify that the applications are present. . . . .	154
2.15.5	Restart the cluster hosting BPC . . . . .	154
2.15.6	Back up the WebSphere configuration HFS datasets. . . . .	154
2.16	BPC Explorer. . . . .	155
2.16.1	Install BPC Explorer using the Integrated Solutions Console . . . . .	155
2.17	Configuring BPC Event Collector and BPC Observer . . . . .	159
2.17.1	Install Event Collector using the Integrated Solutions Console. . . . .	160
2.17.2	Installing Event Collector and BPC Observer using scripts . . . . .	164
2.17.3	Install BPC Observer using the Integrated Solutions Console . . . . .	164
2.18	Verify BPC. . . . .	167
2.18.1	Install the bpcivt application . . . . .	168
2.18.2	Enable CSiv2 Asserted Identity . . . . .	171
2.18.3	Run bpcivt and ZOSBookOrderApp . . . . .	175
2.19	Verify BPC Event Collector and BPC Observer . . . . .	176
2.19.1	Change Event Collector thresholds. . . . .	177
2.19.2	Enable CEI Event Logging . . . . .	179
2.19.3	Start WBILocationMonitor.CEI traces . . . . .	180
2.19.4	Stop and start the BPCECollector application. . . . .	182
2.19.5	Generate some events using a sample application . . . . .	182
2.19.6	Run the BPC Observer in a browser. . . . .	183
2.19.7	Back up the WebSphere configuration HFS datasets. . . . .	200
2.20	Configuring Business Space powered by WebSphere . . . . .	200
2.20.1	Create a data source for Business Space. . . . .	201
2.20.2	Enable Widgets in Business Space . . . . .	202
2.20.3	Run the Business Space Configuration wizard . . . . .	203
2.20.4	Prepare the Business Space DDL . . . . .	205
2.20.5	Execute the Business Space DDL . . . . .	205
2.20.6	Define EJBROLE profile for role Administrator . . . . .	206
2.20.7	Start the applications. . . . .	207
2.20.8	Enable Business Rules in the Business Space. . . . .	207



2.20.9	Configure local Business Space help	209
2.20.10	Verify Business Space	209
2.21	People Directory Provider	216
2.21.1	Customize an LDAP transformation xsl file	217
2.21.2	Create an new LDAP People Directory Provider configuration	219
2.21.3	Add users and groups to the LDAP server	222
2.21.4	Deploy the application	222
2.21.5	Test the application	223
2.22	Failed events	223
2.22.1	What is an event?	223
2.22.2	How to use the Failed Event Manager	224

<b>Chapter 3.</b>	<b>Incorporating WebSphere Business Services Fabric into a production topology on z/OS</b>	<b>227</b>
3.1	Introduction	228
3.2	Planning and preparing the Fabric installation	230
3.2.1	Software pre-requisites	230
3.2.2	Installing WebSphere Business Services Fabric in SMP/E	230
3.2.3	Sample JCL and shell scripts used in this chapter	230
3.2.4	Creating the deployment directory	231
3.3	Installing Fabric in a clustered environment	231
3.3.1	Copy the Fabric artifacts	231
3.3.2	Create required RACF user ID, groups, and profiles	234
3.3.3	Create the Fabric databases	239
3.4	Add Fabric resources to WebSphere Process Server for z/OS	245
3.4.1	Set WebSphere environment variables	246
3.4.2	Create an authentication alias for Fabric	247
3.4.3	Create and configure the data sources	248
3.4.4	Create and configure the service integration bus	252
3.4.5	Create destinations in the service integration bus	256
3.4.6	Configure the JMS provider	259
3.4.7	Configure a mail provider	266
3.4.8	Configure security	267
3.4.9	Configure distributed cache	271
3.4.10	Configure a namespace variable for CEI	274
3.4.11	Install the Fabric core EAR files	275
3.4.12	Grant access to the Fabric Tools Console	277
3.5	Verify the Fabric installation and configuration	278
3.6	Install and test the sample application	282
3.6.1	Update the endpoint URLs of ITSOPBankCBAPrj	283
3.6.2	Change endpoint URLs by editing the .owl file	283
3.6.3	Import the FCA files	284
3.6.4	Assign MDFABADM to the ITSOPBankOrg team	284

3.6.5	Create an External LDAP federation project . . . . .	286
3.6.6	Set Enrollments . . . . .	291
3.6.7	Install the sample application's EAR Files . . . . .	295
3.6.8	Map modules to the Web server . . . . .	298
3.6.9	Update the SCA Import URLs . . . . .	298
3.6.10	Test the sample application . . . . .	300
3.7	Troubleshooting the WebSphere Business Services Fabric for z/OS installation. . . . .	303
3.8	WebSphere Business Services Fabric events . . . . .	303
<b>Chapter 4.</b>	<b>Troubleshooting WebSphere Process Server for z/OS . . . . .</b>	<b>305</b>
4.1	Sources of information about an error. . . . .	306
4.2	WebSphere Process Server errors . . . . .	306
4.2.1	zSMPEInstall.sh failed RC=12 . . . . .	307
4.2.2	Failure in loading T2 native library db2jcc2zos . . . . .	308
4.2.3	-922 SQLCODEN922 connecting to DB2 . . . . .	308
4.2.4	-551 SQLCODEN551 connecting to DB2 . . . . .	309
4.2.5	Basic authentication failed for user . . . . .	309
4.2.6	SCHD0128E: The daemon for scheduler ... could not be started. . . . .	310
4.2.7	Resources not seen in the Administration console . . . . .	311
4.2.8	Events not seen in the CBE Browser during verification. . . . .	311
4.2.9	Could not start TM_PreddefinedTasks_V612_mdcl01_WPS . . . . .	312
4.2.10	WSVR0220I: Application stopped: TaskContainer_mdcl01_WPS. . . . .	313
4.2.11	BBOO0220E: J2CA0081E: Method cleanup failed. . . . .	313
4.2.12	-551 MDDBU;SELECT;SYSIBM.SYSTABLES . . . . .	314
4.2.13	SECJ4062W: Cannot find the credential information . . . . .	315
4.2.14	Errors starting Event Collector and Observer . . . . .	316
4.2.15	-812 SQLCODEN812 when starting BPC Observer . . . . .	317
4.2.16	Product registration failed for ProcessServerZOS . . . . .	318
4.2.17	CWSIJ0063E: A network connection to host name x.x.x.x, port yyyy cannot be established . . . . .	319
4.3	Errors when launching bpcivt . . . . .	320
4.3.1	Error 403: AuthorizationFailed . . . . .	320
4.3.2	NMSV0610I: NamingException, CORBA.NO_PERMISSION, The WSCredential does not contain a forwardable token ... enable Identity Assertion. . . . .	321
4.3.3	CWSIT0086E: Bus SCA.SYSTEM.mpcell.Bus not found . . . . .	322
4.3.4	InstanceNotFoundException running eventbucket.sh. . . . .	322
4.3.5	Various issues purging the CEI event bucket . . . . .	323
4.4	Errors relating to BPC Event Collector and Observer . . . . .	325
4.4.1	CEIEM0025E, CEIES0010E, -253 SQLCODEN253 with DuplicateGlobalInstanceIdException . . . . .	326

4.4.2 No events displayed in BPC Observer; no errors in logs . . . . .	327
4.5 Errors relating to Business Space . . . . .	328
4.5.1 Error configuring Business Space . . . . .	328
4.5.2 -204 SQLCODEN204 starting Business Space applications . . . . .	330
4.5.3 Webgroup not found when using Business Space . . . . .	331
4.5.4 JSPG0235E: The JSP container failed to load the TagExtraInfo . . . . .	332
4.6 Errors related to People Directory Provider . . . . .	333
4.6.1 WSVR0017E and CWWBS0006E failing to bind the JNDI name of the people directory (staff plug-in configuration). . . . .	333
4.7 Errors related to WebSphere Business Services Fabric for z/OS . . . . .	334
4.7.1 Database missing w_dbversion table and -922 SQLCODEN922 . . . . .	334
4.7.2 Unable to locate MessageSource with name 'messageSource' . . . . .	335
4.7.3 Unable to find resource 'VM_global_library.vm' . . . . .	336
4.7.4 Error opening zip file ... soapfabric-2.3.1.jar . . . . .	336
4.7.5 NoClassDefFoundError: com.webify.wsf.spm.mdb.PerformanceMDBBean . . . . .	337
4.7.6 SAXPARSEEXCEPTION during Fabric startup . . . . .	338
4.7.7 CWSIV0958E: SINotAuthorizedException for DA.Event.Topic . . . . .	339
4.7.8 -803 SQLCODEN803 when starting fabric applications . . . . .	340
4.7.9 Garbage on Fabric login window at first logon . . . . .	341
4.7.10 com.ibm.ws.fabric.ocp.ContentImportException . . . . .	342
4.7.11 fabric-da-scdl.xsd doesn't refer to a valid schema . . . . .	342
4.7.12 CWWBF0029E: Process components of ITSOApp cannot be configured in WebSphere configuration repository . . . . .	343
4.7.13 CWWBU0001E in BPC Explorer withTestLoanProcess . . . . .	345
4.7.14 NMSV0610I: A NamingException for wbsf-cbe-emitter-factory . . . . .	349
4.7.15 CEIEM0025E The emitter failed to send the events . . . . .	349
<b>Appendix A. Additional material . . . . .</b>	<b>351</b>
Locating the Web material . . . . .	351
Using the Web material . . . . .	352
How to use the Web material . . . . .	352
Installing the sample JCL, DDL, and shells on a z/OS system . . . . .	355
<b>Abbreviations and acronyms . . . . .</b>	<b>357</b>
<b>Related publications . . . . .</b>	<b>359</b>
IBM Redbooks . . . . .	359
Other publications . . . . .	359
How to get Redbooks . . . . .	360
Help from IBM . . . . .	360

Archived

# 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 illustrate 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.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

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

CICS®  
DB2 Connect™  
DB2 Universal Database™  
DB2®  
Domino®  
IBM®

Lotus®  
RACF®  
Rational®  
Redbooks®  
Redbooks (logo) ®  
SystemPac®

Tivoli®  
WebSphere®  
z/OS®  
zSeries®

The following terms are trademarks of other companies:

Snapshot, and the NetApp logo are trademarks or registered trademarks of NetApp, Inc. in the U.S. and other countries.

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

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

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

# Preface

This IBM® Redbooks® publication describes how to choose and configure a production topology for WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS. This Redbooks publication will be useful to infrastructure architects and systems programmers. Because WebSphere Process Server for z/OS interfaces with DB2® for z/OS® and with a security manager such as RACF®, parts of this Redbooks publication are also relevant to security administrators and to database administrators,

A companion Redbooks publication, *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665, covers the same topic for non-z/OS platforms. SG24-7665 is divided into the following three parts:

- ▶ Part 1. Overview and concepts
- ▶ Part 2. Building production topologies for WebSphere® Process Server
- ▶ Part 3. Extending production topologies

Part 1 of SG24-7665 includes several chapters that describe function, architecture, and security, making it of interest to z/OS customers. It should be read in conjunction with this Redbooks publication. You may think of this Redbooks publication as Part 4 of SG24-7665.

The chapters in Part 1 of SG24-7665 of particular relevance to a z/OS customer are as follows:

- ▶ Chapter 1, “Basic concepts and Business Process Management (BPM) product descriptions”
- ▶ Chapter 2, “Security considerations for BPM”

SG24-7665 discusses security for non-z/OS platforms, but there is useful general information about WebSphere and J2EE™ security that applies to WebSphere Application Server and WebSphere Process Server on any platform. If you are configuring WebSphere Process Server for z/OS but intend to use an LDAP user registry for your cell, you will want to read this chapter because it discusses LDAP considerations in detail.

- ▶ Chapter 4, “Business scenario used in this book”

This Redbooks publication makes use of a sample application that was developed for Redbooks publication SG24-7665, which we deployed into WebSphere Business Services Fabric for z/OS. We do not describe the business scenario or the application development process in detail in this Redbooks publication. Consult SG24-7665 in order to understand this scenario.

Although Part 1 of SG24-7665 contains a lot of information that applies equally to WebSphere Process Server for z/OS, Part 2 and Part 3 of SG24-7665 discuss non-z/OS topologies and many of the considerations that influence the choice of topology on non-z/OS platforms do not apply to WebSphere Process Server for z/OS.

This Redbooks publication is organized as follows:

- ▶ Chapter 1, “Business Process Management production topologies for z/OS” on page 1 contains a discussion of the factors that influence a choice of production topology on z/OS.
- ▶ Chapter 2, “Configuring WebSphere Process Server for z/OS Network Deployment” on page 25 contains step-by-step instructions for configuring both a single-cluster topology and a two-cluster topology.
- ▶ Chapter 3, “Incorporating WebSphere Business Services Fabric into a production topology on z/OS” on page 227 explains how to add Fabric to your production topology and how to verify the installation.
- ▶ Chapter 4, “Troubleshooting WebSphere Process Server for z/OS” on page 305 documents the symptoms of any problem we encountered together with its solution.
- ▶ Appendix A, “Additional material” on page 351 contains the spreadsheet that we used to plan our cell’s topology, all the sample job control language (JCL) we created, and also the shell scripts we created to simplify some of the manual steps.

We hope this Redbooks publication, together with the additional material, will be useful when configuring your production WebSphere Process Server for z/OS topology.

## The team that wrote this book

This Redbooks publication was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

**Kevin Senior** is an IBM certified IT Specialist working for the Worldwide Technology Practice within IBM Software Services for WebSphere and based out of the IBM Hursley laboratory in the UK. He has 28 years’ experience at IBM as a systems programmer working with IMS, DB2, CICS®, and WebSphere products on z/OS. Currently he specializes in WebSphere Portal Server and WebSphere Process Server for z/OS. For the ITSO, Kevin has coauthored several Redbooks



publications and IBM Redpaper publications. Although Kevin is English, he lives in Italy and works throughout Europe.

**Ryan Malynn** is an Information Developer in the United States. He has 9 years of experience in the software field and has worked at IBM for almost 8 years. His areas of expertise include content management and security. He has written extensively on WebSphere Application Server and WebSphere Portal Server.

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

Martin Keen  
IBM ITSO, Raleigh, USA

Naveen Balani  
IBM Mumbai, MH India

Rich Conway  
International Technical Support Organization, Raleigh Center, USA

Russ Heald  
IBM Hursley, UK

Mike Poirier, Dave Bonaccorsi  
IBM Middletown, RI, USA

John Hutchinson  
IBM Gaithersburg, MD, USA

Thanks also to the authors of *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665, that broke most of the ground on non-z/OS platforms and paved the way for this Redbooks publication:

Peter Daly, Thomas McManus, Karen Poyer, Julia Reder, Mohamed Shamseldin Salem, Jeffrey Slone and Vignesh Velusamyavindran.

## Become a published author

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

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you will 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:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

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

[ibm.com/redbooks](http://ibm.com/redbooks)

- ▶ Send your comments in an e-mail to:

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- ▶ Mail your comments to:

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

# Business Process Management production topologies for z/OS

This chapter introduces WebSphere Process Server for z/OS components. It also introduces topology patterns. Topology patterns for z/OS are compared to topology patterns for non-z/OS, described in Redbooks publication *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665.

This chapter contains the following sections:

- ▶ “Introduction” on page 2
- ▶ “WebSphere Process Server components” on page 6
- ▶ “WebSphere Process Server deployment environment patterns” on page 9
- ▶ “Selecting an appropriate topology” on page 17
- ▶ “Incorporating Fabric into the two-clusters topology” on page 22

Two topology patterns are described in Section 1.3, “WebSphere Process Server deployment environment patterns” on page 9:

- ▶ Single-cluster topology
- ▶ Two-clusters topology

Reasons for choosing one pattern over another are discussed in Section 1.4, “Selecting an appropriate topology” on page 17.

For WebSphere Process Server on non-z/OS platforms, topology choices have been given names like bronze, silver, or gold. With WebSphere Process Server for z/OS we have avoided using names that imply that one choice is better than another. Instead we discuss the advantages and disadvantages of a single-cluster topology compared to the two-clusters topology, to help you select a production topology that best meets your requirements.

## 1.1 Introduction

A WebSphere Process Server topology is the physical layout of the deployment environment required to meet your business needs for capacity, availability, and scalability.

When discussing topologies for WebSphere Process Server on non-z/OS platforms, the choices are usually described as follows:

- ▶ Single Cluster topology, or bronze topology
- ▶ Remote Messaging topology, or silver topology
- ▶ Remote Messaging and Remote Support, gold topology, or the ND7 topology
- ▶ Custom topology

The names bronze, silver, and gold come about because performance studies have shown that the scalability depends on the topology. Therefore gold is better for performance than silver, which in turn, is better than bronze.

With WebSphere Process Server for z/OS, names like this are misleading because it is not so clear what makes one topology better than another on z/OS. The choice of topology depends on the particular circumstances of each customer. In most cases it will not be necessary to split the components of WebSphere Process Server for z/OS across multiple clusters in order to achieve significant capacity, availability, and scalability.

This difference between WebSphere Process Server for z/OS and WebSphere Process Server on non-z/OS platforms stems from the different architecture of the underlying WebSphere Application Server for z/OS.

It is possible to have a mixed-stack approach with some parts of your infrastructure on a non-z/OS platform and some on z/OS. A project that created such a topology and compared it with a pure z/OS topology is described in the IBM White Paper *The Mixed Platform Stack Project: Deploying a secure SOA solution into z/OS*, WP101300, which can be obtained from the following Web page:

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

In this Redbooks publication we describe only pure z/OS topologies, but do not approach your choice of production topology in a dogmatic manner. There are good reasons to deploy on non-z/OS platforms, on z/OS, and on a mixed-stack. The correct choice for your organization depends on your unique requirements. These solutions are platform-independent, so they give you the flexibility to choose.

### 1.1.1 WebSphere Application Server for z/OS architecture

A WebSphere Application Server for z/OS server has an architecture (Figure 1-1) that includes a controller region, one or more servant regions, and a separate address space called the control region adjunct (or more simply the adjunct), which hosts the message engines.

By providing an adjunct address space with a separate Java™ Virtual Machine JVM™ for the message engines, the architecture of a single-cluster in WebSphere Application Server for z/OS is analogous to the two-clusters silver topology of a non-z/OS WebSphere Application Server that is described in Redbooks publication *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665.

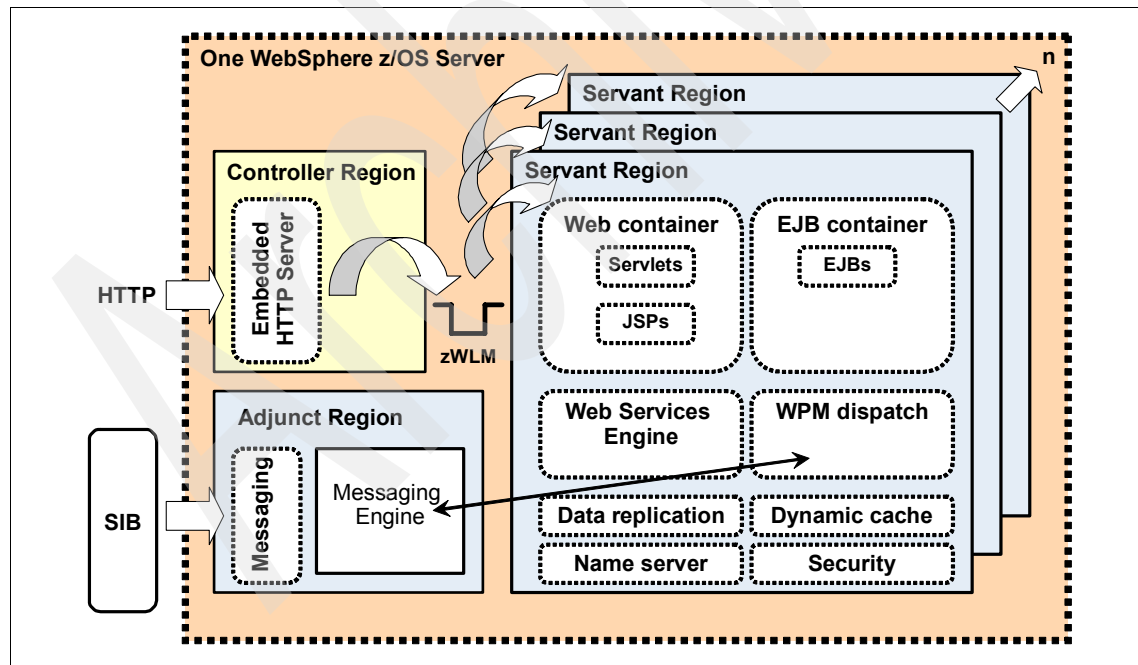


Figure 1-1 Architecture of a server in WebSphere Application Server for z/OS

Therefore the simplest WebSphere Process Server for z/OS production topology comprising one cluster starts at the silver topology. It is not normally necessary to split the messaging engines into a separate cluster in order to achieve performance and scalability.

The WebSphere Process Server for z/OS single-cluster topology is described in more detail in Section 1.3.1, “Single-cluster topology pattern” on page 10.

## **1.1.2 Hardware considerations**

A WebSphere Process Server for z/OS production topology does not normally need to concern itself with the zSeries® hardware, except that more clusters will imply more JVMs and more memory requirement.

You need to ensure that the zSeries machines in your parallel sysplex have enough capacity to support the topology you require.

## **1.1.3 HTTP servers, LDAP servers, and external security managers**

A production topology will include other non-WebSphere Process Server services such as one or more HTTP servers (for Web content) and firewalls.

With WebSphere Process Server for z/OS, load balancing of HTTP requests across an HTTP Server running on z/OS is performed by Sysplex Distributor and is not something that needs to be considered in your topology design. You will, however, have to define one or more Distributed VIPAs in order to exploit the Sysplex Distributor function.

If you are using a LocalOS, SAF-based user registry such as RACF, the user registry is not something you need to consider in your WebSphere Process Server for z/OS production topology. It should already be configured with a RACF database shared across the parallel sysplex. However, if you decide to use LDAP you must plan for high availability and disaster recovery if the LDAP is not running on z/OS.

In an Internet environment it is normal to install a proxy (such as WebSeal) in the De-Militarized Zone (DMZ), and to authenticate using an external security manager such as Tivoli® Access Manager. You will need to plan for the scalability and availability of any external authentication infrastructure.

## 1.1.4 WebSphere Process Server for z/OS scalability and availability

Scalability of WebSphere Process Server for z/OS is achieved by adding more servants rather than adding more cluster members, which is the normal practice when scaling WebSphere Process Server on non-z/OS platforms.

In general you only need multiple cluster members on z/OS to provide greater availability. It is natural to think that two is enough. However it is better to use three than two. This is known as the *Rule of Three*. The reason for preferring three is as follows.

When you run multiple cluster members, you are protecting against the failure of the entire logical partition (LPAR), or of some singleton address space on which the cluster member depends. If the controller region fails, you lose the capacity of that cluster member. When you lose one cluster member, then the remaining cluster member must take up the entire load and it may be able to do so. A sudden increase in load from 50% capacity to 100% may cause the remaining cluster member to fail too. When there are three cluster members, however, and one fails, there are two remaining cluster members to absorb 33% of the workload from the failed cluster member. This means a growth of only about 17% in the workload of each cluster member, which should not cause a problem.

On z/OS, when workload suddenly shifts to another cluster member, the effect is not as bad as it would be on a non-z/OS platform, because the re-assignment of resources to the remaining cluster by zWLM helps it deal with the extra workload. If you have configured the cluster so the number of servants vary with the workload, the impact of one cluster member failing may not cause a problem for one remaining cluster member.

However there will still be one controller region processing all the communications. You can run into a bottleneck with the number of SSL threads in the controller, for example. For this reason, although you may never encounter a situation where two cluster members is not adequate, we recommend three cluster members for maximum availability and stability, even on z/OS.

## 1.1.5 Summary

When planning your WebSphere Process Server for z/OS production topology, consider the following factors:

- ▶ The CPU required and the memory required to support all the JVMs.
- ▶ The number of clusters and cluster members required to support your business. Clusters with three cluster members are best.
- ▶ The number of servants per cluster member required to support the workload.

- ▶ If using LDAP, the network communication, high availability configuration, and disaster recovery requirements of the LDAP server.
- ▶ The number and location of any HTTP servers.
- ▶ Whether you should use WebSphere MQ for z/OS as a JMS provider.

For WebSphere Process Server on non-z/OS platforms, topologies can be configured using the Deployment Environment wizards. With WebSphere Process Server for z/OS, you configure a cluster or clusters using wizards that are located on the cluster panels.

## 1.2 WebSphere Process Server components

A number of components are created and used when configuring a WebSphere Process Server for z/OS topology. These components are discussed in this section.

### 1.2.1 Databases

A production WebSphere Process Server for z/OS cluster should use DB2 for z/OS rather than Derby to hold, store, and track information. You can choose to define all the tables for all components of WebSphere Process Server for z/OS in one database, or create databases for each component.

Unlike WebSphere Process Server on non-z/OS platforms, you do not need to use multiple databases for performance reasons. When using DB2 for z/OS, it is more a matter of administrative convenience to do this. There is no need to consider the physical location of databases when selecting your production topology because they are all in DB2 for z/OS.

Whether you choose one or two clusters, you must configure DB2 for z/OS in data sharing mode so that all cluster members can access the same data. For our two-clusters topology in MP cell we defined databases as follows:

- ▶ Common database (MPWPSDB)

Used as a repository for various components in WebSphere Process Server, this database needs to be created prior to starting WebSphere Process Server. It persists information regarding the following components:

- Application Scheduler
- Business Rules
- Mediations
- Recovery
- Relationships



- Selectors
- ▶ Business Process Choreographer (BPC) database (MPBPCDB)
 

Used by the Business Flow Manager and the Human Task Manager, this database needs to be created prior to starting BPC components.
- ▶ Business Process Observer database (MPOBSDB)
 

Used by the BPC Observer application to store event information from the CEI bus in an event collector table.
- ▶ Messaging engine databases (MP01ADB, MP01BDB, MP01CDB, MP01SDB)
 

These databases are used by the SCA system and application buses, the CEI bus, and BPC bus. We created each bus in a separate database because their Data Definition Language (DDL) uses the same table space names. It also allowed us to assign different bufferpools to each bus if required more easily.
- ▶ Event database (MPEVTDB)
 

This database persists information regarding the Event Service such as Common Based Events and key performance indicators.

## 1.2.2 Service integration buses

A service integration bus is a managed communication mechanism that supports service integration through synchronous and asynchronous messaging. A bus consists of interconnecting messaging engines. WebSphere Process Server makes use of the following service integration buses:

- ▶ SCA system bus
 

This bus is used to host queue destinations for SCA modules. The SCA runtime uses these queue destinations to support asynchronous interactions between components and modules.
- ▶ SCA application bus
 

This bus supports the asynchronous communication between WebSphere Business Integration Adapters and other SCA components.
- ▶ Common Event Infrastructure (CEI) bus
 

This bus is used to transmit common base events asynchronously to a CEI server.
- ▶ BPC bus
 

This bus is used for transmitting messages internally in the Business Flow Manager.

On non-z/OS platforms, experience has shown that running the message engines in the same cluster as the rest of WebSphere Process Server can cause a bottleneck. Therefore, a lot of emphasis is placed on ensuring that the messaging infrastructure runs in its own cluster. However, as discussed in Section 1.1.1, “WebSphere Application Server for z/OS architecture” on page 3, the fact that the message engines run in adjunct regions obviates the need to configure them in a separate cluster.

### 1.2.3 Business Process Choreographer (BPC)

BPC is an enterprise workflow engine that supports both business processes and human tasks. The core of the BPC configuration consists of the following components:

- ▶ Business Flow Manager

The Business Flow Manager provides services to run business processes within an application server.

- ▶ Human Task Manager

The Human Task Manager provides services to run human tasks within an application server.

Applications that make use of Business Flow Manager or Human Task functions must be deployed in a cluster where BPC has been configured. In the two-clusters topology, BPC is configured in the AppTarget cluster. This was the mpcl02.AppTarget cluster in our MP cell.

### 1.2.4 WebSphere Process Server applications

WebSphere Process Server for z/OS provides a variety of Web-based application tools, such as the following:

- ▶ BPC Explorer

BPC Explorer implements a generic user interface for interacting with business processes and human tasks. It is typically used to initiate and test business processes.

- ▶ BPC Observer

BPC Observer creates reports on processes that have been completed and displays the status of running processes.

- ▶ Business rules manager (BRM)

BRM assists business analysts in browsing and modifying business rule values.

In addition to these WebSphere Process Server-specific applications, Business Space powered by WebSphere can be used to interact with WebSphere Process Server for z/OS. Business Space is a browser-based, graphical interface included in WebSphere Process Server for z/OS that allows authorized users to create, manage, and integrate Web interfaces across the IBM BPM Suite.

These applications are all support functions, and therefore would be deployed into a Support and Messaging cluster in the two-clusters topology.

### **1.2.5 Common Event Infrastructure (CEI)**

CEI is an embeddable technology intended to provide event management services to applications that require those services. For service component event points that you monitor, events can be published to the CEI server and stored in the CEI database.

Depending on the extent CEI is used by applications, it may be necessary to define CEI in a separate cluster. In the two-clusters topology CEI is configured in the Support and Messaging cluster, which is remote from the cluster hosting the applications. That is probably enough separation.

## **1.3 WebSphere Process Server deployment environment patterns**

When using WebSphere Process Server on non-z/OS platforms, a deployment environment can be created using IBM-supplied deployment environment patterns. The deployment environment patterns included in the Integrated Solutions Console and the profile management tool represent the most common deployment environments that customers require. Each of the patterns centers around the number of WebSphere Process Server clusters and cluster members. You can think of deployment environment patterns as wizards that define a chosen topology.

With WebSphere Process Server for z/OS, the deployment environment patterns are not useful because the patterns they create are probably not what you want to use on z/OS. Instead, you configure each component of WebSphere Process Server for z/OS into the appropriate cluster using scripts or the wizards that are located on the cluster configuration panels.

WebSphere Process Server for z/OS contains three basic sets of functions that together form a complete production environment. Each of these functions can

be separated into individual, dedicated clusters, but on z/OS it is best to start by assuming they are deployed in the same cluster.

The three sets of functions in the WebSphere Process Server environment are as follows:

- ▶ **Application deployment target**  
The application deployment target is the set of servers to which you install your applications (human tasks, business processes, mediations, and so forth).
- ▶ **Supporting infrastructure**  
The supporting infrastructure is the CEI and other infrastructure services used to support your environment, such as the BPC Observer, BPC Explorer, Business Rules Manager and Business Space.
- ▶ **Messaging infrastructure**  
The messaging infrastructure is the set of servers used to provide asynchronous messaging support for your applications and for the internal messaging needs of the WebSphere Process Server components, such as the internal navigation queues used by long running business processes.

On non-z/OS systems there are well-known advantages to configuring the messaging infrastructure in a separate cluster but with WebSphere Process Server for z/OS there is probably no reason to do this. Message engines already run in separate adjunct regions, which is like running them in a separate cluster.

With WebSphere Process Server for z/OS, the main topology decision you face is whether it is worth splitting major application function, like WebSphere Business Services Fabric for z/OS, into an AppTarget cluster, while leaving the majority of the WebSphere Process Server for z/OS function deployed in the Support and Messaging cluster.

The single-cluster and two-clusters topologies are described in more detail in Section 1.4, “Selecting an appropriate topology” on page 17. The characteristics of these topologies are compared in order to help you choose between them.

### 1.3.1 Single-cluster topology pattern

The single-cluster topology pattern provides one cluster for all the functional components. The user applications, messaging infrastructure, CEI, and support applications are all configured in the same cluster.

On non-z/OS platforms, this topology is referred to as the bronze topology. Typically, the bronze topology is suggested for testing, proofs of concept, and

demonstration environments. However, on z/OS, a single-cluster topology is a viable production topology. This is because of the different architecture of WebSphere Application Server for z/OS.

As mentioned in Section 1.1.1, “WebSphere Application Server for z/OS architecture” on page 3, the fact that the message engines run in a separate adjunct region means that the drawbacks of the bronze topology do not apply to WebSphere Process Server for z/OS. A single-cluster topology on z/OS has characteristics that are analogous to those of the non-z/OS silver topology but there are other operational advantages, such as zWLM-managed servant regions, which make a single-cluster on z/OS what we can call a “polished silver” topology.

A single-cluster topology sample configuration for WebSphere Process Server is shown in Figure 1-2.

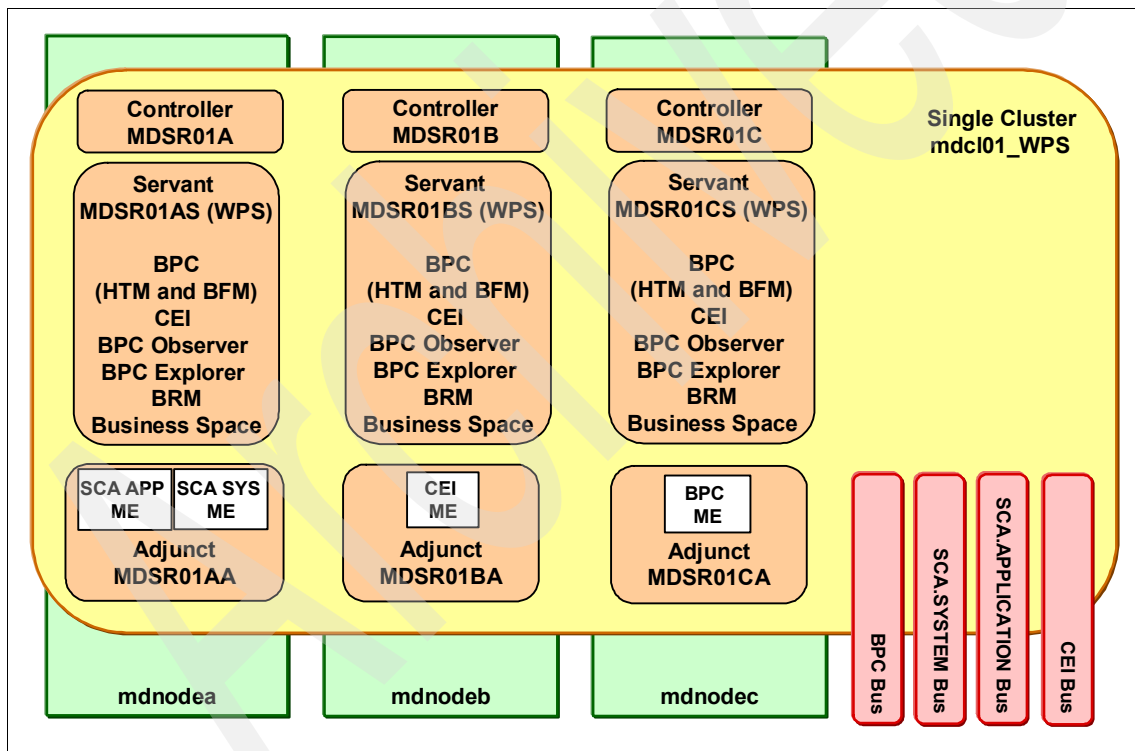


Figure 1-2 WebSphere Process Server for z/OS single-cluster topology

Note the following aspects of this example:

- ▶ All of the components are configured in a single cluster. In our single-cluster topology cell MD this cluster is called mdcl01\_WPS.
- ▶ The mdcl01\_WPS cluster is a member of all four of the required WebSphere Process Server buses:
  - SCA.SYSTEM bus
  - SCA.APPLICATION bus
  - CEI bus
  - BPC bus

The message engines execute in the adjunct regions.

- ▶ Each cluster member has a Business Process and Human Task container.
- ▶ All of the supporting infrastructure applications are configured in the cluster:
  - BPC Explorer
  - BPC Observer
  - Business Rules Manager
  - CEI
  - Business Space
- ▶ Each cluster member is an application deployment target.
- ▶ In Figure 1-2 on page 11, the messaging engines are shown running in specific adjuncts. See Table 1-1.

*Table 1-1 Messaging engines and their adjuncts, as shown in Figure 1-2 on page 11*

Adjunct	Messaging engine
MDSR01AA	SCA.SYSTEM and SCA.APPLICATION
MDSR01BA	CEI
MDSR01C	BPC

This is not the default configuration. By default, each cluster member is capable of running all four of the messaging engines, and the server that starts first automatically runs all four of the engines.

You can configure different Service Integration bus policies to control where message engines run. For more information, see Redbooks publication *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665.

- ▶ Each server scales by running multiple servants, not by adding more cluster members. More than one cluster member is required for availability reasons (and three is a good number) but not for scalability reasons.

The behavior of the messaging engines in a single-cluster topology is different than in a remote cluster. When the messaging engines and the applications are collocated, the default behavior is for message producers and consumers to use a local active messaging engine (if one is available). Thus, if you have two applications deployed to each cluster member that need to communicate asynchronously, once each message producer places messages in the queues, the message consumer on the machine where the engine is local consumes all of the messages produced. The consuming application only processes messages on the server with the local messaging engine.

Read and write local also creates a unique set of issues if you attempt to partition the destinations. When you create more than one active set of messaging engines, partitioning results. Each server's active messaging engines contain a portion of the queues assigned to that engine. Unfortunately, although you can attain additional throughput if there are active messaging engines on each server, partitioning can create issues for your applications.

If you partition destinations when the applications and messaging engines are in the same cluster, you will not be able to maintain message order. This is true even if you attempt to enable event sequencing in WebSphere Process Server. Partitioned destinations can also create unpredictable behavior if one or more messaging engines fails in a single-cluster topology. If you are prepared to endure unpredictable behavior and loss of message order, partitioning the destinations in a single-cluster topology may be acceptable. However, this configuration is discouraged.

**Note:** For information about workload sharing with queue destinations, refer to the WebSphere Application Server for z/OS Information Center at the following Web page:

[http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.pmc.zseries.doc/concepts/cjt0007\\_.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.pmc.zseries.doc/concepts/cjt0007_.html)

### 1.3.2 Two-cluster topology pattern

The two-clusters topology pattern is analogous to the Remote Messaging and Remote Support topology pattern for WebSphere Process Server on non-z/OS platforms where it is also known as the gold topology.

For non-z/OS platforms this is the preferred topology for production environments, but on z/OS there may be no need to define multiple clusters for performance reasons. We decided to create the two-clusters topology more for operational and availability reasons, as discussed in Section 1.4, “Selecting an appropriate topology” on page 17. The two-clusters topology is shown in Figure 1-3 on page 14.

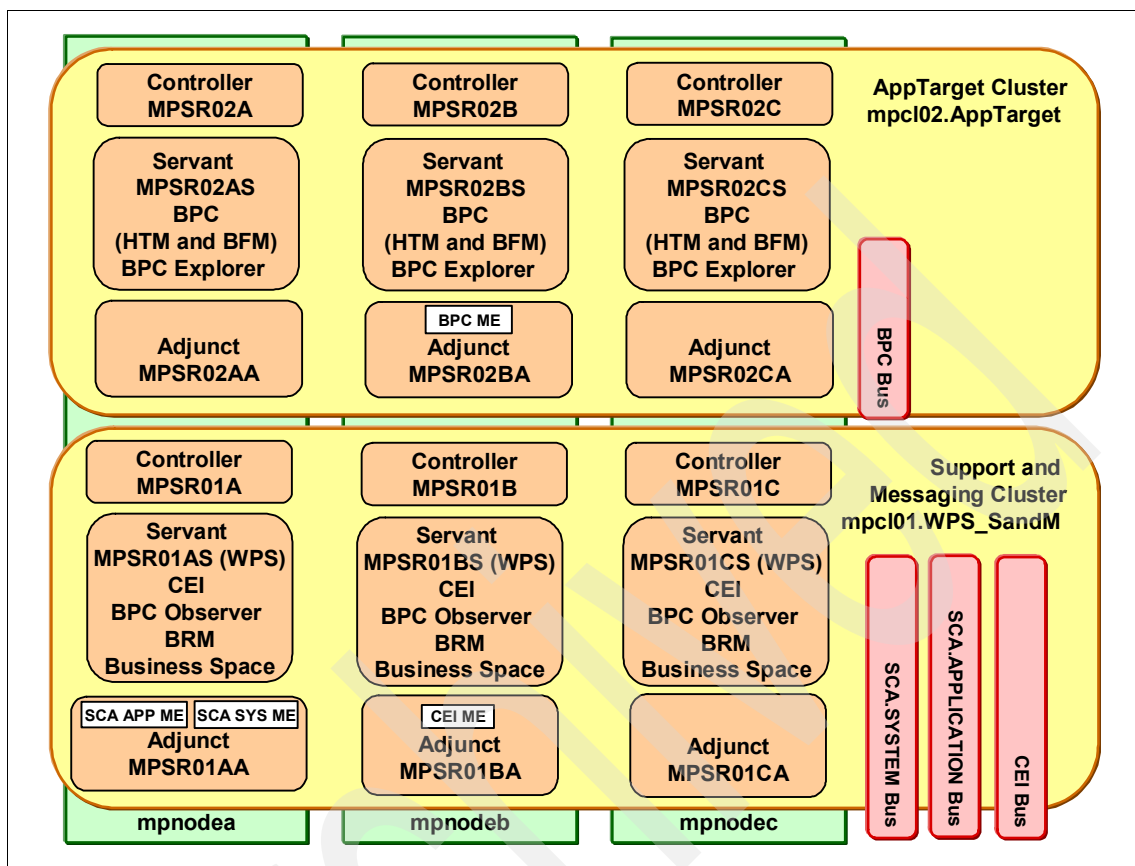


Figure 1-3 WebSphere Process Server for z/OS in the two-clusters topology

Figure 1-3 shows our MP cell, which is comprised of two clusters: mpcl01.WPS\_SandM (providing the Support and Messaging function), and mpcl02.AppTarget (running BPC and the applications).

Configuring this topology in WebSphere Process Server for z/OS is no different from configuring the single-cluster topology. It is largely a matter of invoking the appropriate wizard in the correct cluster.

For our MP cell we only used two nodes. Figure 1-3 shows three nodes in order to illustrate the possible assignment of the buses to specific adjuncts. Also, for availability reasons, we recommend using three nodes with three cluster members in Section 1.1.4, “WebSphere Process Server for z/OS scalability and availability” on page 5.



Note the following aspects of this example:

- ▶ All of the applications are deployed to the mpcl02.AppTarget cluster.
- ▶ The BPC is configured in the mpcl02.AppTarget cluster, so each cluster member has a business process container and a human task container.
- ▶ The mpcl01.WPS\_SandM cluster is a member of three of the four required WebSphere Process Server for z/OS buses:
  - SCA.SYSTEM bus
  - SCA.APPLICATION bus
  - CEI bus

The mpcl02.AppTarget cluster is a member of the BPC bus.

The BPC bus can be defined locally in the same cluster as BPC because the BPC message engine runs in an adjunct, so there is no performance concern with running it in the same cluster.

- ▶ The supporting infrastructure applications are configured in the mpcl01.WPS\_SandM cluster:
  - BPC Observer
  - Business Rules Manager
  - CEI
  - Business Space
- ▶ Remote SCA must be configured in the mpcl02.AppTarget cluster.

**Note:** The BPC Explorer would normally be deployed in the Support and Messaging cluster too, but the wizard only allows you to deploy BPC Explorer into a cluster that runs the Business Process Container. Therefore, it is shown deployed in mpcl02.AppTarget.

- ▶ In Figure 1-3 on page 14, the messaging engines are shown running in specific adjuncts. See Table 1-2.

*Table 1-2 Messaging engines and their adjuncts, as shown in Figure 1-3 on page 14*

Adjunct	Messaging engine
MPSR01AA	SCA.SYSTEM and SCA.APPLICATION
MPSR01BA	CEI
MPSR02BA	BPC

This is not the default configuration. You can customize a Service Integration Policy to assign a preferred server for the active message engine.

- ▶ As with the single-cluster topology, each server scales by running multiple servants, not by adding more cluster members. More than one cluster member is required for availability reasons (and three is a good number) but not for scalability reasons. The two-clusters topology allows you to add servants to cluster members in the AppTarget cluster. This is to assist in adding capacity to the applications, while keeping the capacity of the Support and Messaging infrastructure the same (or vice versa).

Note that the behavior of the messaging engines in the two-clusters topology is different from the behavior when the messaging engines are collocated with the applications. Because the messaging engines are in a remote cluster with respect to the applications, there is no preference for the message producers and consumers to use a local messaging engine. Each member of the mpcl02.AppTarget cluster, for example, connects to the appropriate bus and uses the remote messaging engine for that bus.

As with a single-cluster topology, if you partition destinations when the applications and messaging engines are in separate clusters, you will no longer have the ability to maintain message order. Any time you partition destinations you lose message order. This is true even if you attempt to enable event sequencing in WebSphere Process Server for z/OS.

In addition, partitioned destinations can create additional issues when the messaging engines are remote. By default, you have no control over which active messaging engine your applications will use at run time. This can create situations where two applications on the same server attach to two different messaging engines. If one application produces messages for one engine and the message consumer is using a different engine, stranded messages can result. For these reasons, partitioned destinations are strongly discouraged in the two-clusters scenario.

### 1.3.3 WebSphere MQ for z/OS queue sharing

A unique feature of WebSphere Process Server for z/OS is the ability to use WebSphere MQ for z/OS as a JMS provider. In a clustered environment, the ability to use shared queues with WebSphere MQ for z/OS allows you to run multiple instances of a message engine without the disadvantages of partitioning the queues.

Therefore, if your messaging workload exceeds the capacity of one adjunct region, you can consider configuring a WebSphere MQ for z/OS JMS provider using bindings mode for optimal performance and security. We were unable to test the configuration of WebSphere MQ for z/OS as a JMS provider when writing this Redbooks publication.

WebSphere MQ for z/OS applies to both a single-cluster or two-clusters topology and does not influence the choice between these topologies.

### 1.3.4 Custom topology patterns

If neither of these two topologies is suitable to your needs, you can create a custom topology. However, we do not recommend this unless you have a good reason to depart from one of the two topologies presented here. There are performance advantages to be gained by placing everything in one cluster and it is always a good idea to keep it simple.

Creating a custom topology with WebSphere Process Server for z/OS involves defining the number of clusters you want and deploying the components in the appropriate cluster. You must define remote SCA support in clusters that are remote from the SCA message engines.

If you choose to implement a custom topology pattern, note that it is generally unwise to move support components into non-default locations. For example, do not put the BPC Observer in the mpcl02.AppTarget cluster. The Observer is placed in the Support and Messaging cluster to maximize performance in the AppTarget cluster. Altering this structure will impact application performance.

## 1.4 Selecting an appropriate topology

Selecting an appropriate topology for your production environment depends upon several factors, including those listed below:

- ▶ Available hardware resources
- ▶ Application invocation patterns
- ▶ Types of business processes you plan to implement (interruptible versus non-interruptible)
- ▶ How heavily you intend to use the CEI
- ▶ Scalability requirements
- ▶ Administrative effort involved

On non-z/OS platforms, where resources are typically not shared between servers because they are running on different machines, it is important to consider the characteristics of all the hardware when planning your topology. On z/OS however, CPU, memory, data, message queues, and security profiles can all be shared between the logical partitions of a parallel sysplex. Therefore while you need to plan to have sufficient total capacity on your zSeries machines to support your workload, the topology of the hardware is not something that greatly influences your WebSphere Process Server for z/OS topology choice.

If you are able to use WebSphere MQ as a message provider, the potential bottleneck of running one message engine in an adjunct disappears. For this reason, if your applications make heavy use of messaging, it is more advantageous to use WebSphere MQ as a message provider than to split the message engines into a separate cluster or consider partitioning.

Our recommendation is to keep the number of clusters to a minimum and think carefully before using more than one. We did not describe the two-clusters topology in response to any known performance problem with the single-cluster topology in WebSphere Process Server for z/OS. Our motivation was related to availability and operability, as explained in Section 1.4.2, “The two-clusters topology pattern” on page 19.

Although the two-clusters pattern has some advantages, they come at the cost of remote access to the SCA buses. We were unable to perform performance tests to quantify this, and it is possible that the overhead of accessing SCA remotely offsets any benefits of having applications in a separate cluster. In addition, multiple clusters means more address spaces, which means greater CPU and memory requirements.

As you plan for your production environment, consider the advantages and disadvantages of single-cluster compared to two-cluster. These topologies are now compared in more detail.

### 1.4.1 The single-cluster topology pattern

With WebSphere Process Server for z/OS, a single-cluster topology may be suitable for your production environment, but because each server instance must run both the infrastructure and your applications, the heap requirements of the servant JVMs is more likely to necessitate the use of the 64-bit mode than the two-clusters topology.

With WebSphere Process Server for z/OS, you can configure zWLM so that asynchronous interactions are routed to different servants from those running synchronous requests, minimizing the impact of asynchronous requests without having to resort to a separate cluster. Thus, unlike non-z/OS systems, where a single cluster topology is typically used for proof of concept, development, and testing environments, a single cluster is a practical production topology on z/OS.

From an administrative and scalability perspective, the single-cluster topology has advantages. A single cluster where each member runs all the WebSphere Process Server components is easier to administer. Instead of several server instances in multiple clusters, you have a single cluster with fewer members. Remember that with WebSphere Process Server for z/OS, when the needs of your environment grow, you scale by simply running more servants. It is not

necessary to configure additional nodes and cluster members. The number of servants can be set to vary dynamically depending on the workload, so it is not necessary to manually add capacity in response to increased workload.

In a non-z/OS environment, when additional nodes or cluster members are added, all components are scaled at the same rate. For example, each additional cluster member adds more CEI processing whether you need it or not. If you have the messaging engines spread across server members using service integration policies, there is some additional administrative effort in creating and maintaining the policies. With WebSphere Process Server for z/OS, however, adding additional servants provides more capacity for the business workload without adding additional infrastructure overhead.

Expanding the cluster beyond three cluster members has no affect on message processing capability because the one-of-n high availability policy limits message processing capacity to the adjunct that is running the active message engine. The other adjuncts host message engines that act as stand-bys. Therefore, use service integration policies to spread the active message engines across the three cluster members, as shown in Figure 1-2 on page 11.

## 1.4.2 The two-clusters topology pattern

As discussed in the opening comments of Section 1.4, “Selecting an appropriate topology” on page 17, we did not describe the two-clusters topology in response to any performance problem with the single-cluster topology. Our motivation was related to availability and operability, specifically the following two concerns:

- ▶ WebSphere Business Services Fabric for z/OS will add to the start up times of a cluster member. On our system, a cluster member in the single-cluster topology took six minutes to start, while in the two-clusters topology the Support and Messaging cluster member took three minutes to start and the AppTarget cluster member took three minutes.
- ▶ The applications will probably have different availability needs compared to the Support and Messaging infrastructure. By placing WebSphere Business Services Fabric for z/OS, BPC, and applications in a separate cluster, those cluster members can be restarted separately from the Support and Messaging infrastructure.

Creating a second cluster, each with its own functions and applications, creates some additional administrative burden, but this is not excessive on z/OS. When you add cluster members and clusters on non-z/OS platforms, your performance tuning plan and the troubleshooting burden can expand greatly, but on z/OS you can classify different workloads using zWLM and obtain detailed performance data on each workload. The topology does not affect your ability to monitor your workloads and give them correct priorities.

Spreading messaging engines across the members of the messaging cluster adds the administrative burden associated with creating and maintaining policies, but this is mostly a one-off activity.

As was stated earlier, in WebSphere Process Server for z/OS you may not need to add a second cluster for scalability reasons, but having two clusters allows you to scale the cluster members hosting applications separately from those hosting the Support Messaging infrastructure.

Expanding the Support and Messaging cluster beyond three cluster members has no effect on message processing capability. The scalability of message processing is limited by the workload that can be processed by one adjunct, so the best you can do is spread the active message engines across three cluster members.

In summary, the two-clusters topology in WebSphere Process Server for z/OS offers all the advantages of the Remote Messaging and Remote Support topology on non-z/OS platforms. Because the application target cluster is only responsible for running your business integration applications, performance tuning and diagnostics are much simpler than in a single-cluster topology where one cluster runs everything. The two-clusters topology is also ideal for environments that make extensive use of CEI for monitoring and auditing, because CEI processing does not take place in the same cluster as the applications.

### 1.4.3 Custom topology

You can create a custom topology by defining as many clusters as you want and then launching the wizards that configure each component of WebSphere Process Server for z/OS on the appropriate cluster. The process of creating such a topology is essentially the same as that for the two-clusters topology we describe in this Redbooks publication.

You might want to depart slightly from the topologies we describe if your organization has no need for some of the WebSphere Process Server for z/OS components. If you know you will not use the CEI, for example, you can create a topology that does not include CEI support and the BPC Observer. Similarly, if your organization has governance rules that prevent you from taking advantage of the Business Rules Manager, you can remove it from your topology. These decisions do not significantly affect the choice of topology, except that when you configure fewer components it tends to make the single-cluster topology more practical.

In summary, consider a custom topology for good reasons. A single-cluster topology with WebSphere Process Server for z/OS is good, and it is possible to imagine reasons to adopt the two-clusters topology described in this Redbooks publication, but topologies with more clusters than this are animals that should be approached with extreme caution.

## 1.4.4 Condensed topology selection criteria

Table 1-3 provides a condensed list of the advantages and disadvantages to each of the topology patterns. Consider the information listed in Table 1-3 as a quick guide to selecting your production topology.

*Table 1-3 Topology selection considerations*

Consideration	Single Cluster topology	Two-cluster topology
Number of clusters to maintain	One cluster for all components	One cluster for applications One cluster for the support and messaging infrastructure
Hardware requirements	Minimum footprint for a high-availability production environment	More address spaces means more JVMs and more real storage requirements
Asynchronous interactions	Use z/WLM to isolate synchronous and asynchronous interactions	Use z/WLM to isolate synchronous and asynchronous interactions
Cross-server communication overhead	None, because everything is running in the same cluster members	Some overhead in accessing remote SCA buses from the application cluster
Long running processes, state machines and human tasks	No issue. zWLM is used to assign appropriate priorities	No issue. zWLM is used to assign appropriate priorities
Heavy CEI activity	Heavy CEI use may impact applications	Suitable environment for heavy CEI use
Scalability	Easy to scale by adding servant regions. zWLM manages the distribution of work across the servants and can dynamically control the number of servants in response to the workload.	Easy to scale by adding servant regions. zWLM manages the distribution of work across the servants and can dynamically control the number of servants in response to the workload. Allows selective scaling of either the application cluster or the Support and Messaging cluster

## 1.5 Incorporating Fabric into the two-clusters topology

A WebSphere Process Server for z/OS production topology can also include other products from the WebSphere business integration portfolio.

Chapter 3, “Incorporating WebSphere Business Services Fabric into a production topology on z/OS” on page 227 presents information about configuring WebSphere Business Services Fabric for z/OS in both the single-cluster and the two-clusters topology pattern.

If you include WebSphere Business Services Fabric for z/OS in your production environment, it is likely that you will want to use the two-clusters topology. The reason for choosing the two-clusters topology is not related to any known performance issue with running Fabric in a single cluster with WebSphere Process Server for z/OS, but is recommended for practical reasons related to availability and operability as described in Section 1.4, “Selecting an appropriate topology” on page 17.

If you choose to include WebSphere Business Services Fabric for z/OS in a single-cluster topology, be sure to test the scalability and verify that the restart times of the cluster members are satisfactory.

### 1.5.1 WebSphere Business Services Fabric

When WebSphere Business Services Fabric for z/OS is added to the two-clusters topology, the mpcl02.AppTarget cluster is a member of the WebSphere Business Services Fabric bus (fabricbus), and the WebSphere Business Services Fabric core applications are added to the AppTarget cluster. This topology is illustrated in Figure 1-4 on page 23.



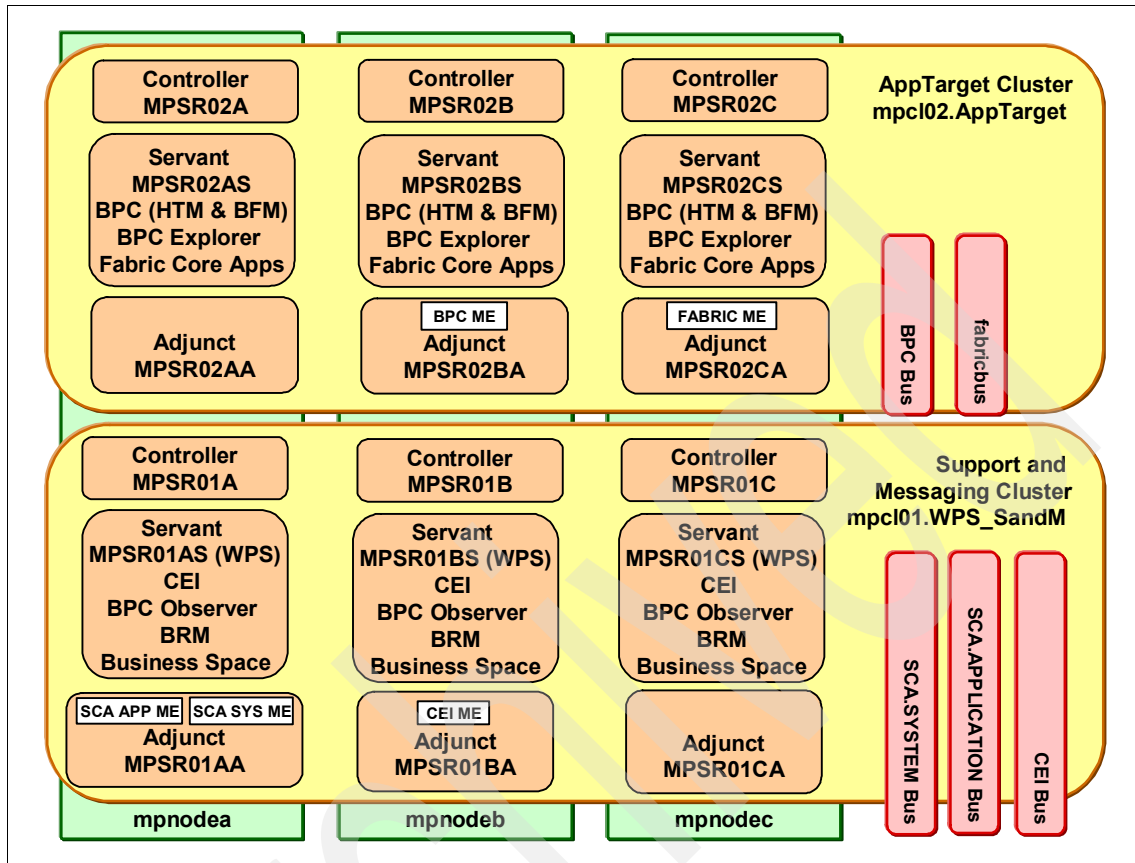


Figure 1-4 WebSphere Business Services Fabric for z/OS in the two-clusters topology

Chapter 3, “Incorporating WebSphere Business Services Fabric into a production topology on z/OS” on page 227 provides detailed instructions on how to add WebSphere Business Services Fabric for z/OS to both a single-cluster and two-clusters topology.

Notice that the topology shows both clusters in the same nodes. This means that the AppTarget cluster hosting Fabric must run the same service level of WebSphere Application Server for z/OS as the Support and Messaging cluster. This is not a cause for concern, but be aware that changes to WebSphere Application Server for z/OS service levels affects both clusters.



# Configuring WebSphere Process Server for z/OS Network Deployment

This chapter describes how to plan and configure a production WebSphere Process Server for z/OS topology that includes WebSphere Business Services Fabric for z/OS.

The step-by-step instructions in this chapter implement a single-cluster topology but where there are differences for the two-clusters topology, these differences are highlighted.

This chapter has the following sections:

- ▶ “Overview” on page 26
- ▶ “Plan the Network Deployment configuration” on page 29
- ▶ “Prepare a WebSphere cell, RACF, and DB2” on page 59
- ▶ “Create the deployment manager and an empty node” on page 70
- ▶ “Add WebSphere Process Server to the DMGR” on page 72
- ▶ “Configure DB2 z/OS using createDB.sh” on page 79
- ▶ “Adding WebSphere Process Server to the empty nodes” on page 88
- ▶ “Configure the common database in DB2 for z/OS” on page 98
- ▶ “Create a secondary node” on page 101
- ▶ “Create WebSphere Process Server clusters” on page 108

- ▶ “Define HTTP servers or proxies” on page 120
- ▶ “Service Component Architecture” on page 121
- ▶ “Remote SCA in AppTarget cluster” on page 128
- ▶ “Configuring CEI using the Integrated Solutions Console” on page 129
- ▶ “Business Process and Human Task containers” on page 146
- ▶ “BPC Explorer” on page 155
- ▶ “Configuring BPC Event Collector and BPC Observer” on page 159
- ▶ “Verify BPC” on page 167
- ▶ “Verify BPC Event Collector and BPC Observer” on page 176
- ▶ “Configuring Business Space powered by WebSphere” on page 200
- ▶ “People Directory Provider” on page 216
- ▶ “Failed events” on page 223

## 2.1 Overview

There are two main ways to approach the task of configuring a WebSphere Process Server for z/OS cluster:

- ▶ Configure WebSphere Process Server for z/OS in a stand-alone server, federate it into a Network Deployment cell, and then create a cluster by cloning that server.
- ▶ Configure a Network Deployment cell with empty nodes and then add WebSphere Process Server for z/OS to the nodes.

An advantage of the first option is that it is easier and quicker because when you run the `zWPSConfig.sh` as part of augmenting the stand-alone server, it configures almost all components of WebSphere Process Server for z/OS at that time. You must still configure Business Space as a separate task, however.

The disadvantages of starting with a stand-alone server are as follows:

- ▶ All the components are in one server, which means that after federating it into a Network Deployment cell you will end up with a single-cluster topology.
- ▶ The names of buses include the cell name of the server in which they were created.

When the buses are transferred into a Network Deployment cell during federation of the stand-alone server, they retain their original names. Therefore, after federation the names of the buses will contain the cell name of the old stand-alone server and not the cell name of the Network Deployment cell. This is not a serious issue but one of which you need to be aware.

If you intend to adopt a single-cluster topology then you may prefer to follow the procedure described in White Paper WP101253, *WebSphere Process Server for z/OS V.6.1 “Easy” Network Deployment Configuration*, which explains that approach. The White Paper can be obtained at the IBM Technical Sales Library at the following Web page:

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

To create the two-clusters topology, we need the flexibility to control where each component is deployed, so in this Redbooks publication, we describe how to add WebSphere Process Server for z/OS to a Network Deployment cell composed of a deployment manager and two or more empty nodes.

In summary, the procedure is as follows:

1. Plan the Network Deployment configuration.

This is discussed on page 29. We describe how to use a spreadsheet to plan the cell. The spreadsheet also generates the response file input to the zProfile Management Tool (zPMT). This section also covers planning security and the use of DB2 for z/OS.

2. Prepare a WebSphere cell, RACF, and DB2.

This is discussed on page 59. We describe how to use the zPMT to define the deployment manager's node and the empty nodes that will support the cluster(s).

3. Add WebSphere Process Server to the DMGR.

This is discussed on page 72. In this step you augment the deployment manager node

4. Configure DB2 z/OS using createDB.sh.

This is discussed on page 79. In this step you create the WebSphere Process Server databases in DB2 z/OS and execute the Data Definition Language (DDL) to create all the tables, indexes and views used by WebSphere Process Server for z/OS.

5. Adding WebSphere Process Server to the empty nodes.

This is discussed on page 88. In this step you augment the first empty node to support WebSphere Process Server for z/OS and then federate it into the cell.

6. Configure the common database in DB2 for z/OS.

This is discussed on page 98. We recommend that you use createDB.sh to generate the DDL that defines all tables, indexes and view in DB2 for z/OS. However, this step describes how to do this if you do not wish to use createDB.sh.

7. Create a secondary node.

This is discussed on page 101. You need at least one secondary node in order to create a horizontal cluster. The process for creating the secondary node using the jobs generated by the zPMT is not described, but we explain how to augment a secondary node to support WebSphere Process Server for z/OS. Repeat this step for each LPAR that hosts a node used by your clusters.

8. Create WebSphere Process Server clusters.

This is discussed on page 108. Decisions in this step depend on whether you wish to create a single-cluster topology or the two-clusters topology.

9. Define HTTP servers or proxies.

This is discussed on page 120. You will almost certainly require one or more HTTP Servers or external proxies in front of your WebSphere Process Server for z/OS cluster(s). In this step we describe the options.

10. Add the following WebSphere Process Server for z/OS components to the clusters once the WebSphere Application Server for z/OS cluster topology has been configured:

- Service Component Architecture (SCA)

This is discussed on page 121.

- Remote SCA in AppTarget cluster when using a two cluster topology only

This is discussed on page 128.

- Configuring CEI using the Integrated Solutions Console (Add CEI before BPC)

This is discussed on page 129.

- Business Process and Human Task containers (BPC)

This is discussed on page 146.

- BPC Explorer (BPCE)

This is discussed on page 155.

- Configuring BPC Event Collector and BPC Observer

This is discussed on page 159.

11. Verify that the Business Process and Human Task containers, and related tools like Explorer and Observer, are functioning normally.

- Verify BPC

This is discussed on page 167.

- Verify BPC Event Collector and BPC Observer

This is discussed on page 176.

12. Configure Configuring Business Space powered by WebSphere and verify that it is working properly. This is discussed on page 200.

At this point the WebSphere Process Server for z/OS configuration is complete. If your applications make use of human tasks and use an external user registry for People resolution, you may need to configure a People Directory Provider, as detailed in Section 2.21, “People Directory Provider” on page 216.

This chapter provides an overview of how to manage Failed events, which is detailed in Section 2.22, “Failed events” on page 223.

After completing the tasks in this chapter you can add WebSphere Business Services Fabric for z/OS to the topology, as described in Chapter 3, “Incorporating WebSphere Business Services Fabric into a production topology on z/OS” on page 227.

If you have experience with WebSphere Application Server for z/OS, and access to someone with good DB2 for z/OS skills, you can configure WebSphere Process Server for z/OS in about two days. This assumes that you do not lose too much time resolving problems and have planned properly.

This Redbooks publication aims to help you to avoid the human errors that may occur during configuration of WebSphere Process Server for z/OS. With this in mind, Chapter 4, “Troubleshooting WebSphere Process Server for z/OS” on page 305 details most of problems we experienced, with their solutions.

## 2.2 Plan the Network Deployment configuration

This section describes the key planning tasks to be performed before starting to configure WebSphere Process Server for z/OS.

Many of the planning items involve choices, so we discuss some key early decisions you will make.

### 2.2.1 Hardware and software pre-requisites

The latest information about the supported hardware and software configurations for WebSphere Process Server for z/OS can be found on the following Web page:

<http://www.ibm.com/support/docview.wss?uid=swg27007565>

The Web page instructs you to use the latest service level of IBM WebSphere Application Server for z/OS 6.1. It is important to note that WebSphere Process Server for z/OS V6.1.2 requires at least WebSphere Application Server for z/OS service level 6.1.0.17.

The Derby embedded database shipped with WebSphere Process Server for z/OS is used only for a configuration with a single JVM such as a stand-alone server with one servant. Although the Derby Network Server is supported, it is not suitable for use as a production database because of performance and integrity considerations.

If you intend to use more than one servant or a Network Deployment configuration of any kind, plan to use one of the supported releases of DB2 for z/OS. DB2 for z/OS V9.1 has improved the performance of LOB management and relaxed some security rules on setting the current schema. Both these changes benefit WebSphere Process Server for z/OS and for that reason DB2 for z/OS V9.1 is recommended.

We used the WebSphere Application Server Toolkit (AST) upgraded to V6.1.1.6 when using the zProfile Management Tool (zPMT) to customize our WebSphere cell.

The AST is upgraded using the Rational® Application Developer (RAD) upgrade tool that is installed when you install the AST. The best way to upgrade the AST is to download the fix packs to your workstation and then configure the RAD update tool to upgrade from the downloaded files. To locate AST fix packs search on WebSphere application server toolkit download at the following Web page:

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

In summary the software we used was as follows:

- ▶ z/OS 1.9
- ▶ DB2 for z/OS 9.1
- ▶ WebSphere Application Server for z/OS 6.1.0.17
- ▶ WebSphere Process Server for z/OS 6.1.2
- ▶ WebSphere Business Services Fabric for z/OS 6.1.2
- ▶ WebSphere Application Server Toolkit (AST) 6.1.1.6
- ▶ Tivoli Directory Server for z/OS 6.0

## 2.2.2 Installing WebSphere Process Server for z/OS using SMP/E

In this Redbooks publication we do not explain how to install WebSphere Process Server for z/OS using SMP/E. That is described in the document, *Program Directory for WebSphere Process Server for z/OS V6.1.0*, GI11-2880. A printed copy is shipped with the product, or you can download the Program Directory from the IBM Publications Center at the following Web page:

<http://www.ibm.com/shop/publications/order>

Search on the document number, GI11-2880.



WebSphere Application Server for z/OS is included with WebSphere Process Server for z/OS, as shown in Figure 2-1.

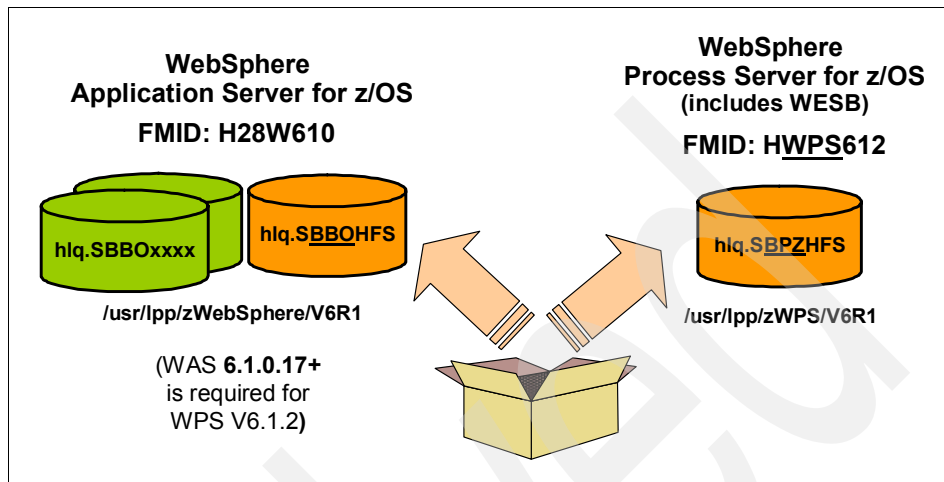


Figure 2-1 Packaging of WebSphere Process Server

The starting point for this section is after you have successfully unloaded the products from the installation media, installed them using SMP/E, and mounted the product HFS data sets into your UNIX® hierarchical file system (HFS). If you ordered WebSphere Process Server for z/OS using CBPDO, the installation using SMP/E involves the following tasks:

- ▶ Obtaining the latest copy of the Program Directory document.
- ▶ Obtaining the latest Preventative Service Planning (PSP) document for WebSphere Application Server and WebSphere Process Server. The PSP has the list of the latest important PTFs to install. The name of this document and instructions on obtaining it are in the Program Directory.

If you ordered the products using IBM SystemPac® or ServerPac, follow the instructions in *ServerPac: Installing your Order that ships with your SystemPac or ServerPac* that came with the product. Also, the product Information Centers contain information about the installation using SMP/E.

**Note:** WebSphere Process Server for z/OS includes WebSphere Enterprise Service Bus for z/OS and there is only the one HFS data set that contains both products. Therefore, after you have installed WebSphere Process Server for z/OS with SMP/E you will have hlq.SBBO\* target libraries belonging to WebSphere Application Server for z/OS and hlq.SBPZ\* libraries for WebSphere Process Server for z/OS, but you will not find hlq.SBSB\* data sets for WebSphere Enterprise Service Bus for z/OS.

## 2.2.3 Choosing the topology

Before starting to configure WebSphere Process Server for z/OS you must know the target topology you are aiming for. Topologies are discussed in detail in Chapter 1, “Business Process Management production topologies for z/OS” on page 1.

We configured two topologies:

- ▶ A single-cluster topology in a cell with an MD prefix
- ▶ A two-clusters topology with an MP prefix

The spreadsheet that we describe in Section 2.2.5, “Planning using a spreadsheet” on page 33 can be used to help plan both topologies.

## 2.2.4 Planning dataset names and disk space

We recommend that you use ZFS™ rather than HFS when allocating the configuration datasets for the nodes.

A cell running WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS will require a large configuration dataset for each node. After configuring WebSphere Business Services Fabric for z/OS in a WebSphere Process Server for z/OS network deployment configuration, the deployment manager's configuration occupied 1100 CYLs while each node occupied 1000 CYLs. These values were added to the spreadsheet used to plan the cell. WebSphere Process Server for z/OS V6.1.2 will require configuration datasets that are around 800–900 CYLs.

When you have large datasets like this, dumping them can be tricky because a dump of all the configuration datasets on disk can easily exceed 3000 CYL. We suggest that you speak to your disk space administrator about setting up a pool of 3390-9 disks to hold the configuration datasets and dumps of the WebSphere Process Server for z/OS cell.

You should also request that DFSMS maintains sufficient free space on these volumes that will allow the node configuration datasets to extend. Running out of space on a volume when configuring WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS is the cause of many strange problems and will probably result in having to restore the configuration to the point before the failure.

You will need to plan the dataset high-level qualifier used by the configuration datasets because your DFSMS ACS routines will need to be updated to map that high-level qualifier to the new storage pool of 3390-9s.

## 2.2.5 Planning using a spreadsheet

The White Paper WP101030, *WebSphere z/OS V6.1 - A Top Down Configuration Approach* explains how to use a spreadsheet to create a WebSphere Application Server for z/OS V6.1 cell. You can obtain that paper at the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101030>

We downloaded the *PRS1331 - zPMT in AST 6.1.1 WebSphere V6.1 Configuration.xls* spreadsheet from the following Techdocs Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS1331>

We made some enhancements to it to help plan a cell that contains WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS.

**Single-cluster topology spreadsheet:** The spreadsheet we used to plan our single-cluster topology MD cell is called MD - zPMT WPS V6.1.2 Configuration.xls. It can be found in the `zos_planning_spread` sheet directory in the additional material that accompanies this Redbooks publication. See Appendix A, “Additional material” on page 351.

We made several changes to the standard WebSphere Application Server for z/OS spreadsheet in order to add support for WebSphere Process Server for z/OS. These changes are listed on the Documentation worksheet in MD - zPMT WPS V6.1.2 Configuration.xls.

You do not have to use a spreadsheet to plan your WebSphere Process Server cluster but we suggest it as a quality-control tool. Many errors that occur when configuring WebSphere Process Server for z/OS come from poor or inconsistent choices, or simple typing errors. By using the spreadsheet you will avoid those kinds of errors and the spreadsheet becomes a record of the choices you made and what you did. It becomes your audit trail.

The naming convention used in the spreadsheet follows that proposed by the IBM Washington Systems Center. If you do not want to follow that naming convention, do not give up on the spreadsheet. Experience has shown that it is better to change the formulae in the spreadsheet so they match your naming convention, and then use the generated response files for input into zPMT. This is much more likely to result in a successful installation than working from ad-hoc notes or typing all the input into zPMT manually.

## How to use the spreadsheet

This section explains how to use the extensions we made to the WebSphere Application Server for z/OS spreadsheet to add support for WebSphere Process Server for z/OS.

We do not describe in detail how to use the spreadsheet for configuring the underlying WebSphere Application Server for z/OS nodes because that is already covered in White Paper WP101030, *WebSphere z/OS V6.1 - A Top Down Configuration Approach*. However, the checklist worksheet provides a summary of the steps involved. It guides you through customizing the Variables worksheet, copying the response-file worksheets from the spreadsheet, and importing them into zPMT. This allows you to generate the .CNTL and .DATA datasets that you use to configure the cell and each node.

This is the same process that is documented in WP101030. However, there are some important differences compared to the standard spreadsheet provided in PRS1331 on the Techdocs web site.

- ▶ The worksheet names have been slightly changed and there are separate worksheets for configuring the primary and secondary empty nodes.
- ▶ The checklist worksheet summarizes the tasks so you can check them off as you complete them. The checklist contains any additional tasks that we suggest you complete before starting to execute the BBO\* jobs generated by the zPMT. These are the tasks related to the best practices described in Section 2.2.8, “Using intermediate symbolic links and ICFCAT aliases” on page 45 and are shown in Figure 2-2.

Prepare the primary node
Check and run job MDNAALIA from the JCL_P worksheet to define ICFCAT aliases for the Deployment Manager and the Primary node.
Check and run job MDNASYML from the JCL_P worksheet to define intermediate symbolic links to the current product mount points.
Create directory /wasv61config/mdcell before running this job.

Figure 2-2 Checklist tasks to define ICFCAT aliases and intermediate symbolic links

You start by completing the Variables worksheet and then follow the checklist worksheet from top to bottom. Each major heading or task in the checklist has a matching section in this Redbooks publication so you can locate additional information about any step by consulting this document. Many fields also have comments that provide help.

- ▶ Just below the top of the Variables worksheet there are fields in which you choose the LPARs that will host the primary and a secondary node. You also assign a single character qualifier to each node as shown in Figure 2-3 on page 35.

	Primary Node	Secondary Node
Enter System Names of the LPARs that will host nodes ==>	SC42	SC53
Enter One Letter Node (LPAR) Qualifier ==>	a	b

Figure 2-3 Variables controlling the node topology

Set the names of the LPARs that will host nodes and choose a single character node identifier. The values in the Primary Node column are used in the EmptyNode\_P worksheet that generates the response file you will use when defining the primary node. The values in the Secondary Node column are used to generate the response file on the EmptyNode\_S worksheet. You will use that when defining a secondary node.

If you plan to create a cluster across more than two nodes, create the first secondary node, and then return to the Variables worksheet to alter the LPAR name and node identifier in the Secondary Node column so the values apply to the third LPAR. The EmptyNode\_S worksheet will then contain values relating the third node and you can import that response file into zPMT to define the third node. The sections in the checklist that apply to a secondary node will contain customized instructions for the third node.

- Below the node topology there are fields that control the cluster topology. To the right of the **Choose WPS Topology (single-cluster | two-clusters)** field there is a list box with two choices. When you choose the single-cluster topology only one cluster name is generated are shown in Figure 2-4.

Choose WPS Topology (single-cluster   two-cluster) ==>	single-cluster	Generated cluster names
Enter 2-character cluster identifier for WPS cluster ==>	01	mdcl01_WPS
For a single-cluster topology, this field is not applicable.		

Figure 2-4 Choosing the single-cluster topology

When you choose the two-clusters topology it causes different cluster names to be generated (Figure 2-5). The cluster name includes the 2-digit cluster identifier you decided to use.

Choose WPS Topology (single-cluster   two-clusters) ==>	two-clusters	Generated cluster names
Enter 2-digit identifier for WPS Support and Messaging cluster ==>	01	mdcl01.SandM
Enter 2-digit cluster identifier for WPS AppTarget cluster ==>	02	mdcl02.AppTarget

Figure 2-5 Choosing the two-clusters topology

There is nothing special about the cluster names and you can change the formulae to generate any cluster name you like. Note that instructions in the checklist worksheet differ depending on the choice of topology.

- Further down the Variables worksheet, the 2-digit cluster identifiers are used to assign a unique port range to each cluster, as shown in Figure 2-6.

TCP/IP Port Allocations						
Enter Starting TCP/IP Port ==>		20000			Cluster number	
	Daemon	DMGR	Node Agent		01	02
JMX SOAP		20010	20020		20040	20060
Bootstrap/ORB IIOP	20002	20011	20021		20041	20061
ORB SSL	20003	20012	20022		20042	20062
Cell/Node Discovery		20013	20023			
Node Multicast Discovery Port			20024			
High availability manager communication port		20015	20025		20043	20063
Session Initiation Protocol (SIP) Port					20044	20064
Session Initiation Protocol (SIP) Secure Port					20045	20065
Administrative console port		20018			20046	20066
Administrative console secure port		20019			20047	20067
Service Integration port					20050	20070
Service Integration secure port					20051	20071
Service Integration MQ Interoperability port					20052	20072
Service Integration MQ Interoperability secure port					20053	20073
HTTP					20048	20068
HTTP SSL					20049	20069
Interim Values for Stand-Alone Server's Daemon						
ORB IIOP	20000					
ORB SSL	20001					
Additional Node Agent Ports						
CSIV2 SSL MUTUALAUTH LISTENER ADDRESS			20030		20054	20074
CSIV2 SSL SERVERAUTH LISTENER ADDRESS			20031		20055	20075
DRS CLIENT ADDRESS			20032			
NODE IPV6 MULTICAST DISCOVERY ADDRESS			20033			
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS			20034		20056	20076

Figure 2-6 Unique port range assigned to each cluster

- Specify the details of the DB2 for z/OS system that hosts your WebSphere Process Server for z/OS databases and plan out the database names, schemas, volumes, and bufferpools you will use. Part of the table used to plan these DB2 names is shown in Figure 2-7.

	Common	APP ME	BPC ME	CEI ME	SCA ME
Abbreviation used to build names ==>	WPS	A	B	C	S
Database Name ==>	MDWPSDB	MD01ADB	MD02BDB	MD01CDB	MD01SDB
Schema Name ==>	MDWPS	MD01A	MD02B	MD01C	MD01S
Database storage group ==>	MDWPSSG	MDWPSSG	MDWPSSG	MDWPSSG	MDWPSSG
Database volumes ==>	***	***	***	***	***
Database VCAT ==>	WASCFG	WASCFG	WASCFG	WASCFG	WASCFG
Database 4K buffer pool ==>	BP1	BP3	BP3	BP3	BP3
Database 32K buffer pool ==>	BP32K	BP32K	BP32K	BP32K	BP32K
Database index buffer pool ==>	BP2	BP4	BP4	BP4	BP4
CEI 4K buffer pool ==>					
8K buffer pool ==>					
16K buffer pool ==>	BP16K1				

Figure 2-7 Part of the table used to plan database names

The numeric part of the schema and database names of the SIBs is derived from the cluster identifier. This allows you to potentially create more than one cluster running BPC with its own SIB, for example.

The light green fields contain generated defaults that you can change by altering the formulae, which you would not normally update when configuring a new cell. The dark green fields, however, must be updated to reflect your requirements. The bufferpool fields will not allow you to choose BP0, BP8K0, or BP16K0.

- The LDAP variables shown in Figure 2-8 are used on the WPS\_LDAP worksheet to generate the sample Idif file. We used a Tivoli Directory Server for z/OS LDAP server with WebSphere Business Services Fabric for z/OS.

LDAP Variables			
Enter hostname for LDAP ==>	wpsplex.itso.ibm.com		
Enter port for LDAP ==>	2389		
Unqualified LDAP Administrator userid ==>	LDAPADM		
Attribute for suffix ==>	o		
LDAP suffix ==>	ibm	Example Suffix ==>	o=ibm
Attribute for users ==>	uid		
Container for users ==>	People	Example User ==>	dn: uid=LDAPADM,cn=People,o=ibm
Attribute for groups ==>	cn		
Container for groups ==>	Groups	Example Group ==>	dn: cn=MDBPEADG,cn=Groups,o=ibm
Attribute for mail ==>	mail		
Mail provider's mail domain	itso.ibm.com		

Figure 2-8 The LDAP variables on the Variables worksheet

After completing the Variables worksheet you can start to build the cell by following the checklist. However you might want to look at the WPS\_RACF worksheet before you start.

The WPS\_RACF worksheet is where all the user IDs and groups that are used by WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS are planned. Some of the fields on the WPS\_RACF worksheet are dark green, meaning that you are expected to set your values in those fields (for example, the password values on the WPS\_RACF worksheet). Values on this worksheet are used in the xxRACF job on the JCL\_P worksheet. That job creates all the user IDs required with WebSphere Process Server for z/OS.

If you plan to use an LDAP user registry for the WebSphere Process Server for z/OS cell rather than RACF, review the WPS\_LDAP worksheet and ensure that the user IDs and groups are added to your LDAP server. The user IDs and groups on the WPS\_LDAP worksheet are derived from those on the WPS\_RACF worksheet because you must define the cell's started task user IDs, groups and administrator user ID to the LDAP server. Because these have to be defined in RACF even when the cell is using LDAP user registry for authentication and authorization, it is less confusing if the same names are used in both registries.

## 2.2.6 Planning security

When you enable security in a WebSphere Process Server for z/OS cell, you have to choose a user registry. You can use a LocalOS, SAF-based user registry like RACF, an LDAP user registry, a Custom registry, or a Federated registry.

In preparing this Redbooks publication we used a RACF user registry, and therefore there are several steps that help you configure WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS to use RACF.

The WPS\_RACF worksheet on the spreadsheet described in Section 2.2.5, “Planning using a spreadsheet” on page 33 can be used to help plan all the user IDs, groups and EJBROLE profiles that would be needed in RACF.

We do not describe in detail how to configure an LDAP user registry, but the spreadsheet includes a worksheet called WPS\_LDAP, which contains entries in Idif format for loading into Tivoli Directory Server for z/OS. Details about the LDAP server are set on the Variables worksheet. The users and groups are derived from the other worksheets. Most of them come from the WPS\_RACF worksheet where the majority of WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS user IDs and groups are planned.

It is important to have a clear security plan before you start. The zProfile Management Tool (zPMT) will generate JCL and RACF commands that configure RACF to support the needs of WebSphere Application Server for z/OS, but you must create your own commands to support the needs of WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS.

We suggest that you use the spreadsheet together with jobs MDRACF and MDEJBROL that are provided in the additional material that accompanies this Redbooks publication and are described in Appendix A, “Additional material” on page 351. For most of the roles we defined a RACF group that would uniquely provide access to that role. This would allow us to give a user ID that role by connecting it to that group. Some of the EJBROLE profiles were set to UACC(READ), which is the same as making the role available to all authenticated users.

For some roles we defined a new user ID that was connected to the role group. For example, user ID MDBPEADM was defined in group MDBPEADG, which was on the access list for the EJBROLE MD.BPESystemAdministrator. Obviously you can use your existing user IDs and simply connect them into the correct role groups. However, when you configure BPC you are prompted for both a user ID and a group that is associated with the roles of BPESystemAdministrator and HumanTaskSystemAdministrator. We thought it was better to create a specific task user ID for this, rather than to use an existing, personal, user ID that might not be persistent in the system.



The fundamental choice you have to make is what user registry to use. Will you use a SAF user registry such as RACF, or an LDAP user registry such as Tivoli Directory Server for z/OS? The choice is influenced by many factors but it is certainly important to know which one you will use before you attempt to configure WebSphere Process Server for z/OS.

Note that even if you decide to use RACF as the user registry for WebSphere Process Server for z/OS you will need to define an LDAP server for the users that access WebSphere Business Services Fabric for z/OS.

## Why use Tivoli Directory Server for z/OS?

LDAP servers are available on many platforms but because this Redbooks publication is about WebSphere Process Server for z/OS this section describes some of the reasons to consider running LDAP on z/OS.

There are two different LDAP Servers available on z/OS since z/OS 1.8.

- ▶ z/OS Integrated Security Services (ISS) LDAP
- ▶ IBM Tivoli Directory Server (TDS) for z/OS V6.0 LDAP

IBM Tivoli Directory Server (TDS) for z/OS is preferred because the ISS LDAP is functionally stabilized and its schema is less compatible with other LDAP servers.

**Note:** The TDS z/OS schema is compatible with TDS on other platforms. Products like Tivoli Access Manager, for example, can equally use TDS on z/OS as their user registry. A non-z/OS WebSphere Application Server can also use the TDS z/OS LDAP as its user registry.

Tivoli Directory Server for z/OS offers several advantages compared to a non-z/OS LDAP server that must be taken into account by z/OS customers when selecting an LDAP server. These advantages are discussed below.

- ▶ TDS for z/OS is shipped with the base z/OS operating system and is therefore available at no charge to z/OS customers.
- ▶ TDS for z/OS has the LDAP Native Authentication (LNA) feature. Native Authentication is where a user's password is held in RACF rather than on the userpassword attribute in the LDAP database. Using LNA can increase the security of powerful system administration user IDs such as those with access to the Integrated Solutions Console. Such users can be defined as LNA user IDs where their passwords are managed with RACF, while other normal users are held in the LDAP database. Note that passtickets can be used with LNA user IDs, also.

- ▶ TDS for z/OS has its database in DB2 z/OS, which can be shared across a sysplex:
  - You have free high availability (HA) because you start an LDAP server on each LPAR in the sysplex, and the load is balanced across the LDAP Servers by sysplex distributor. You tune the LDAP with properties that control number of concurrent connections, and so forth. You just need the processor/memory to support the load you need.
  - Disaster recovery for TDS z/OS is free because DB2 z/OS data is nearly always part of an existing, funded, tested, disaster recovery solution. Therefore, you avoid the cost of providing and testing disaster recovery of an LDAP server on some other platform.
- ▶ You can also configure the TDS z/OS in failover mode, rather than load sharing mode, just as on other platforms. You can even replicate between sysplex nodes.
- ▶ TDS z/OS uses the z/OS Unix file system, which is almost certainly more secure than any other Unix file system because the powerful privileges are controlled by RACF. This means that the TDS z/OS LDAP is more secure than one running on non-z/OS platforms.
- ▶ The passwords of users held in the LDAP database can be encrypted with the private key stored in the z/OS hardware. This means that there are no keystore files protected by passwords that can be hacked.
- ▶ The LDAP configuration file, which needs to be protected against unauthorized read/write, can be held in a z/OS partitioned dataset, which makes it even harder for someone to gain access to the LDAP configuration.
- ▶ TDS LDAP z/OS also provides an SDBM schema that maps directly on to the RACF database. This allows a security administrator to use a graphical LDAP client to manage the RACF database. This might be useful for some customers.
- ▶ TDS on z/OS can use the hardware crypto cards on the z machine to manage crypto processing related to SSL. This ought to give great scalability of logins.
- ▶ For scalability in general, the DB2 for z/OS subsystem that provides the database for TDS is certainly a scalable, available, database. Typically, the majority of frequently used LDAP pages would end up in memory in the coupling facility (that is, in DB2 global buffer pools). So access to LDAP data is at memory speeds. Also, as stated before, you can run an LDAP server on each LPAR of your sysplex. That gives great availability and scalability.
- ▶ The TDS for z/OS LDAP server has also a new internal cache management implementation, with inter-cache consistency in a sysplex.

## 2.2.7 Managing EJB roles using RACF GEJBROLE profiles

There are a number of roles associated with WebSphere Application Server for z/OS, WebSphere Process Server for z/OS, and WebSphere Business Services Fabric for z/OS. When using RACF to perform EJB™ role authorization checking there are two main ways to approach the management of the RACF profiles.

- ▶ The Resource-oriented approach starts by creating profiles to protect all the resources (in this case EJBROLE profiles), then considering the access list for each profile.
- ▶ The Organization-oriented approach starts by considering how many different types of users there are, then defining the minimum number of profiles required to provide the correct access.

We initially took the granular resource-oriented approach by defining an EJBROLE profile for each role and creating a number of different RACF groups that we permitted access to each role. A drawback of this approach is that it leads to a large number of EJBROLE profiles. The organization-oriented approach results in fewer RACF profiles and is easier to maintain.

When following the organization-oriented approach, a useful technique is to define RACF GEJBROLE profiles rather than EJBROLE profiles. When you define a profile in class EJBROLE, it refers to only one role. When you define a profile in class GEJBROLE, it can contain more than role. Roles are added to a GEJBROLE profile using the ADDMEM(<role>) option. For example, suppose that your organization consists of two different organizational groups:

- ▶ People responsible for administering WebSphere Process Server for z/OS
- ▶ Ordinary users that may need certain roles to use applications

You can create a GEJBROLE profile called WPSAdmins and add to it all the roles that provide administrator privileges. Then you can create a second GEJBROLE profile called WPSUsers and add to it all the roles that an ordinary user needs to run.

Having only two GEJBROLE profiles like this is at the opposite extreme to having one EJBROLE profile for each role. You will probably find that the correct solution for your organization lies between these two extremes. It is not possible to define only two GEJBROLE profiles for administrators and users, for example, because you will need to define at least two more profiles to support the JMSAPIUser and EscalationUser runAs roles. Nevertheless, we recommend the organization-oriented approach, however, because it will reduce 20 profiles to around five.

A possible solution using GEJBROLES is shown in Table 2-1 on page 42. Notice that administrators are in RACF group WPSADM, and that group is on the access list of all profiles.

Table 2-1 Possible solution using GEJBROLEs

Organizational role	GEJBROLE profile name	Roles added to the GEJBROLE profile	Groups on Access List
Administrators	WPSAdmins	BPESystemAdministrator, BPESystemMonitor, TaskSystemAdministrator, TaskSystemMonitor, BPCIVTUser, eventAdministrator, catalogAdministrator, WBIOperator, ObserverUser, Administrator	WPSADM
API Users	WPSAPIUsers	BPEAPIUser, TaskAPIUser, BRRestAPIUsers	WPSADM, WPSAPIU
Normal Users	WPSUsers	eventConsumer, eventCreator, eventUpdater, catalogReader, WebClientUser	WPSADM, WPSUSR
JMSAPIUser runAs role	JMSAPIUser	JMSAPIUser	none
Escalation runAs role	EscalationUser	EscalationUser	none

We created sample JOB GEJBROLE to implement the role assignments in the table above for our MD cell. An extract that shows the commands to define the WPSAdmins GEJBROLE profile is shown in Example 2-1 on page 43. Also shown is the GEJBROLE profile for role JMSAPIUser that assigns the runAs user ID of MDJMSAPI in the APPLDATA property.

The JMSAPIUser role is a runAs role whose purpose is to map the role to a user ID, which for our MD cell was MDJMSAPI. The MD.JMSAPIUser GEJBROLE profile needs no access list, and since the runAs user ID is assigned to the role on the APPLDATA option of the GEJBROLE profile, you cannot group it together with other roles in the same GEJBROLE profile.

*Example 2-1 Extract of sample job GEJBROLE*

```

/*   Define WPS Administrator roles   */
RDELETE GEJBROLE MD.WPSAdmins
RDEFINE GEJBROLE MD.WPSAdmins
    UACC(NONE)
    OWNER(WASCFG)
    DATA('MD WPS Administrator roles')
    ADDMEM(
        MD.BPESystemAdministrator,
        MD.BPESystemMonitor,
        MD.TaskSystemAdministrator,
        MD.TaskSystemMonitor,
        MD.BPCIVTUser,
        MD.eventAdministrator,
        MD.catalogAdministrator,
        MD.WBIOperator,
        MD.ObserverUser,
        MD.Administrator,
    )
PERMIT MD.WPSAdmins
    CLASS(GEJBROLE)
    RESET
PERMIT MD.WPSAdmins
    CLASS(GEJBROLE)
    ID(MDWPSADM)
    ACCESS(READ)

/*   Define WPS API User roles   */
RDELETE GEJBROLE MD.WPSAPIUsers
RDEFINE GEJBROLE MD.WPSAPIUsers
    UACC(NONE)
    OWNER(WASCFG)
    DATA('MD WPS API User roles')
    ADDMEM(
        MD.BPEAPIUser,
        MD.TaskAPIUser,
        MD.BRRestAPIUsers,
    )
PERMIT MD.WPSAPIUsers
    CLASS(GEJBROLE)
    RESET
PERMIT MD.WPSAPIUsers
    CLASS(GEJBROLE)
    ID(MDWPSAPI,MDWPSADM)
    ACCESS(READ)

/*   Define WPS User roles   */
RDELETE GEJBROLE MD.WPSUsers

```

```

RDEFINE GEJBROLE MD.WPSUsers                                +
    UACC(NONE)                                              +
    OWNER(WASCFG)                                           +
    DATA('MD WPS User roles')                             +
    ADDMEM(                                                 +
        MD.eventConsumer,                                  +
        MD.eventCreator,                                  +
        MD.eventUpdater,                                  +
        MD.catalogReader,                                  +
        MD.WebClientUser,                                  +
        MD.BusinessRuleUser,                              +
    )                                                         +
PERMIT MD.WPSUsers                                          +
    CLASS(GEJBROLE)                                         +
    RESET                                                    +
PERMIT MD.WPSUsers                                          +
    CLASS(GEJBROLE)                                         +
    ID(MDWPSAPI,MDWPSADM,MDWPSUSR)                          +
    ACCESS(READ)                                             +
/* These assign RunAs userids to MDBs via APPLDATA */
RDELETE GEJBROLE MD.JMSAPIUser
RDEFINE GEJBROLE MD.JMSAPIUser                                +
    UACC(NONE)                                              +
    OWNER(WASCFG)                                           +
    APPLDATA('MDJMSAPI')                                    +
    DATA('MDCELL JMSAPIUser runAs')                       +
    ADDMEM(                                                 +
        MD.JMSAPIUser,                                    +
    )                                                         +
PERMIT MD.JMSAPIUser                                        +
    CLASS(GEJBROLE)                                         +
    RESET                                                    +
RDELETE GEJBROLE MD.EscalationUser
RDEFINE GEJBROLE MD.EscalationUser                            +
    UACC(NONE)                                              +
    OWNER(WASCFG)                                           +
    APPLDATA('MDESCAL')                                    +
    DATA('MDCELL EscalationUser runAs')                   +
    ADDMEM(                                                 +
        MD.EscalationUser                                +
    )                                                         +
PERMIT MD.EscalationUser                                    +
    CLASS(GEJBROLE)                                         +
    RESET                                                    +
SETR RACLIST(EJBROLE) REFRESH

```

---

**Note:** Once you have decided to use GEJBROLE profiles, define all EJB role profiles in that class. Do not mix EJBROLE and GEJBROLE profiles. Although a GEJBROLE profile containing only one role is the same as a single EJBROLE profile, it is best to define the GEJBROLE profile with one member. RACF considers profiles in both classes and merges the access lists of any roles that appear in more than one profile. This can cause confusion.

## 2.2.8 Using intermediate symbolic links and ICFCAT aliases

When you configure a node, you need to specify the paths to the WebSphere Application Server for z/OS and WebSphere Process Server for z/OS product mount points. We suggest that you create intermediate symbolic links that indirectly reference the physical mount points of the products rather than referring directly to the mount points.

WebSphere Application Server for z/OS also has some z/OS load libraries and we recommend that you indirectly reference these libraries through the use of ICFCAT aliases rather than coding the real dataset names in JCL.

### Why use intermediate symbolic links?

Intermediate symbolic links allow you to easily switch a node from one service level to another by deleting and redefining the intermediate symbolic links.

**Note:** For a full discussion on this best practice, refer to White Paper WP100396, *Test, Production and Maintenance: The essential concepts needed to know how to configure isolation between environments*, available from the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100396>

We recommend that you use intermediate symbolic links in order to have the flexibility to upgrade one node at a time. By using intermediate symbolic links, the upgrade granularity is at the level of a node.

If you do not use intermediate symbolic links, an upgrade to a new service level of WebSphere Process Server for z/OS will affect all the nodes that refer to the product's mount point. You will have to stop all the nodes on the LPAR referring to the product ZFS, unmount it, mount the new service level, and then restart all the nodes on the LPAR. In other words, when you do not use an intermediate symbolic link the upgrade granularity is that of the LPAR.

We explain how we created intermediate symbolic links for the nodes in our MD cell in 2.3.1, "Create intermediate symbolic links to products" on page 60.

## Why use ICFCAT aliases?

WebSphere Application Server for z/OS has the following z/OS load libraries:

- ▶ SBBOLPA
- ▶ SBBGLOAD
- ▶ SBBOLoad
- ▶ SBBOLOD2

You can choose to load the contents of these libraries into the Link Pack Area (LPA) so all WebSphere address spaces share the same copy of the programs. Using LPAR will reduce virtual storage requirements and improve program load performance, but it means that all the cells that load the WebSphere Application Server for z/OS modules from LPA will need to be upgraded at the same time a new service level is loaded into the LPA. In other words, a service upgrade will affect multiple cells, possibly all the cells in an LPAR.

In order to have the flexibility to run nodes at different service levels, you must configure each node so it references the SBBO\* libraries through STEPLIB.

When using STEPLIB, we recommend that you create a set of ICFCAT aliases for each node that refers to the real dataset names of the SBBO\* libraries. By doing this, you will avoid the need to make any manual changes to the STEPLIB in all the started task JCL, and to the export STEPLIB statement in the setupCmdLine.sh, when it is time to upgrade a node to a new service level.

You need only to delete and redefine the ICFCAT alias to switch a node from the set of SBBO\* libraries at the old service level to the set of SBBO\* libraries at the new service level. We explain how we created a set of ICFCAT aliases for each of the nodes in our MD cell in Section 2.3.2, “Create ICFCAT aliases to SBBO\* libraries” on page 61.

When configuring the WebSphere Application Server for z/OS nodes, always use the ICFCAT aliases names rather than the real dataset names.

## 2.2.9 Planning DB2 names

Plan DB2 names together with your DB2 system administrator. The spreadsheet we describe in Section 2.2.5, “Planning using a spreadsheet” on page 33 includes a worksheet that helps you choose names for the databases you will need.

A key question you must answer is “Do I put all the WebSphere Process Server for z/OS components in one database or do I create a separate database for each component?” We suggest that you plan to create separate databases for each component.



The WebSphere Process Server for z/OS Information Center also recommends using separate databases and states that this is for performance reasons. That is good advice for DB2 on non-z/OS platforms. However, when using DB2 for z/OS, choosing to define multiple databases only has an indirect effect on performance.

Using multiple databases has an indirect affect on performance when using DB2 for z/OS because you have the opportunity to specify the BUFFERPOOL and INDEXBP options on the CREATE DATABASE statements. Much of the DDL provided with WebSphere Process Server for z/OS does not specify BUFFERPOOL or INDEXBP. The bufferpools are set according to the default that applies to the database, table, or index in which it is created.

If you have one database the majority of tables and indexes will end up in the same buffer pools. If this is what you want, you might choose to use only one database. However, we suggest that you create separate databases and assign different BUFFERPOOL and INDEXBP values to each component, at least initially. After you have some experience with WebSphere Process Server for z/OS, you can alter the buffer pool assignments using the DB2 ALTER TABLESPACE or ALTER INDEX commands.

Apart from this indirect performance advantage, we recommend multiple databases simply because it is easier to manage in DB2 when the tables and indexes of each component are separated in their own database. It is possible that you will find a need to delegate some DB2 administration privileges to individuals in your organization. This is much easier to do if you can simply GRANT DBADM to such people.

The planning spreadsheet assigns different database names to each component by default. If you want to define only one database then set the same database name for each component on the Variables worksheet.

Although the createDB.sh shell script that is described in Section 2.6.2, “Customize the sample createDB.sh script” on page 81 creates objects in one database, you can simply run createDB.sh once for each WebSphere Process Server for z/OS component, assigning a different database name on each invocation. This is described in Section 2.6.3, “Execute createDB.sh to generate the DDL” on page 83.

## 2.2.10 Choosing currentSchema or currentSQLID

An understanding of schema and how it relates to sqlid is essential to configure WebSphere Process Server for z/OS successfully. This section contains background information about schemas in DB2 for z/OS.

### What is the schema?

The currentSchema and currentSQLID properties can be set on the custom properties of a data source. Both properties affect the qualifier used when an application issues unqualified SQL.

What is qualifier and what is unqualified SQL? When an application issues an SQL statement like `SELECT * FROM MYTABLE`, this is unqualified SQL because the table name is not qualified like `<qualifier>.MYTABLE`.

There can be multiple instances of a table called MYTABLE in DB2 for z/OS and a qualifier is used to distinguish one instance of MYTABLE from another. The default qualifier is the authorization ID (usually the user ID) executing the unqualified SQL statement, so if the user ID executing the SQL statement is FRED, then `SELECT * FROM MYTABLE` is equivalent to `SELECT * FROM FRED.MYTABLE`.

The term qualifier is now more commonly known as the schema so from now on in this Redbooks publication we will talk about schema rather than qualifier.

### WebSphere Process Server for z/OS implications

When an SQL request is made by an application running in WebSphere Process Server for z/OS there are several ways of specifying the user ID that the request runs under. Therefore it is possible to influence the schema (qualifier) by setting the appropriate user ID when making the request. Alternatively, it is possible to execute the SQL request under any user ID but specify the schema you want to use when making unqualified requests.

Another possibility is that the application discovers the schema it needs to use at runtime (perhaps from an xml file) and then issues fully-qualified SQL requests rather than unqualified SQL requests.

When an application issues fully-qualified SQL there is no issue to resolve. The question of whether you need to use currentSchema or currentSQLID arises only when applications make unqualified SQL requests.

## Unqualified SQL from WebSphere Process Server for z/OS

Currently, several components of WebSphere Process Server for z/OS make unqualified SQL requests.

This means that if you do not code either `currentSchema` or `currentSQLID` on the data source Custom properties, the SQL request will inherit a schema equal to the data source's J2C authentication alias user ID.

For example, if you have configured a data source with a J2C authentication alias for user ID `MDDBU`, then a statement like `SELECT * FROM MYTABLE` will execute as `SELECT * FROM MDDBU.MYTABLE` in DB2.

### Using the J2C authentication alias as the schema

One option is to define all the WebSphere Process Server for z/OS tables with a schema equal to the J2C authentication alias user ID. If you choose this option you do not need to code either `currentSchema` or `currentSQLID`.

Because this seems to be the easy option, why not do it? The reason not to use the J2C authentication alias as the schema is that DB2 for z/OS gives certain implicit privileges to the schema and these give more power to the WebSphere application than a database administrator will normally allow. The implicit privileges include `DROP`, for example. Normally, `DROP` is not something that an application is allowed to do.

For this reason, the DB2 for z/OS database administrator may refuse to allow tables to be created with a schema equal to the user ID that accesses those tables. Many DB2 for z/OS database administrators have a rule that no table can be created with a qualifier equal to any user ID. They use a security policy that says that all schemas must be set to RACF groups, rather than user IDs.

For these reasons you will probably not want the user ID executing the SQL request to be equal to the schema. This means that tables have to be created with some other schema name, and there needs to be a way to specify the schema that is used when an application makes an unqualified SQL request.

### Specifying the schema with `currentSchema`

In the custom properties of a data source, there is a property called `currentSchema`. If the application makes an unqualified SQL request, and you specify a value in `currentSchema`, the SQL request is qualified by the value of `currentSchema`.

For example, if all the WebSphere Process Server for z/OS tables have been created with a schema of `MDWPS`, specifying `currentSchema=MDWPS` will allow all unqualified SQL statements from WebSphere Process Server for z/OS to operate on the set of tables that are qualified with `MDWPS`.

The authorization ID in DB2 used for unqualified SQL requests is the J2C authentication alias user ID. Therefore, an implication of using currentSchema is that the J2C authentication alias user ID must be granted the privileges it needs to work with the tables, indexes, sequences, and functions in DB2. For example, if the J2C authentication alias is MDDBU and the schema is MDWPS, in order for MDDBU to issue `SELECT * FROM MYTABLE` it is necessary to issue the following command:

```
GRANT SELECT ON MDWPS.MYTABLE TO MDDBU
```

## **Specifying the schema with currentSQLID**

In the custom properties of a datasource, there is a property called currentSQLID. This property causes the user ID under which an SQL statement executes to switch to the user ID named in the currentSQLID property.

For example, on a data source that has a J2C authentication alias for user ID MDDBU, SQL requests will execute under user ID MDDBU. However, coding currentSQLID=MDWPS is the same as issuing the statement `SET CURRENT SQLID='MDWPS'` before executing the application's SQL statement.

Therefore, if currentSQLID=MDWPS, an application issuing the SQL command `SELECT * FROM MYTABLE` will switch their current SQLID special register to MDWPS and then execute `SELECT * FROM MYTABLE`. Because the schema will default to the SQLID, this has the same affect as issuing the following command:

```
SELECT * FROM MDWPS.MYTABLE
```

## ***Why not use currentSQLID then?***

You can read in the Information Center that currentSQLID can be used to specify the schema and you will notice that WebSphere Process Server for z/OS adds the schema to the currentSQLID property on its datasource called WBI\_Datasource. So what is wrong with using currentSQLID?

The use of currentSQLID is something you might do in a non-z/OS environment, which is why it appears in documentation. In a non-z/OS environment, the DB2 database is almost always remote with respect to the WebSphere Process Server and is accessed across the network. In that situation you may want to use a different currentSQLID to connect to different WebSphere Process Server databases.

The problem with using currentSQLID with DB2 for z/OS is that switching from one authorization ID to another is something that has to be authorized. Using currentSQLID is a bit like issuing an `su` (switch user) in UNIX. On a z/OS system, switching from one user ID to another is not allowed unless you have been granted the privilege to do so.

If you set the currentSQLID on a data source, the J2C authentication alias user ID will need to be granted the ability to switch to the schema you want to set on currentSQLID.

The Information Management Software for z/OS Solutions Information Center lists the authorization required in order to issue SET CURRENT SQLID. See the following Web page for this information:

<http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp>

Here is an quote from the Web page:

*If any of the authorization IDs of the process has SYSADM authority, CURRENT SQLID can be set to any value. Otherwise, the specified value must be equal to one of the authorization IDs of the application process. This rule always applies, even when SET CURRENT SQLID is a static statement.*

Therefore, a user ID with SYSADM authority can accomplish this, but you do not want to grant DB2 SYSADM authority to the J2C authentication alias user ID.

The phrase “*equal to one of the authorization IDs of the application process*” in the quote above means that the user ID can switch to an SQLID that is equal to one of the RACF groups to which it is connected. For example, if the J2C authentication alias is connected to a RACF group whose name is equal to the schema, it can switch SQLID to the schema.

For secondary authorization groups to be considered, DB2 for z/OS must be using RACF to control authorization to the objects it manages, as opposed to using DB2’s own security controls based on GRANTS. For example, if all the following factors are true, the SQLID can be switched to the schema MDWPS:

- ▶ DB2 takes into account connections to RACF groups (secondary authorization IDs) to manage authorization checking of DB2 resources.
- ▶ The WebSphere Process Server for z/OS tables have been created with a schema of MDWPS.
- ▶ A RACF group called MDWPS has been created.
- ▶ The J2C authentication alias is connected to RACF group MDWPS.
- ▶ The data source property currentSQLID is set to MDWPS.

One thing to notice about doing this, however, is that connecting the J2C authentication alias user ID to the schema group is the same as giving it full ownership privileges over the tables under the schema. This is what we want to avoid. When the objective is to arrange things so that no user ID in the WebSphere cell has more than the minimum level of access to DB2 resources, it cannot be done by using currentSQLID.

### ***What if DB2 does not use RACF for authorization checking?***

If DB2 is not using RACF to perform authorization checking, but is using its own internal security management system, then DB2 does not take into consideration the RACF groups to which a user is connected when it makes authorization decisions. Therefore, a user ID can only be authorized to switch the SQLID by being granted DB2 SYSADM authority.

### **Schema recommendation**

Using the currentSQLID property is indirectly allowing WebSphere Process Server for z/OS to use the correct schema and brings additional security issues that are unlikely to be acceptable to a security-conscious DB2 system administrator. When using the currentSQLID property, the J2C authentication alias user ID ends up with more privileges than it needs, including DROP. We suggest that you use currentSchema rather than currentSQLID.

**Changes in handling currentSchema from V8.1 to V9.1:** The way DB2 for z/OS handles currentSchema changed from DB2 for z/OS V8.1 to V9.1. In DB2 for z/OS V8.1, it was not possible to set the currentSchema to a value different from the current SQLID without granting the current SQLID schema privileges. In other words, the SQLID (which would be the database J2C authentication alias user ID) would have to be connected to the schema group. In DB2 for z/OS V9.1, however, this restriction no longer applies and it is possible to use currentSchema without granting extra privileges to either the servant or J2C authentication alias user ID.

**Note:** Although we suggest not using currentSQLID property on a data source, there are times when you will need to use the SET CURRENT SQLID command when executing the DDL to define all the WebSphere Process Server for z/OS tables in DB2 for z/OS. You will need to issue the DB2 command **SET CURRENT SQLID='<schema>'**; before executing the **CREATE TABLE** statements because some of the DDL is unqualified. This is explained in “Unqualified DDL considerations” on page 53.

Your choice of currentSQLID or currentSchema also affects the GRANT commands you will need to use. That is discussed in Section 2.2.11, “Implications for GRANT statements in DB2” on page 54.

## Unqualified DDL considerations

Some of the DDL statements generated by WebSphere Process Server for z/OS that create all the tables, indexes and views in DB2 are unqualified. For example, there might be a statement like **CREATE TABLE MYTABLE IN ...** When a CREATE statement is unqualified, it is assigned a schema (qualifier) equal to the authorization ID executing the command.

When you run the DDL to define all the WebSphere Process Server for z/OS tables it is important to execute the CREATE statements under an authorization ID equal to the correct schema. For example, when the DDL is unqualified, to create table MYTABLE with a schema of MDWPS it is necessary to issue **SET CURRENT SQLID='MDWPS';** before executing the **CREATE TABLE MYTABLE** statement.

As discussed in “Specifying the schema with currentSQLID” on page 50, it is not possible to switch SQLID unless you are authorized. Someone with DB2 SYSADM authority can execute the DDL because a SYSADM can issue **SET CURRENT SQLID**. Or, if the user ID executing the DDL is connected into the schema’s RACF group, it is possible to switch the SQLID to that of the schema.

We suggest that you ask your DB2 system administrator to execute the DDL. Ensure that you have inserted the correct **SET CURRENT SQLID** statements before the CREATE statements. Otherwise, all the WebSphere Process Server for z/OS tables are created with a schema equal to the authorization ID of the DB2 SYSADM.

In order to draw attention to this, we structured our JCL that executed DDL so that the SYSIN contained **SET CURRENT SQLID** as the first statement. The DDL statements to be executed were then concatenated on SYSIN below this. An extract of the job that executed our BPC DDL is shown in Example 2-2 to illustrate this concept.

*Example 2-2 The use of SET CURRENT SQLID in JCL that executes DDL*

---

```
. .  
//SYSTSIN DD *  
  DSN SYSTEM(D9FG)  
  RUN PROGRAM(DSNTEP2) PLAN(DSNTEP91) LIB('DB9FU.RUNLIB.LOAD')  
//SYSIN DD *  
  SET CURRENT SQLID = 'MDWPS' ;  
//      DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDBPCTB)  
//      DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDOBSTS)  
//      DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDOBSTB)
```

---

## 2.2.11 Implications for GRANT statements in DB2

The GRANT statements that you will need to issue differ depending on whether you use `currentSQLID` or `currentSchema`. We did not have time to explore all the possibilities, and your particular requirements may differ, but this section will help you plan what you need to do.

### When using `currentSQLID`

When using `currentSQLID=<schema>` the user ID executing the SQL is the same person as the schema and so has all implicit privileges over that schema. Therefore, there is no need to issue additional grants when using `currentSQLID` set to the schema. Nevertheless we issued the GRANTs in Example 2-3. We created sample job MDCREDB to create the databases and issue the GRANTs.

#### *Example 2-3 GRANT statements issued when using `currentSQLID`*

---

```
GRANT SELECT ON SYSIBM.SYSTABLES TO MDDBU;

GRANT DBADM ON DATABASE MDWPSDB TO MDWPS;
GRANT DBADM ON DATABASE MDBPCDB TO MDWPS;
GRANT DBADM ON DATABASE MDOBSDB TO MDWPS;
GRANT DBADM ON DATABASE MDESDB TO MDWPS;
GRANT DBADM ON DATABASE MDEVTDDB TO MDWPS;
GRANT CREATEIN, DROPIN ON SCHEMA MDWPS TO MDWPS;
GRANT DBADM ON DATABASE MD01ADB TO MD01A;
GRANT DBADM ON DATABASE MD01BDB TO MD01B;
GRANT DBADM ON DATABASE MD01CDB TO MD01C;
GRANT DBADM ON DATABASE MD01SDB TO MD01S;
GRANT CREATEIN, DROPIN ON SCHEMA MD01A TO MD01A;
GRANT CREATEIN, DROPIN ON SCHEMA MD01B TO MD01B;
GRANT CREATEIN, DROPIN ON SCHEMA MD01C TO MD01C;
GRANT CREATEIN, DROPIN ON SCHEMA MD01S TO MD01S;
GRANT USE OF STOGROUP MDWPSSG TO MDWPS;
GRANT USE OF STOGROUP MDWPSSG TO MD01A;
GRANT USE OF STOGROUP MDWPSSG TO MD01B;
GRANT USE OF STOGROUP MDWPSSG TO MD01C;
GRANT USE OF STOGROUP MDWPSSG TO MD01S;
< not shown are GRANT USE OF BUFFERPOOL statements >
```

---

Our DB2 for z/OS system was using RACF to perform authorization checking. We decided to grant DBADM to the schema MDWPS. A RACF group called MDWPS was created, which allowed us to delegate schema (owner) privileges by connecting user IDs into the schema group. We did not connect the J2C authentication alias MDDBU to that group. There was no need when using `currentSchema`.

Notice in Example 2-3 that each bus has a different schema with a name like MD01A. The example does not show the many GRANT USE OF BUFFERPOOL statements. Because some of the DDL that creates tables is also unqualified, it is



important to issue `SET CURRENT SQLID='<schema>'` before executing the DDL. It is necessary to issue `GRANT USE OF BUFFERPOOL` to the schema groups.

### When using currentSchema

When using `currentSchema`, SQL statements execute under the identity of the J2C authentication alias user ID. That user ID has no implicit privileges over the tables under the schema, making it necessary to issue `GRANT` statements to allow it to `SELECT`, `UPDATE`, `INSERT`, and `DELETE` rows in the tables.

The `sibDDLGenerator.sh` creates `GRANT` statements when it generates DDL for the SIBs, but it is necessary to construct `GRANT` commands based on the `CREATE` statements that were issued for the other WebSphere Process Server for z/OS components. We did this by copying the generated DDL into a series of `GRANTxxx` members in our `WASCFG.MDCELL.WPS.SQL` library and then editing it to convert the `CREATE` statements into `GRANT` commands. Then we created sample JCL `MDGRANT` to execute the `GRANT` statements. See Appendix A, “Additional material” on page 351 for details on how to obtain these `GRANT` statements and the `MDGRANT` job.

Because we did not know exactly what kinds of accesses were being performed by WebSphere Process Server for z/OS we issued **`GRANT ALL PRIVILEGES ON`** commands for each table.

## 2.2.12 Plan and create DB2 buffer pools

When planning buffer pools, the most important thing to do is to avoid the use of the `BP0`, `BP8K0` and `BP16K0` buffer pools that are used by DB2 itself. Unfortunately the generated DDL often defaults to these values, so you must review all DDL before executing it and set the correct bufferpools.

Work with your DB2 database administrator to plan the buffer pools you will use. We used the spreadsheet described in Section 2.2.5, “Planning using a spreadsheet” on page 33 to plan all the names and buffer pools we used in DB2. The table on the Variables worksheet that plans database names and bufferpools does not allow you to choose `BP0`, `BP8K0` or `BP16K0`. For each bufferpool we issued **`-ALTER BUFFERPOOL(BPxxx) VPSIZE(yyyy)`** to define a buffer pool `BPxx` with `yyyy` pages.

Because we were configuring a cluster, which in turn required DB2 in data-sharing mode, any new buffer pools we added also required cache structures to be defined in the Coupling Facility for Global Buffer pools.

For instruction on how to do this, refer to the DB2 for z/OS Information Center article *Defining coupling facility structures* available at the following Web page:

[http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db29.doc.dshare/db2z\\_definingcfstructures.htm](http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db29.doc.dshare/db2z_definingcfstructures.htm)

**Buffer pool assignments:** The buffer pool assignments we used are not recommended values based on the findings of any performance study.

We assigned different buffer pools to the different components of WebSphere Process Server for z/OS to illustrate the importance of planning and defining buffer pools, and the importance of avoiding the use of BP0, BP8K0, and BP16K0.

After configuring WebSphere Process Server, the DB2 system administrator can monitor DB2 performance and change buffer pool assignments if necessary. Therefore, it is not critical that you get this exactly right from the beginning. Just be sure not to use BP0, BP8K0, or BP16K0.

### 2.2.13 Decide how you will create and execute the DDL

Defining a connection between WebSphere Process Server for z/OS and DB2 for z/OS involves definitions in both WebSphere Process Server for z/OS and in DB2. Because various names must match in these two places, WebSphere Process Server for z/OS provides shell scripts and Integrated Solutions Console wizards that take various names as input and ensure that the definitions in WebSphere Process Server for z/OS match those in DB2 for z/OS.

Although the shell scripts and wizards hide the details of all the definitions that are required, an implication is that a user ID is required that has sufficient privileges both to perform administration functions in WebSphere Process Server for z/OS and database administration functions in DB2 for z/OS. If such a user ID does not exist in your organization, it is necessary to avoid creating tables in DB2 for z/OS at the same time as defining the DB2 JDBC™ provider and data sources in WebSphere Process Server for z/OS.

In order to avoid granting both WebSphere administration authority and DB2 SYSADM authority to the same user ID, you can ask your DB2 system administrator to create the databases and stogroups and then issue GRANT statements to delegate authority over the databases, stogroups, and buffer pools to the user ID that is executing the DDL. There are two issues, then, to consider:

- ▶ Generation of the DDL
- ▶ Execution of the DDL

WebSphere Process Server offers several alternative methods for generating and executing the DDL statements that create the necessary tables and indexes in DB2. One of the early decisions you must make is the method you will use to generate the DDL, and the method you will use to execute it.

## Generating the DDL

The options for generating the required DDL are as follows:

- ▶ When you configure WebSphere Process Server for z/OS into a stand-alone server, all the required DDL is generated when you run the `zWPSConfig.sh` shell to augment the stand-alone server.

When you configure WebSphere Process Server for z/OS into a Network Deployment cell, only the DDL for the common database is generated when you run the `zWPSConfig.sh` to augment the deployment manager. You must generate DDL for other components of WebSphere Process Server for z/OS using one of the other methods.

- ▶ DDL is generated for components like BPC and CEI when you run the wizards that configure these components.
- ▶ The `sibDDLGenerator.sh` generates DDL for a Service Integration Bus
- ▶ There is a `createDB.sh` that can both generate all the DDL and execute it.

If you start by configuring WebSphere Process Server for z/OS into a stand-alone server, all the components of WebSphere Process Server for z/OS are resident in the same server and will use one database in DB2 by default. You can specify separate databases if you edit the response file before running `zWPSConfig.sh`

We wanted to have the flexibility to configure BPC in a different cluster from the Support and Messaging components of WebSphere Process Server for z/OS and to define different databases for each component. We configured a Network Deployment cell comprising a deployment manager and two empty nodes, then we generated DDL using `createDB.sh`.

When you have generated the required DDL and edited it to match your requirements there are several ways to execute it:

- ▶ Create tables when executing the various wizards in the Integrated Solutions Console by checking the box Create Tables.

We do not recommend creating tables when you run the wizards because you will likely want to inspect the DDL before it is executed. There are also likely to be security issues executing DDL in this way.

- ▶ Use `DBUtility.sh`

See explanation for “Use `createDB.sh`.”

- Use createDB.sh

The createDB.sh wraps the DButility.sh and is easier to use than DButility.sh. Therefore if you wish to execute the DDL from the UNIX environment rather than using DSNTEP2, we suggest that you use createDB.sh.

- Use SPUFI

We do not recommend using SPUFI because of the risk of problems if the code page used by your terminal is not correct.

- Use DSNTEP2

See explanation for “Use DB2 Connect™.”

- Use DB2 Connect™

Because the DDL is generated on z/OS, the use of DB2 Connect would require the DDL to be downloaded to a workstation first. For a DB2 for z/OS system it is easier to use DSNTEP2 or createDB.sh.

The two most attractive options for executing DDL are to use createDB.sh and to use DSNTEP2.

**Note:** You do not have to use createDB.sh to execute the DDL if you use it to generate the DDL. You may use createDB.sh to generate the DDL and use DSNTEP2 to execute it.

## **At what stage in the configuration can you execute the DDL?**

When configuring WebSphere Process Server for z/OS into a Network Deployment cell, much of the DDL is generated when you run wizards or shell scripts to configure each component (such as SCA, CEI, or BPC). The DDL is not available until you have completed that stage of the configuration. In contrast, you can run createDB.sh just after augmenting the deployment manager and execute all the DDL at that point.

We found that it was much less effort to use createDB.sh to generate all the DDL in one instance. Also, when all the DDL already exists before you run the configuration wizards, there are a lot less errors in the logs to worry about. We strongly suggest using createDB.sh to generate the DDL, and then execute it using either createDB.sh or DSNTEP2 before you start to configure SCA, CEI, and BPC.

In this Redbooks publication we describe the use of createDB.sh to generate all the DDL just after augmenting the deployment manager. We also describe executing the DDL at that time. However if you prefer not to use createDB.sh we also describe how to use the DDL generated by the various wizards that configure components such as SCA, CEI, or BPC.

## 2.2.14 WebSphere Application Server for z/OS 64-bit mode

If you are able to run your workload within the heap available to a 31-bit JVM, do so, because running in 64-bit mode incurs a cost. It is advised to add more servants before going to 64-bit mode. However, when your heap is so stressed that you suffer significant garbage collection overhead, you may achieve better throughput running in 64-bit mode.

If you plan to run a single-cluster configuration you are more likely to need to run the WebSphere Process Server for z/OS cluster members in 64-bit mode. However, do not switch to 64-bit mode until you are sure that limited heap is limiting performance. Make sure you have enough real storage to support the sum of all the heaps, otherwise you will suffer heavy paging, which will make your performance far worse. Also, think about the impact the large heap will have on such things as disk space to support SVC dumps should a servant abend.

Enabling 64bit mode is fully described in the IBM White Paper *WebSphere for z/OS V6.1 - 64-bit Addressing Support*, WP100920, available from the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100920>

## 2.3 Prepare a WebSphere cell, RACF, and DB2

Make sure you have read Section 2.2, “Plan the Network Deployment configuration” on page 29. It is important to have planned all RACF and DB2 names before you start.

All the sample JCL and shell scripts referred to in this chapter can be downloaded from the additional material that accompanies this Redbooks publication. See Appendix A, “Additional material” on page 351. The spreadsheet that we used for planning the cell is in the `/zos_planning_spread` sheet directory. The JCL we used when configuring WebSphere Process Server for z/OS is in the `/zos_wps` directory.

It is important to be clear about your naming convention, your choice of schema names, and the RACF user IDs and groups that you will need. This is discussed in the following sections:

- ▶ Section 2.2.6, “Planning security” on page 38
- ▶ Section 2.2.9, “Planning DB2 names” on page 46
- ▶ Section 2.2.10, “Choosing currentSchema or currentSQLID” on page 48

The disk space administrator, RACF administrator, DB2 system administrator, and the network systems programmer should have contributed to your planning. After planning your cell (Section 2.2.5, “Planning using a spreadsheet” on page 33), the major tasks to prepare a WebSphere Process Server for z/OS cell are as follows:

1. Create intermediate symbolic links to products.  
See page 60.
2. Create ICFCAT aliases to SBBO\* libraries.  
See page 61.
3. Create RACF user IDs, groups, and EJBROLE profiles.  
See page 62.
4. Create a DB2JccConfiguration.properties file.  
See page 65.
5. Customize a .profile.  
See page 68.
6. Verify Java and the DB2 Universal JDBC Driver.  
See page 69.

### 2.3.1 Create intermediate symbolic links to products

We decided to always use an intermediate symbolic link to indirectly reference the mount points of the WebSphere Application Server for z/OS and WebSphere Process Server for z/OS products. The advantages of doing this are explained in Section 2.2.8, “Using intermediate symbolic links and ICFCAT aliases” on page 45.

When configuring the deployment manager’s node (mddmnode) we created sample JCL MDDMSYML to do this, as shown in Example 2-4.

*Example 2-4 Job MDDMSYML to create intermediate symbolic links for mddmnode*

---

```
//MDDMSYML JOB (0), 'MDADMIN', CLASS=A, MSGCLASS=A, NOTIFY=&SYSUID
/*JOBPARM SYSAFF=SC42
/*
/* Create symbolic links for WebSphere
/* Set correct ownership:group
/*
//WASSYML EXEC PGM=IKJEFT01, REGION=1024K
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
```

```

BPXBATCH SH +
ln -s /usr/lpp/zWebSphere/V6R1 +
                               /wasv6lconfig/mdcell/mddmnode_was_smpe; +
chown -Rh MDADMIN:MDCFG /wasv6lconfig/mdcell/mddmnode_was_smpe;
/*
//WPSSYML EXEC PGM=IKJEFT01,REGION=1024K
/*
/* Create symbolic links for Process Server
/* Set correct ownership:group
/*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
ln -s /usr/lpp/zWPS/V6R1M2 /wasv6lconfig/mdcell/mddmnode_wps_smpe; +
chown -Rh MDADMIN:MDCFG /wasv6lconfig/mdcell/mddmnode_wps_smpe ;
/*

```

---

Another sample job, MDNASYML, defined intermediate symbolics links for our primary empty node, mdnodea. Sample job MDNBSYML did the same for our secondary empty node, mdnodeb

## 2.3.2 Create ICFCAT aliases to SBBO\* libraries

Create a set of ICFCAT aliases for each node in your WebSphere Process Server for z/OS cell, including the deployment manager's node. We created sample job MDDMALIA (Example 2-5) to create a set of ICFCAT aliases for mddmnode that map to a set of physical SBBO\* datasets called BBO6142.SBBO\*. Notice that the second qualifier in the alias name matches the node name. Create a set of aliases for each node.

*Example 2-5 Sample JCL to define a set of ICFCAT aliases for mddmnode*

---

```

//MDDMALIA JOB (0),'MDADMIN',CLASS=A,MSGCLASS=H,NOTIFY=&SYSUID
/*JOBPARM SYSAFF=SC42
/*
/* Define a set of aliases for the deployment manager's node
/*
//S1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DEF ALIAS (NAME('BBO6142.MDDMNODE.SBBOEXEC') +
           RELATE('BBO6142.V6R1M2.SBBOEXEC'))
DEF ALIAS (NAME('BBO6142.MDDMNODE.SBBGLOAD') +
           RELATE('BBO6142.V6R1M2.SBBGLOAD'))

```

```

DEF ALIAS (NAME('BB06142.MDDMNODE.SBBOLOAD')      +
           RELATE('BB06142.V6R1M2.SBBOLOAD'))
DEF ALIAS (NAME('BB06142.MDDMNODE.SBBOLD2')        +
           RELATE('BB06142.V6R1M2.SBBOLD2'))
DEF ALIAS (NAME('BB06142.MDDMNODE.SBBOLPA')        +
           RELATE('BB06142.V6R1M2.SBBOLPA'))
DEF ALIAS (NAME('BB06142.MDDMNODE.SBBOMSG')        +
           RELATE('BB06142.V6R1M2.SBBOMSG'))
/*

```

---

We ran job MDDMALIA for the deployment manager node, then ran similar jobs for each empty node in our cluster, as follows:

- ▶ MDNAALIA for node A
- ▶ MDNBALIA for node B

When configuring the WebSphere Application Server for z/OS nodes, always use the ICFCAT aliases names rather than the real dataset names. When using the spreadsheet to plan the cell, if you choose to use ICFCAT aliases on the Variables worksheet, the alias names are used in the response files that are generated as input to the zPMT.

### 2.3.3 Create RACF user IDs, groups, and EJBROLE profiles

When configuring a WebSphere Application Server for z/OS V6.1 cell using the zProfile Management Tool (zPMT), jobs are generated that configure the RACF user IDs, groups, and profiles that are required by the WebSphere Application Server for z/OS cell. However, additional RACF user IDs, groups, and profiles are required for WebSphere Process Server for z/OS. In Section 2.2.5, “Planning using a spreadsheet” on page 33, we created a sample job called MDRACF that defined all the user IDs and groups we had planned for our cell.

We also created a job called MDEJBROL that defined RACF EJBROLE profiles to support the roles used by WebSphere Process Server for z/OS, including CEI, and Business Space. That job creates one profile per role. If you want to use GEJBROLE profiles to group roles together in fewer profiles, see job GEJBROLE.

It is a good idea to create all the required user IDs, groups, and EJBROLE profiles before you configure WebSphere Process Server for z/OS. The user IDs, groups, and EJBROLE profiles we created are summarized in Table 2-2 on page 63.



**Roles and groups in Table 2-2 and Table 2-3:** The roles and groups in these tables reflect a resource-oriented approach to defining RACF profiles, where there is a profile for each role and a RACF group provides access to that role.

Some roles have been defined as UACC(READ), meaning that anyone that has authenticated will have that role. You may not want to use UACC(READ) for the roles shown in the tables. We have done this for the API roles to show you that you can use UACC(READ).

An unauthenticated (guest) user ID cannot access roles that are UACC(READ).

We discuss an alternative approach called organization-oriented in Section 2.2.7, “Managing EJB roles using RACF GEJBROLE profiles” on page 41. You may wish to adopt an organization-oriented approach to simplify the RACF setup for EJB roles

*Table 2-2 WebSphere Process Server for z/OS user IDs, groups and Roles*

Group or User Description	RACF EJBROLES	Group or user ID
BPE API User Group	MD.BPEAPIUser	UACC(READ)
BPE Administrator's Group	MD.BPESystemAdministrator	MDBPEADG
BPE Admin User		MDBPEADM
BPE System Monitor's Group	MD.BPESystemMonitor	MDBPESMG
BPE SysMon User		MDBPESM
Human Task API User Group	MD.TaskAPIUser	UACC(READ)
Human Task Administrator's Group	MD.TaskSystemAdministrator	MDHTADMG
HT Admin User		MDHTADM
Human Task System Monitor's Group	MD.TaskSystemMonitor	MDHTSMG
HT SysMon User		MDHTSM
BPC Web Client Role Group	MD.WebClientUser	MDWCUG
BPC Web Client User		MDWCU
BPC Observer User Role Group	MD.ObserverUser	MDOBSUG
BPC Observer User		MDOBSU
BPC IVT User Role Group	MD.BPCIVTUser	MDWCUG

Group or User Description	RACF EJBROLES	Group or user ID
MDB RunAs Role User Group		MDMDBG
MDB JMSAPIUser Role	MD.JMSAPIUser	MDJMSAPI
MDB Escalation User	MD.EscalationUser	MDESCAL
CEI Event Administrator Group	MD.eventAdministrator	MDEVTADG
CEI Event Admin User		MDEVTADM
CEI Event Consumer Group	MD.eventConsumer	MDEVTCO
CEI Event Creator Group	MD.eventCreator	MDEVTCRG
CEI Event Updater Group	MD.eventUpdater	MDEVТУРG
CEI Catalog Administrator Group	MD.catalogAdministrator	MDCATADG
CEI Catalog Admin User		MDCATADM
CEI Catalog Reader Group	MD.catalogReader	MDCATRG
Failed Event Manager User Group	MD.WBIOperator	MDCFG, MDBPEADG, MDEVTADG
Business Space Administrator Group	MD.Administrator	MDBSPADG
Business Rules Manager user role Group	MD.BusinessRuleUser	MDBRMUG, MDBSPADG
Business Rules API role Group	MD.BRRestAPIUsers	MDBRAPIG, MDBSPADG

Table 2-3 summarizes the user IDs, groups and EJBROLE profiles we created for WebSphere Business Services Fabric for z/OS.

*Table 2-3 WebSphere Business Services Fabric for z/OS user IDs, groups and Roles*

Group or User Description	RACF EJBROLEs	Group / user ID
Fabric Administrator's Group	MD.FabricAdministrator	MDFADMG
Fabric Administrator user ID		MDFABADM
Fabric Governance Administration Group	MD.FabricGovernanceAdministrator	MDFGADMG
Fabric Governance Administration user ID		MDFGADM
Fabric Performance User Group	MD.FabricPerformanceUser	MDFPUG
Fabric Performance User user ID		MDFPU
Fabric Subscriber Manager Group	MD.FabricSubscriberManager	MDFSMG
Fabric Subscriber Manager user ID		MDFSMU
Fabric Basic User Group	MD.FabricBasicUser	MDFBUG
Fabric Basic User user ID		MDFBU
Fabric Studio User Group	MD.FabricStudioUser	MDFSUG
Fabric Studio Use user ID		MDFSU

### 2.3.4 Create a DB2JccConfiguration.properties file

A DB2JccConfiguration.properties file is required when using a type 2 data source to connect to DB2 for z/OS. This file must contain at least the property db2.jcc.ssid, which specifies the subsystem id (SSID) or group sharing name of the DB2 system to which your WebSphere Process Server for z/OS cell will connect. For example, we specified db2.jcc.ssid=D9FG, where D9FG is the group-sharing name of our DB2 for z/OS system.

In order to understand how the DB2JccConfiguration.properties file is used and why we use a file with that long name, the following section presents background information about JDBC driver properties.

## Background

When accessing DB2 for z/OS through the DB2 JDBC Universal driver there are several ways to specify the DB2 subsystem you wish to connect to. See *DB2 Universal Database for z/OS Application Programming Guide and Reference for Java*, SC18-7414, for a full discussion of the options. The options are as follows:

- ▶ Pass system property `-Ddb2.jcc.ssid` when invoking java. For Java-stored procedures, such properties can be coded in file referred to by a `//JVMPROPS DD` card.
- ▶ Specify the location of a properties file using the JVM system property `db2.jcc.propertiesFile`. This is how WebSphere Application Server for z/OS does it.
- ▶ Add properties to a file called `DB2JccConfiguration.properties` and make that file available to the classpath. It is possible to include the directory that holds the `DB2JccConfiguration.properties` file on the `CLASSPATH`, or include the file in a jar file that is included on the `CLASSPATH`.
- ▶ Specify the DB2 subsystem name in the customized `DSNHDECP` module. Ensure that the customized `SDSNEXIT` that holds that module is above the `SDSNLOAD` in any `JOBLIB` or `STEPLIB` concatenation of any address space accessing DB2.

When using the second or third options, specify the name of the DB2 subsystem to which you want to connect on the property `db2.jcc.ssid`, which you code in the properties file. The options listed above take precedence in the order listed. For example, if `db2.jcc.propertiesFile` is not set, DB2 searches the classpath for a file called `DB2JccConfiguration.properties`. If DB2 also fails to find a file of that name, DB2 tries to contact the DB2 whose SSID is coded in the `DSNHEDCP` module located in the customized `SDSNEXIT` library.

### **Using -D JVM system properties**

It is not common to pass the JDBC properties individually as JVM system properties. Typically, only the `-Ddb2.jcc.propertiesFile` is passed to specify the location of a file that contains the JDBC properties.

### **Using `db2.jcc.propertiesFile`**

In WebSphere Application Server for z/OS the `db2.jcc.propertiesFile` system property is normally the method used to specify the location of the DB2 Universal JDBC driver properties file. The `zWPSConfig.sh` will set this property when you augment nodes to support WebSphere Process Server.

When you use the `db2.jcc.propertiesFile` to set the location of the JCC properties file, you can call the properties file anything you like. We decided to call this file `DB2JccConfiguration.properties`. so it can also be located through a `CLASSPATH` search as described in “Using `DB2JccConfiguration.properties`”.

### ***Using DB2JccConfiguration.properties***

When the db2.jcc.propertiesFile has not been set, DB2 will search CLASSPATH for a DB2JccConfiguration.properties file. When configuring WebSphere Process Server, batch jobs and shell scripts will normally obtain their JCC properties using this method.

The group-sharing name of our DB2 was D9FG, so we created a directory called /etc/d9fg, and created a DB2JccConfiguration.properties file in that directory. We coded db2.jcc.ssid=D9FG in the DB2JccConfiguration.properties file.

**User IDs require read access to the file:** Both the user ID running the configuration jobs and the user ID of the WebSphere Application Server for z/OS Control Region require read access to the file.

Because /etc is not a shared mount point we created an /etc/d9fg directory on each LPAR that hosts a node in our cluster. Alternatively, you can create the DB2JccConfiguration.properties in a directory under a shared mount point so there is only one copy for the whole sysplex.

### ***Using DSNHDECP***

Using DSNHDECP is not a suggested method for specifying the DB2 subsystem to which the DB2 Universal JDBC Driver will connect. It is described here so you are aware of the fact that DB2 will check DSNHDECP if it fails to find JDBC properties by one of the other methods.

The module DSNHDECP is created together with the DSNZPARM module when the DB2 system administrator customizes and executes the job DSNTIJUZ as part of DB2 system customization.

The job DSNTIJUZ assembles and links module DSNHDECP into a customized SDSNEXIT library. If DSNHDECP has not been customized with the correct DB2 SSID, DB2 will try to locate a DB2 subsystem that uses the default SSID of DSN.

Our DSNHDECP module in the customized DB9FU.SDSNEXIT specified the correct SSID for our DB2. Address spaces and shell scripts can then connect to the correct DB2, even if the DB2JccConfiguration.properties file was not found, provided that the customized SDSNEXIT was included above SDSNLOAD in the address space's STEPLIB concatenation.

If you plan to use the DSNHDECP, it is important to specify the customized SDSNEXIT on the STEPLIB concatenation in effect for your UNIX environment. Usually this is done by creating a .profile that includes the following export STEPLIB statement:

```
export STEPLIB=DB9FU.SDSNEXIT:DB9FU.SDSNLOAD:DB9FU.SDSNLOAD2:$STEPLIB
```

## 2.3.5 Customize a .profile

The default UNIX environment that is specified in /etc/profile might be using old versions of Java and DB2 for z/OS. To ensure that the UNIX command-line environment is customized for the correct version of Java and DB2 Universal JDBC Driver, it is necessary to customize a .profile for each user ID that will work with WebSphere Application Server and DB2 z/OS from the command line or from UNIX batch jobs.

For WebSphere Application Server for z/OS, the administrator user ID and the controller user ID share a home directory. For example, the home directory of the administrator user ID of MDCELL was /var/WebSphere/home/MDCFG.

We created a .profile in /var/WebSphere/home/MDCFG that set PATH, LIBPATH, STEPLIB, and CLASSPATH so the UNIX environment was correctly customized for Java and JDBC. (See Example 2-6.) This allowed any shell script we invoked to make JDBC calls and connect to the correct DB2.

JAVA\_HOME is set to point to the same Java used by the deployment manager so that any scripts launched under user MDADMIN will run with the same level of Java as that used by WebSphere Application Server itself.

Rather than concatenate the paths to their existing settings, we decided to make sure the CLASSPATH was how we wanted it by using **unset CLASSPATH** to clear the current CLASSPATH before setting it.

*Example 2-6 Part of a .profile showing JAVA\_HOME and paths necessary for DB2*

---

```
# Java setup
#
export JAVA_HOME=/wasv61config/mdcell/mddmnode/DeploymentManager/java
PATH=$JAVA_HOME/bin/classic:$JAVA_HOME/bin:$PATH
#
# DB2 setup
#
export DB2_PATH=/usr/lpp/db2/d9fg/db2910_jdbc
STEPLIB=DB9FU.SDSNLOAD
STEPLIB=DB9FU.SDSNLOAD2:$STEPLIB
STEPLIB=DB9FU.SDSNEXIT:$STEPLIB
PATH=$DB2_PATH/bin:$DB2_PATH/bin:$PATH
LIBPATH=$DB2_PATH/lib:$DB2_PATH/lib:$LIBPATH
unset CLASSPATH
CLASSPATH=$DB2_PATH/classes/sqlj.zip
CLASSPATH=$DB2_PATH/classes/db2jcc_javax.jar:$CLASSPATH
CLASSPATH=$DB2_PATH/classes/db2jcc_license_cisuz.jar:$CLASSPATH
CLASSPATH=$DB2_PATH/classes/db2jcc.jar:$CLASSPATH
```

```
CLASSPATH=/etc/d9fg/DB2JccConfiguration.properties:$CLASSPATH
export PATH STEPLIB LIBPATH CLASSPATH
#
echo 'JAVA_HOME='$JAVA_HOME
echo 'DB2_PATH='$DB2_PATH
echo 'PATH='$PATH
echo 'STEPLIB='$STEPLIB
echo 'CLASSPATH='$CLASSPATH
echo 'LIBPATH='$LIBPATH
```

---

In addition to setting a .profile for the WebSphere administrator user ID, it is a good idea for the WebSphere systems programmer to use the same .profile for their personal TSO user ID.

### 2.3.6 Verify Java and the DB2 Universal JDBC Driver

After modifying the .profile, we ran a couple of simple tests to check that the command line environment for Java and the DB2 Universal JDBC driver were correct. These tests are not a full verification of JDBC function.

#### Checking that the Java environment is correct

To check that the Java environment is correct, start an OMVS session, and issue the commands in Example 2-7.

*Example 2-7 Testing the Java environment*

---

```
SENIOKJ: /u/seniokj: > su - mdadmin
FSUM5019 Enter the password for mdadmin: <enter the password>
MDADMIN @ SC42:/SC42/var/WebSphere/home/MDCFG> > java -fullversion
java full version "J2RE 1.5.0 IBM z/OS build pmz31devifx-20080502 (SR7
+ IZ15683 + IZ21286)"
```

---

#### Verifying the DB2 JDBC Universal driver check

To verify the DB2 JDBC Universal driver check, use the **db2jcc -version** command, as shown in Example 2-8.

*Example 2-8 Testing that db2jcc can be invoked*

---

```
MDADMIN @ SC42:/SC42/var/WebSphere/home/MDCFG> > db2jcc -version
IBM DB2 JDBC Universal Driver Architecture 3.6.67
```

---

Using the db2jcc command is not a comprehensive test of the JDBC driver. It only confirms that the UNIX command line environment is good enough to launch db2jcc. If you wish to verify that the DB2 Universal Driver has been configured correctly, refer to the Infocenter article *Verify the installation of the IBM Data Server Driver for JDBC and SQLJ* available at the following Web page:

[http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db29.doc.java/db2z\\_jccverifyinstall.htm](http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db29.doc.java/db2z_jccverifyinstall.htm)

## 2.4 Create the deployment manager and an empty node

Configuring a WebSphere Application Server for z/OS deployment manager and an empty node is described in the following IBM White Papers:

- ▶ WP101030, *WebSphere z/OS V6.1 - A Top Down Configuration Approach*

This White Paper is available for download at the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101030>

- ▶ WP100653, *WebSphere for z/OS V6 Sample ND Configuration*

This White Paper is available for download at the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100653>

These papers mention the importance of using intermediate symbolic links and ICFCAT aliases. You should have already prepared these as described in Section 2.3.1, “Create intermediate symbolic links to products” on page 60 and Section 2.3.2, “Create ICFCAT aliases to SBBO\* libraries” on page 61.

In Section 2.4.1, “Task overview” on page 71, we summarize the main tasks involved in configuring the deployment manager and the empty nodes. However, the steps are included in the checklist worksheet in the spreadsheet described in Section 2.2.5, “Planning using a spreadsheet” on page 33.

The starting point of a WebSphere Process Server for z/OS configuration is when you have successfully started the deployment manager and have prepared at least one empty node, but not yet federated it into the cell. It is important not to federate any empty nodes into the cell until after the node had been augmented to support WebSphere Process Server for z/OS.



## 2.4.1 Task overview

After planning the cell (Section 2.2.5, “Planning using a spreadsheet” on page 33) and performing the preparation steps (Section 2.3, “Prepare a WebSphere cell, RACF, and DB2” on page 59) you can configure the deployment manager and an empty node.

In summary, the steps to configure the WebSphere Application Server for z/OS deployment manager and an empty node are as follows:

1. Use the WebSphere Application Server for z/OS ISPF panels or the zPMT tool to create the .CNTL and .DATA datasets with the JCL and properties to configure the deployment manager.
2. Follow the instructions generated by ISPF or zPMT and run the BBO\* jobs to configure the security domain and the deployment manager.
3. Use the WebSphere Application Server for z/OS ISPF panels or the zPMT tool to create the .CNTL and .DATA datasets with the JCL and properties to configure the first empty node.
4. Follow the instructions generated by ISPF or zPMT and run the BBO\* jobs to configure the empty node, but stop before running the job BBOWMNAN that will federate the node into the cell. You must augment the empty node to support WebSphere Process Server for z/OS before federating the node.
5. Repeat the last task to prepare any additional empty nodes you will use.

We used the zProfile Management Tool (zPMT) in the Application Server Toolkit (AST) to configure our WebSphere cell. For an introduction to the AST and zPMT see IBM White Paper WP100871, *Introducing the zPMT Configuration Tool for WebSphere z/OS*, which is available for download at the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100871>

**Note:** The Workstation Customization Tool (WCT) is a new tool that includes zPMT and used to configure WebSphere Application Server for z/OS V7 cells. It can be used to configure WebSphere Application Server V6.1 cells too. You can download the WCT from the following Web page:

<http://www.ibm.com/support/docview.wss?rs=180&uid=swg24020368>

Before using the WCT for the first time, it is suggested to obtain the presentation in Techdoc PRS3357 *WebSphere Application Server for z/OS V7.0: Introducing the WCT for z/OS*, which can be download from the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS3357>

## 2.5 Add WebSphere Process Server to the DMGR

When you have defined a deployment manager you can install support for WebSphere Process Server for z/OS.

Adding WebSphere Process Server for z/OS to the deployment manager's node consists of the following steps:

1. Back up the deployment manager's ZFS data set.  
See page 73.
2. Run the zSMPInstall script.  
See page 73.
3. Prepare DmgrDB2.rsp.  
See page 76.
4. Back up the deployment manager's ZFS.  
See page 77.
5. Run zWPSCfg.sh.  
See page 77.
6. Start the deployment manager.  
See page 78.
7. Set the cell's JVMs to run on local time.  
See page 78.

## 2.5.1 Back up the deployment manager's ZFS data set

Before running zSMPInstall.sh, back up the deployment manager's configuration ZFS data set. We used the JCL shown in Example 2-9 to perform the back up.

*Example 2-9 JCL to back up the deployment manager's node*

```
//MDDUMP JOB NOTIFY=&SYSUID,CLASS=A,MSGCLASS=A,  
// REGION=OM,TIME=1440  
//*  
//* DUMP CONFIG HFS FOR DMGR AND NODE A  
//*  
//DUMP EXEC PGM=ADDRSSU  
//DASD DD DISP=(NEW,CATLG),DSN=WASCFG.MDCELL.DUMP.PREINS,  
// UNIT=3390,SPACE=(CYL,(500,100),RLSE)  
//SYSPRINT DD SYSOUT=*  
//SYSIN DD *  
DUMP DATASET( -  
INCLUDE( -  
WASCFG.MDCELL.MDDMNODE.CONFIG.ZFS -  
) -  
) -  
OUTDD(DASD) -  
TOL(ENQF) OPTIMIZE(4) COMPRESS  
/*
```

**Note:** Remember that all the JCL examples in this chapter are provided in Appendix A, “Additional material” on page 351.

## 2.5.2 Run the zSMPInstall script

When invoked with the -install option the zSMPInstall.sh script performs the following tasks:

- ▶ Validates the prerequisites.
- ▶ Creates the symbolic links from the WebSphere Process Server for z/OS installation file system to the deployment manager's /lib and /bin directories in the configuration file system.
- ▶ Updates the code base permissions.
- ▶ Updates the Integrated Solutions Console with the WebSphere Process Server for z/OS product definitions, including definitions for the following components:
  - Business Process Choreographer (BPC)
  - WebSphere Process Server for z/OS core resources
  - Common Event Infrastructure (CEI)

## Using zSMPInstall.sh

The use of zSMPInstall.sh is now described in more detail.

1. Check to ensure that the deployment manager is stopped before running the zSMPInstall.sh script.
2. You can invoke zSMPInstall.sh from a telnet command line, but we created sample job MDDMINS to do this, as shown in Example 2-10.

On the -smproot property we specified an intermediate symbolic link to the WebSphere Process Server for z/OS product,  
/wasv61config/mdcell/mddmnode\_wps\_smpe, instead of referring directly to the real mount point at/usr/lpp/zWPS/V6R1M2.

*Example 2-10 Job MDDMINS to invoke zSMPInstall.sh*

---

```
//MDDMINS JOB (0),CLASS=A,MSGCLASS=A,REGION=0M,
// USER=MDADMIN,PASSWORD=MDADMIN
//*JOBPARM SYSAFF=SC42
/*****
/*  Install WPS profiles into DMGR                                */
/*****
/*****
/*  STEP TO CREATE LINKS TO WPS LIBRARIES                        */
/*****
//INSTO EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
  cd /wasv61config/mdcell/mddmnode/DeploymentManager/bin; +
  export PATH=.:$PATH; +
  /wasv61config/mdcell/mddmnode_wps_smpe/zos.config/bin/+
  zSMPInstall.sh +
  '-smproot /wasv61config/mdcell/mddmnode_wps_smpe' +
  '-runtime /wasv61config/mdcell/mddmnode/DeploymentManager' +
  '-install' +
  1> /tmp/dmgrinstallonly_0511.out +
  2> /tmp/dmgrinstallonly_0511.err
/*****
/*  STEPS TO COPY THE OUTPUT THE JOB LOG                        */
/*****
//CPOUT EXEC PGM=IKJEFT01,REGION=0M
//SYSEXEC DD DISP=SHR,DSN=BB06142.MDDMNODE.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
  BBOHFSWR '/tmp/dmgrinstallonly_0511.out'
  BBOHFSWR '/tmp/dmgrinstallonly_0511.err'
/*
//
```

---

**Note:** All jobs discussed in this chapter, including MDDMINS, are included in Appendix A, “Additional material” on page 351.

The output from this job will look like Example 2-11.

*Example 2-11 MDDMINS job output*

---

```
BBOHFSWR '/tmp/installonly_1032.out'
CWPIZ0253I: parsing command arguments...
CWPIZ0254I: parsing arguments complete
CWPIZ0255I: setting up configuration...
CWPIZ0256I: set up configuration complete
CWPIZ0257I: creating the symbolic links...
CWPIZ0259I: creation of symbolic links complete
CWPIZ0260I: doing post install file updates...
CWPIZ0262I: post install updates complete
CWPIZ0263I: running Configuration Manager update...
CWPIZ0264I: Configuration Manager update complete
```

---

If any errors occur in the job output, start by looking in the `<dmgr-home>/logs/wbi/install/installconfig.log`. When you find the error, if you issue **FIND ‘buildfile’ PREVIOUS**, it will tell you the ant file that failed. This will direct you to one of the following logs to troubleshoot the problem. It is helpful to sort the logs by date to look at the most recent log file first. In our configuration, `<dmgr_home>` was as shown in Example 2-12:

*Example 2-12 <dmgr\_home> in our configuration*

---

```
/wasv61config/mdcell/mddmnode/DeploymentManager
/<dmgr_home>/logs/wbi/c2n.log
/<dmgr_home>/logs/wbi/zSMPInstall.log
/<dmgr_home>/logs/wbi/zSMPInstall.trace
/<dmgr_home>/logs/wbi/zWPSConfig.log
/<dmgr_home>/logs/wbi/zWPSConfig.trace
/<dmgr_home>/logs/wbi/100SCleanOSGICache.ant.log
/<dmgr_home>/logs/wbi/15SCopyBspaceIfNewer.ant.log
/<dmgr_home>/logs/wbi/80SCopyInstallValidatorLog.ant.log
/<dmgr_home>/logs/wbi/85SConfigNoProfileFirstStepsWBI.ant.log
/<dmgr_home>/logs/wbi/90SCleanDeployTool.ant.log
/<dmgr_home>/logs/wbi/90SConfigureWSPProfileForWBI.ant.log
/<dmgr_home>/logs/wbi/90SConfigWBIMigrationScript.ant.log
/<dmgr_home>/logs/wbi/91SConfigNoProfileFirstStepsCharset.ant.log
/<dmgr_home>/logs/wbi/93SDeployBPCAdminConsolePlugins.ant.log
/<dmgr_home>/logs/wbi/93SDeployBusinessSpaceAdminConsoleFeature.log
/<dmgr_home>/logs/wbi/93SDeployServerAdminConsolePlugins.ant.log
/<dmgr_home>/logs/wbi/93SDeployWBICommonAdminConsolePlugins.ant.log
/<dmgr_home>/logs/wbi/94SDeployCoreAdminConsolePlugins.ant.log
```

---

## 2.5.3 Prepare DmgrDB2.rsp

Parameters and their values are passed to the zWPSConfig.sh in a response file. A sample response file called DmgrDB2.rsp is provided with WebSphere Process Server for z/OS for augmenting a deployment manager when you will use DB2 for z/OS as a data store.

1. Copy the DmgrDB2.rsp response file from /<wps\_install\_root>/zos.config/ to a work directory and rename it mddmnodeDmgrDB2.rsp. In our case, <wps\_install\_root>was /usr/lpp/zWPSMD/V6R1M2.

The work directory was /var/WebSphere/home/MDCFG/wpwork.

**Note:** We were configuring more than one WebSphere Process Server for z/OS cell and so decided to change the name of the response file from DmgrDB2.rsp to mddmnodeDmgrDb2.rsp to avoid confusion.

2. Customize the properties in the response file as shown in Example 2-13.

*Example 2-13 mddmnodeDmgrDB2.rsp*

---

```
JMSUSER=MDJMSU
JMSPASS=MDJMSU
CONFIGSERVER=dmgr
DBPRODUCT=DB2UDBOS390_V9_1
DBLOCATION=DB9F
DBPROPERTIES=/etc/d9fg
DBJDBCCLASSPATH=/usr/lpp/db2/d9fg/db2910_jdbc/classes
DBUSER=MDDBU
DBPASS=MDDBU
DBHOSTNAME=wtsc42.itso.ibm.com
DBSERVERPORT=37893
SQLID=MDWPS
SQLDB=MDWPSDB
SQLSTO=MDWPSSG
augment
profileName=default
profilePath=/wasv61config/mdcell/mddmnode/DeploymentManager/\
profiles/default
templatePath=/wasv61config/mdcell/mddmnode/DeploymentManager/\
profileTemplates/dmgr.wbiserver
cellName=mdcell
nodeName=mddmnode
enableAdminSecurity=true
adminUserName=$JMSUSER
adminPassword=$JMSPASS
```

```

serverName=$CONFIGSERVER
dbCreateNew=false
dbDelayConfig=true
configureScaSecurity=true
scaSecurityUserId=$JMSUSER
scaSecurityPassword=$JMSPASS
configureAppScheduler=false
appSchedulerServer=$CONFIGSERVER
dbName=$SQLDB
dbStorageGroup=$SQLSTO
dbType=$DBPRODUCT
dbConnectionLocation=$DBLOCATION
dbJDBCProperties=$DBPROPERTIES
dbJDBCClasspath=$DBJDBCCLASSPATH
dbUserId=$DBUSER
dbPassword=$DBPASS
dbHostName=$DBHOSTNAME
dbServerPort=$DBSERVERPORT
dbSchemaName=$SQLID

```

---

## 2.5.4 Back up the deployment manager's ZFS

Back up the deployment manager configuration ZFS data set again before running `zWPSConfig.sh`. We used the JCL shown in Example 2-9 on page 73 but changed the `//DASD DD` card to `//DASD DD DSN=WASCFG.MDCELL.DUMP.PRECFGD`.

## 2.5.5 Run `zWPSConfig.sh`

After preparing the response file, we executed the shell script `zWPSConfig.sh` by submitting job `MDDMCFG`, as shown in Example 2-14. This job took about 6 minutes to run.

*Example 2-14 MDDMCFG JCL to run `zWPSConfig` for the DMGR*

---

```

//MDDMCFG JOB (0),CLASS=A,MSGCLASS=A,REGION=0M,
// USER=MDADMIN,PASSWORD=MDADMIN
/*JOBPARM SYSAFF=SC42
/*****
/* Configure WPS in deployment manager mddmnode */
/*****
/* Run zWPSConfig.sh */
/*****

```

```

//INSTO EXEC PGM=IKJEFT01,REGION=OM,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
  cd /wasv61config/mdcell/mddmnode/DeploymentManager/bin; +
  ./zWPSConfig.sh +
  -response /var/WebSphere/home/MDCFG/wpswork/+
  mddmnodeDmgrDB2.rsp +
  -augment +
  1> /tmp/mddmnodezWPSConfig_0511.out +
  2> /tmp/mddmnodezWPSConfig_0511.err;
/*
/*****
/* STEPS TO COPY THE OUTPUT THE JOB LOG */
/*****
//CPOUT EXEC PGM=IKJEFT01,REGION=OM
//SYSEXEC DD DISP=SHR,DSN=BB06142.MDDMNODE.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
  BBOHFSWR '/tmp/mddmnodezWPSConfig_0511.out'
  BBOHFSWR '/tmp/mddmnodezWPSConfig_0511.err'
/*

```

---

Be sure to run the job using the WebSphere Administrator user ID. Ensure that the response file (mddmnodeDmgrDB2.rsp in Example 2-14 on page 77) is readable by the WebSphere administrator user ID.

## 2.5.6 Start the deployment manager

Now that the deployment manager's node has been configured, start the deployment manager to confirm that it initializes properly before configuring the empty nodes. We used the following command:

```
S MDMGCR,JOBNAME=MDDMGR,ENV=MDCELL.MDDMNODE.MDDMGR
```

## 2.5.7 Set the cell's JVMs to run on local time

By default, the time-stamps of messages issued by the cell's JVMs are in Coordinated Universal Time (UCT). This makes it more difficult to debug problems because the timestamps will probably not match messages elsewhere in the joblogs or in the system's syslog. It is a good idea to run the JVMs using local time.



To set the cell's JVMs to run on local time, perform the following steps:

1. Log in to the WebSphere Integrated Solution Console and navigate to **Environment** → **WebSphere variables**.
2. Set the scope to the cell level and click **New**.
3. Add the variable name `ras_time_local` and set its value to 1 as shown in Figure 2-9.

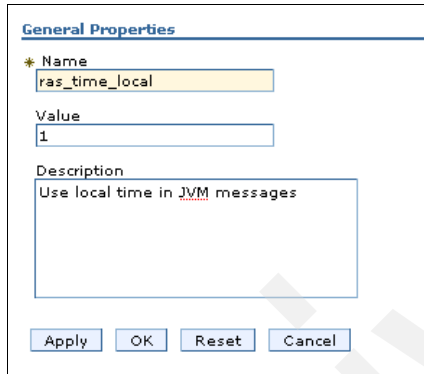


Figure 2-9 Setting the cell's JVMs to run on local time

4. Click **OK** and then save the configuration changes.
5. Click **OK** to save the changes.

**Remember:** The change will not take effect in the deployment manager until the next time it is restarted.

## 2.6 Configure DB2 z/OS using createDB.sh

As soon as the deployment manager has been augmented we suggest that you use the `createDB.sh` to generate all the DDL.

Note that you should execute `createDB.sh` from a telnet client. Do not use an OMVS session because OMVS does not support vi and `createDB.sh` will use vi to make some global changes to DDL.

When you use `createDB.sh` to generate the DDL, the customization is more complete and there are fewer files to manage. Therefore it is certainly the quickest and best way to generate DDL.

Perform the following steps to configure DB2 z/OS using createDB.sh:

1. Copy createDB.sh to a work directory.  
See page 80.
2. Customize the sample createDB.sh script.  
See page 81.
3. Execute createDB.sh to generate the DDL.  
See page 83.
4. Modify the generated DDL.  
See page 84.
5. Create the databases.  
See page 86.
6. Execute the DDL using createDB.sh.  
See page 86.
7. Execute the DDL using DSNTEP2.  
See page 87.
8. Grant table privileges to the J2C auth alias user ID.  
See page 88.

### 2.6.1 Copy createDB.sh to a work directory

Set up a work directory to hold a copy of createDB.sh that you will customize.

1. Open a telnet session and login as the WebSphere administrative user.
2. Create a wpswork directory under the home directory of the WebSphere administrative user ID. For example, for our MD cell we issued the following command:

```
mkdir /var/WebSphere/home/MDCFG/wpswork
```

3. Copy the sample script createDB.sh from  
/<wps\_install\_root>/zos.config/samples to the work directory.

In our case, <wps\_install\_root> is /usr/lpp/zWPS/V6R1M2. The work directory is /var/WebSphere/home/MDCFG/wpswork.

4. From the same directory, also copy the sibDropxx.sql file and the sibSchemaxx.sql file into the work directory.

## 2.6.2 Customize the sample createDB.sh script

The next step is to modify the createDB.sh.

1. Although the permissions should be correct, we assigned the appropriate permissions to the new version of createDB.sh. See Example 2-15.

*Example 2-15 Assigning permissions to createDB.sh*

---

```
cd /var/WebSphere/home/MDCFG/wpswork
chmod 755 createDB.sh
```

---

2. Update the copy of createDB.sh script with the values appropriate for your environment as follows:
  - a. The top section of the script contains a list of fields used during the execution of the script. Modify the default values of the fields to match your environment.

**Note:** DBDelayConfig=true prevents the createDB.sh from executing the DDL. If you set DBDelayConfig=false, the properties DBCREATE, DBWPSCreate, ... DBSIBCreate control which DDL is executed. If you pass the +All option when invoking createDB.sh, it sets all the DBxxCreate option to true.

If your DB2 for z/OS system's SDSNLOAD and SDSNLOD2 libraries are not in LINKLST, you will need to export STEPLIB as well as LIBPATH as shown in Example 2-16.

*Example 2-16 Modified createDB.sh for our MD cell*

---

```
LIBPATH=/usr/lpp/db2/d9fg/db2910_jdbc/jcc/lib:$LIBPATH
STEPLIB=DB9F9.SDSNEXIT:DB9F9.SDSNLOAD:DB9F9.SDSNLOD2:$STEPLIB
export LIBPATH STEPLIB
SCRIPTHOME=`pwd`
export SCRIPTHOME
SKIP=0
INVALIDoption=0;

DBCREATE=true
DBUSESQLID=true

DBWPSCreate=true
DBBPCCreate=true
DBCEICreate=true
DBSIBCreate=true
```

```

DBGenerate=true

DBPREFIX=MD
DBSCOPE=01
DBJDBCClasspath=/usr/lpp/db2/d9fg/db2910_jdbc/classes
DBUSER=MDDBU
DBPASSWORD=MDDBU
DBJDBCProperties=/etc/d9fg
DBConnectionLocation=DB9F
DBDelayConfig=true
DBHostName=wtsc42.itso.ibm.com
DBServerPort=37893
DBVOLUMES="'*'"
DBVCAT=WASCFG

CEIBP4K=BP9
CEIBP8K=BP8K2
CEIBP16K=BP16K2
CEISIZE=10

SCRIPTTEMP=$SCRIPTHOME/tmp
OUTPUT=$SCRIPTTEMP/output.out
ERROR=$SCRIPTTEMP/error.out

```

---

- b. An additional modification we made was to change the storage group, schema, and database name that were being generated and used by the script so they matched the names we wanted to use. The variables to change are further down the createDB.sh and are shown in Example 2-17.

*Example 2-17 Setting storage group, schema and database name in createDB.sh*

---

```

DBSTO=${DBPREFIX}WPSSG
DBSCHEMA=${DBPREFIX}WPS
DBNAME=${DBPREFIX}WPSDB

```

---

**Note:** There is only one variable DBNAME in createDB.sh. This means that if you run createDB.sh with the +All option, all the tables are created in one database.

If you want to define the different components of WebSphere Process Server for z/OS in different databases, you can run createDB.sh multiple times, specifying a different DBNAME variable for each component of WebSphere Process Server for z/OS.

Alternatively, you can use the +All option and then manually edit the DDL later as described in Section 2.6.4, “Modify the generated DDL” on page 84.

We found it was easier to execute createDB.sh with the +All option and edit the database names in the DDL afterwards.

- c. Because we were using DB2 version 9, the following two global changes were made to replace hard coded information in the createDB.sh script.
  - i. We replaced DB2zOSV8 with DB2zOSV9.
  - ii. We replaced DB2UDBOS390\_V8\_1 with DB2UDBOS390\_V9\_1.

### 2.6.3 Execute createDB.sh to generate the DDL

After setting up the work directory and modifying the script, you can run the createDB.sh script to generate the DDL.

1. Set up the correct UNIX environment by running the setupCmdLine.sh. See Example 2-18.

*Example 2-18 Setting the command line*

---

```
cd /wasv61config/mdcell/mddmnode/DeploymentManager/bin
. setupCmdLine.sh
```

---

Notice that the command **. setupCmdLine.sh** is preceded by a period (“.”). Running the command in this fashion retains the export of WAS\_HOME that is performed by setupCmdLine.sh and makes WAS\_HOME available to the createDB.sh script.

2. Run the modified createDB.sh script to generate the DDL. See Example 2-19.

*Example 2-19 Executing createDB.sh*

---

```
cd /var/WebSphere/home/MDCFG/wpswork
./createDB.sh +All
```

---

When invoking createDB.sh to generate DDL for one component, you need to pass the -All option (minus-All instead of plus-All) to suppress creation of DDL for everything. You then specify an option like +DBBPC to generate DDL for just the BPC component as shown in Example 2-20.

*Example 2-20 Executing createDB.sh for one component*

---

```
cd /var/WebSphere/home/MDCFG/wpswork
./createDB.sh -All +DBBPC
```

---

3. Confirm that the DDL was successfully created. When we ran createDB.sh to generate DDL for DB2 for z/OS V9, the files shown in Example 2-21 were generated in the following directory:

<dmgr\_home>/profiles/default/dbscripts/CommonDB/DB2zOSV9/MDWPSDB.

*Example 2-21 DDL files generated by createDB.sh*

---

bpc.sql	createDatabaseX.sql	sibCEI.sql
ceidb.sql	sibAPP.sql	sibSCA.sql
common.sql	sibBPC.sql	

---

## 2.6.4 Modify the generated DDL

After the DDL is generated using the createDB.sh script, some manual changes are still needed.

**Note:** The DDL generated by createDB.sh is in ASCII. If you intend to execute the DDL using createDB.sh, it must remain in ASCII.

If you have z/OS V1.9, you can use ISPF option 3.17 to edit an ASCII file. Alternatively, use an FTP client to download the DDL in BINARY to a workstation where it can be edited and uploaded again in BINARY.

If you intend to execute the DDL using DSNTEP2, it is probably better to convert the DDL to EBCDIC, copy it into z/OS sequential files, and then edit it using the normal ISPF editor. This is described in Section 2.6.7, “Execute the DDL using DSNTEP2” on page 87.

Be sure to pre-allocate the z/OS sequential files with sufficient space. When using oget to copy files from Unix to z/OS you are not warned if the files runs out of space. Some records may not be copied.

Here is a summary of the changes we made to the files listed in Example 2-21 on page 84:

► `bpc.sql`

The `bpc.sql` required a several changes because we wanted to use separate databases for BPC and BPC Observer, but the generated DDL assigns them to a single database.

If you generate the DDL using `createDB.sh` with the `+All` option, make a global change of the database name from the common database name to the database name you want to use for BPC.

For our MD cell, for example, we changed `MDWPSDB` to `MDBPCDB`. Locate the section in `bpc.sql` with the BPC Observer DDL and replace all the `MDBPCDB` database references with `MDOBSDB`. The ISPF editor `exclude` command (x) is useful for making selective changes like this.

► `ceidb.sql`

The buffer pools for `ceidb.sql` were properly passed from `createDB.sh`, so the only change required was a global change of the database name. The database name was changed from `MDWPSDB` to `MDEVTDDB`.

► `common.sql`

Only the buffer pools need to be changed for `common.sql`. A global find and replace was done to change `BP0` to `BP1` and `BP16K0` to `BP16K1`.

► `createDatabaseX.sql`

We made no changes to this file because we did not use it. We created some JCL called `MDCREDB`, which created the databases and issued appropriate `GRANTS` to delegate authority over the databases. This is described in Section 2.6.5, “Create the databases” on page 86.

► For all 4 `sibxxx.sql` files, `sibAPP.sql`, `sibBPC.sql`, `sibCEI.sql`, and `sibSYS.sql`, we changed all the `PRIQTY` and `SECQTY` entries to be `-1`. This must be done carefully because in some of the DDL the `PRIQTY` or `SECQTY` property are not on the same line as their values.

**Note:** Setting `PRIQTY` and `SECQTY` to `-1` allows DB2 for z/OS to manage space allocation of tablespaces and indexes depending on the volume of data they contain. This means you do not have to worry about space allocations at the time you create tablespaces and indexes.

- ▶ For all 4 sibxxx.sql files, sibAPP.sql, sibBPC.sql, sibCEI.sql and sibSYS.sql, we set the correct buffer pools and changed the database name as follows.
  - sibAPP.sql  
The database name changed from MDWPSDB to MD01ADB.
  - sibBPC.sql  
The database name changed from MDWPSDB to MD01BDB.
  - sibCEI.sql  
The database name changed from MDWPSDB to MD01CDB.
  - sibSCA.sql  
The database name changed from MDWPSDB to MD01SDB.

## 2.6.5 Create the databases

Instead of running createDB.sh to create one database using the DDL in file createDatabaseX.sql, we created a sample job MDCREDB that created several databases and granted the needed permissions. See Appendix A, “Additional material” on page 351 for information about how to obtain this sample JCL.

We recommend that you pay particular attention to the MDCREDB job because it consolidates a lot of our experiences related to properly setting up security. The GRANT statements in MDCREDB are discussed in Section 2.2.11, “Implications for GRANT statements in DB2” on page 54. Ask your DB2 database administrator to execute MDCREDB because SYSADM authority is needed.

## 2.6.6 Execute the DDL using createDB.sh

**Note:** If you prefer to execute the DDL using DSNTDP2 see Section 2.6.7, “Execute the DDL using DSNTDP2” on page 87.

After customizing the DDL and creating the database(s) you can use createDB.sh to execute the DDL too.

**Note:** Remember to change to the <dmgr\_root>/bin directory and execute the command . setupCmdLine.sh before running createDB.sh.



We ran the createDB.sh four times (as shown in Example 2-22), once for each major component, so we can more easily isolate any errors. The majority of the output messages and any error messages are written to files output.out and error.out in the /tmp directory that is created under the /wpswork directory from which you executed createDB.sh.

*Example 2-22 createDB.sh commands with results summary*

---

```
./createDB.sh -All +DBWPS -gen -DBDelayConfig false
Configuring WPS Definitions
[sql] 88 of 88 SQL statements executed successfully
./createDB.sh -All +DBBPC -gen -DBDelayConfig false
Configuring BPC Definitions
[sql] 2033 of 2033 SQL statements executed successfully
./createDB.sh -All +DBCEI -gen -DBDelayConfig false
Configuring CEI Tables
[sql] 375 of 375 SQL statements executed successfully
./createDB.sh -All +DBSIB -gen -DBDelayConfig false
Configuring SIB Definitions
SCA Bus Schema MD01S
[sql] 40 of 40 SQL statements executed successfully
APP Bus Schema MD01A
[sql] 40 of 40 SQL statements executed successfully
BPC Bus Schema MD01B
[sql] 40 of 40 SQL statements executed successfully
CEI Bus Schema MD01C
[sql] 40 of 40 SQL statements executed successfully
```

---

## 2.6.7 Execute the DDL using DSNTPE2

**Note:** Skip this step if you executed the DDL using createDB.sh in Section 2.6.6, “Execute the DDL using createDB.sh” on page 86.

If you prefer to execute the DDL using DSNTPE2, you must first copy it from the UNIX file system into z/OS sequential files or as members of a partitioned dataset. Because the DDL is in ASCII, it must also be converted to EBCDIC before it can be executed by DSNTPE2.

We created sample job MDCPYDDL to do this. The job converts the DDL from ASCII to EBCDIC and copies it into z/OS sequential files where you can edit it with the ISPF editor. We suggest that you pre-allocate these with SPACE(TRK(1,2)) to avoid the risk that some records are not copied.

The sequential files are allocated as VB. After editing the DDL, copy it into a partitioned dataset with an LRECL of 80 so it can be executed using DSNTEP2. We used a FB LRECL 80 PDS called WASCFG.MDCELL.WPS.SQL. We created sample job MDDDL to execute the DDL using DSNTEP2. Both the MDCPYDDL and MDDDL jobs are provided in the additional material in Appendix A, “Additional material” on page 351.

## 2.6.8 Grant table privileges to the J2C auth alias user ID

As described in Section 2.2.11, “Implications for GRANT statements in DB2” on page 54, if you decided to use currentSQLID on the data sources, and granted sufficient authority for the servant region to switch to that user ID, the J2C authentication alias user ID will have implicit privileges over the tables. You do not need to issue GRANTS for table privileges. If you are using currentSchema, however, issue GRANT commands that give the J2C authentication alias user ID privileges over the tables.

We created a series of members called GRANTxxx in our WPS.SQL dataset by copying the supplied DDL and editing all the CREATE TABLE statements so they became GRANT ALL ON commands.

We then created sample job MDGRANT to execute all the GRANT statements using DSNTEP2. You can obtain the GRANT statements we used and the sample JCL MDGRANT from the additional material in Appendix A, “Additional material” on page 351.

## 2.7 Adding WebSphere Process Server to the empty nodes

This section describes the steps to install WebSphere Process Server for z/OS into an empty node:

1. Empty node: Run the zSMPInstall script  
See page 89.
2. Empty node: Prepare ManagedDB2.rsp  
See page 90.
3. Back up the WebSphere configuration HFS datasets  
See page 92.
4. Empty node: Run zWPSConfig.sh  
See page 92.

5. Set the currentSchema property on WBI\_DataSource  
See page 93.
6. Clear the OSGi cache  
See page 94.
7. Back up the WebSphere configuration HFS datasets  
See page 94.
8. Federate the primary empty node into the cell  
See page 95.
9. Update the primary node's Node Agent ports  
See page 95.
10. Create a personal certificate for the servant  
See page 98.
11. Back up the WebSphere configuration HFS datasets  
See page 98.

An empty node should be configured up to the BBOMHFSB job but not federated into the cell. The node must be augmented to support WebSphere Process Server before it is federated.

### 2.7.1 Empty node: Run the zSMPInstall script

We created sample job MDNAINS, shown in Example 2-23, to run the zSMPInstall.sh against the first empty node. It took about 10 minutes to run.

*Example 2-23 Sample job MDNAINS*

---

```
//MDNAINS JOB (0),CLASS=A,MSGCLASS=A,REGION=0M,
// USER=MDADMIN,PASSWORD=MDADMIN
/*JOBPARM SYSAFF=SC42
//*****
/* Install WPS profiles into empty node MDNODEA */
//*****
//INSTO EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
cd /wasv61config/mdcell/mdnodea/AppServer/bin; +
export PATH=.:$PATH; +
/wasv61config/mdcell/mdnodea_wps_smpe/zos.config/bin/+
zSMPInstall.sh +
```

```

'-smproot /wasv61config/mdcell/mdnodea_wps_smpe' +
'-runtime /wasv61config/mdcell/mdnodea/AppServer' +
'-install' +
1> /tmp/mdnodeainstall_0511.out +
2> /tmp/mdnodeainstall_0511.err
/*****
/*  STEPS TO COPY THE OUTPUT THE JOB LOG          */
/*****
//CPOUT EXEC PGM=IKJEFT01,REGION=0M
//SYSEXEC DD DISP=SHR,DSN=BB06142.MDNODEA.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
        BBOHFSWR '/tmp/mdnodeainstall_0511.out'
        BBOHFSWR '/tmp/mdnodeainstall_0511.err'
/*

```

---

## 2.7.2 Empty node: Prepare ManagedDB2.rsp

Perform the following steps to prepare ManagedDB2.rsp.

1. Copy the ManagedDB2.rsp response file from /<wps\_install\_root>/zos.config/ to a work directory. In our case, <wps\_install\_root> is /usr/lpp/zWPSMD/V6R1M2. The work directory is /var/WebSphere/home/MDCFG/wpswork

It is a good idea to include the node name in the response file name so you can tell one from the other. For example for the primary node of our cell called MDCELL, we named the response file mdnodeaManagedDB2.rsp.

2. Customize the properties in the response file for your installation. We set up the values as shown in Example 2-24 on page 91.

**Note:** We found that the sample ManagedDB2.rsp was missing the property DBJDBCCLASPPATH=/usr/lpp/db2/d9fg/db2910\_jdbc/classes. We added that property by copying it from the DmgrDB2.rsp.

Also, the templatePath is split across two lines in Example 2-24 but is coded on one line in the response file.

*Example 2-24 mdnodeaManagedDB2.rsp for the primary node*

---

```
JMSUSER=MDJMSU
JMSPASS=MDJMSU
DBPRODUCT=DB2UDBOS390_V9_1
DBLOCATION=DB9F
DBPROPERTIES=/etc/d9fg
DBJDBCCLASSPATH=/usr/lpp/db2/d9fg/db2910_jdbc/classes
DBUSER=MDDBU
DBPASS=MDDBU
DBHOSTNAME=wtsc42.itso.ibm.com
DBSERVERPORT=37893
SQLDB=MDWPSDB
SQLSTO=MDWPSSG
augment
profileName=default
profilePath=/wasv61config/mdcell/mdnodea/AppServer/profiles/default
templatePath=/wasv61config/mdcell/mdnodea/AppServer/profileTemplates/\
managed.wbiserver
cellName=mdemptya
nodeName=mdnodea
dbCreateNew=false
dbDelayConfig=true
configureScaSecurity=true
scaSecurityUserId=$JMSUSER
scaSecurityPassword=$JMSPASS
configureAppScheduler=false
dbName=$SQLDB
dbStorageGroup=$SQLSTO
dbType=$DBPRODUCT
dbConnectionLocation=$DBLOCATION
dbJDBCProperties=$DBPROPERTIES
dbJDBCClasspath=$DBJDBCCLASSPATH
dbUserId=$DBUSER
dbPassword=$DBPASS
dbHostName=$DBHOSTNAME
dbServerPort=$DBSERVERPORT
```

---

## 2.7.3 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and the primary empty node's configuration ZFS data set before running zWPSCfg.sh against the empty node. We used the sample JCL shown in Example 2-25.

*Example 2-25 Sample JCL to dump the DMGR and primary node*

---

```
//MDDUMP    JOB NOTIFY=&SYSUID,CLASS=A,MSGCLASS=A,
// REGION=OM,TIME=1440
//*
//*    DUMP CONFIG HFS FOR DMGR AND NODE A
//*
//DUMP      EXEC PGM=ADRDSSU
//DASD      DD DISP=(NEW,CATLG),DSN=WASCFG.MDCELL.DUMP.PRECFGD,
// UNIT=3390,SPACE=(CYL,(1000,100),RLSE),VOL=SER=TST052
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD *
    DUMP DATASET(      -
        INCLUDE(      -
            WASCFG.MDCELL.MDDMNODE.CONFIG.ZFS      -
            WASCFG.MDCELL.MDNODEA.CONFIG.ZFS      -
        )              -
    )                  -
    OUTDD(DASD)      -
    TOL(ENQF) OPTIMIZE(4) COMPRESS
/*
```

---

## 2.7.4 Empty node: Run zWPSCfg.sh

After preparing the mdnodeaManagedDB2.rsp response file, we executed the shell script zWPSCfg.sh by submitting the job MDNACFG, as shown in Example 2-26 on page 93. This job took about three minutes to run.

The job ran under the WebSphere Administrator user ID so that the updates can be made to the node's configuration. You must ensure that the response file (/var/WebSphere/home/MDCFG/wpswork/mdnodeaManagedDB2.rsp in Example 2-14 on page 77) is readable by the WebSphere Administrator user ID.

```
//MDNACFG JOB (0),CLASS=A,MSGCLASS=A,REGION=0M,
// USER=MDADMIN,PASSWORD=MDADMIN
/*JOBPARM SYSAFF=SC42
/*****
/* Configure WPS in empty node mdnodea */
/*****
/*****
/* Run zWPSCfg.sh */
/*****
//INSTO EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
    cd /wasv61config/mdcell/mdnodea/AppServer/bin; +
    ./zWPSCfg.sh +
-response /var/WebSphere/home/MDCFG/wpswork/+
mdnodeaManagedDB2.rsp +
-augment +
    1> /tmp/mdnodeazWPSCfg_0511.out +
    2> /tmp/mdnodeazWPSCfg_0511.err;
/*
/*****
/* STEPS TO COPY THE OUTPUT THE JOB LOG */
/*****
//CPOUT EXEC PGM=IKJEFT01,REGION=0M
//SYSEXEC DD DISP=SHR,DSN=BBO6142.MDNODEA.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
    BBOHFSWR '/tmp/mdnodeazWPSCfg_0511.out'
    BBOHFSWR '/tmp/mdnodeazWPSCfg_0511.err'
/*
```

---

## 2.7.5 Set the currentSchema property on WBI\_DataSource

The WBI\_Datasource that is created at cell level by the zWPSCfg.sh script has the schema set on the currentSQLID property. We recommend that you use currentSchema rather than currentSQLID as discussed in 2.2.10, “Choosing currentSchema or currentSQLID” on page 48.

To use currentSchema rather than currentSQLID, perform the following steps.

1. Log in to the Integrated Solutions Console and then navigate to **Resources** → **JDBC** → **JDBC providers**.
2. Select the DB2 Universal JDBC Driver Provider that is defined at cell level, and click **Data sources**.
3. Click the data source **WBI\_Datasource** and then **Custom properties**.
4. Click the currentSQLID property and clear its value.
5. Click the currentSchema property. Set its value to the schema for the WebSphere Process Server for z/OS common database tables. For our MD cell we set the currentSchema property to MDWPS.
6. Click **OK**. Save the configuration changes.
7. After updating the WBI\_Datasource, restart the Deployment Manager.

## 2.7.6 Clear the OSGi cache

Clear the OSGi cache as explained in Technote 1289944, *NoClassDefFoundError error during SCA configuration*, available at the following Web page:

<http://www.ibm.com/support/docview.wss?uid=swg21289944>

Running the osgiCfgInit.sh script deletes the OSGi cache and makes it ready to be rebuilt when WebSphere Process Server for z/OS is started. This will prevent possible NoClassDefFoundError exceptions. From a telnet session, login with the WebSphere administrator user ID and then enter the following commands:

```
cd /wasv61config/mdcell/mddmnode/DeploymentManager/bin
./osgiCfgInit.sh
```

## 2.7.7 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and the Empty Node's configuration ZFS data set before federating the node. We used the JCL shown in Example 2-25 on page 92 but changed the //DASD DD card to the following code:

```
//DASD DD DSN=WASCFG.MDCELL.DUMP.PREFEDA
```



## 2.7.8 Federate the primary empty node into the cell

Perform the following steps to Federate the primary empty node into the cell.

1. Return to the CNTL dataset that contains the jobs for configuring the primary empty node. Locate the BBOMNAN job, and edit it.
2. Add the user ID and password of the WebSphere administrator user to the job card. For example, the job card when we federated our primary node into the MDCELL was as shown in Example 2-27.

*Example 2-27 Job card*

---

```
//BBOWMNAN JOB (0),'WPS SETUP',CLASS=A,REGION=OM,  
//          MSGCLASS=H,NOTIFY=&SYSUID,  
//          USER=MDADMIN,PASSWORD=MDADMIN  
/*JOBPARM SYSADFF=SC42
```

---

**Note:** If you do not run the job under the WebSphere administrator user ID, and security is enabled in the cell, the BBOWMNAN job will fail with an error related to SSL. This is because the user ID running the job must have the deployment manager's CA certificate in its keyring in order for an SSL handshake with the deployment manager to succeed.

The WebSphere administrator user ID has such a keyring. A normal TSO user ID will not, unless you create a keyring for your TSO user ID and connect the cell's CA certificate to the keyring. Running BBOWMNAN under the WebSphere administrator user ID avoids any problems.

Use a /\*JOBPARM SYSAFF card to ensure that the BBOWMNAN job runs on the LPAR, which is the home of the primary node you are federating. If you chose the option to start the node agent after federation, the BBOWMNAN job will issue a start command for the node agent. BBOWMNAN must run on the LPAR where the primary node's node agent is started.

3. Submit BBOMNAN. If you chose to start the node agent during federation, the node agent is started by job BBOWMNAN just before it ends.

## 2.7.9 Update the primary node's Node Agent ports

Most of the Node Agent ports match those you planned to use because they are configured during federation according to the values that were passed into the zPMT tool when configuring the node. However, some of the ports take default values.

Set these to the planned port numbers by performing the following steps.

1. Log in to the WebSphere Integrated Solutions Console.
2. Navigate to **System administration** → **Node agents** and click the link to the first Node Agent.

**Note:** If you were already logged into the WebSphere Integrated Solutions Console before the node was federated, you may not see any nodes listed when you navigate to **System administration** → **Node agents**.

Log out of the Integrated Solutions Console and log in again.

3. Under Additional properties, click **Ports**. The ports are listed as shown in Figure 2-10. Notice how some are taking default values.

Select	Port Name ↕	Host ↕	Port ↕
<input type="checkbox"/>	<a href="#">BOOTSTRAP_ADDRESS</a>	wtsc42.itso.ibm.com	20021
<input type="checkbox"/>	<a href="#">CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS</a>	wtsc42.itso.ibm.com	9200
<input type="checkbox"/>	<a href="#">CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS</a>	wtsc42.itso.ibm.com	9201
<input type="checkbox"/>	<a href="#">DCS_UNICAST_ADDRESS</a>	*	20025
<input type="checkbox"/>	<a href="#">DRS_CLIENT_ADDRESS</a>	wtsc42.itso.ibm.com	7888
<input type="checkbox"/>	<a href="#">NODE_DISCOVERY_ADDRESS</a>	wtsc42.itso.ibm.com	20023
<input type="checkbox"/>	<a href="#">NODE_IPV6_MULTICAST_DISCOVERY_ADDRESS</a>	ff01::1	20033
<input type="checkbox"/>	<a href="#">NODE_MULTICAST_DISCOVERY_ADDRESS</a>	232.133.104.73	20024
<input type="checkbox"/>	<a href="#">ORB_LISTENER_ADDRESS</a>	*	20021
<input type="checkbox"/>	<a href="#">ORB_SSL_LISTENER_ADDRESS</a>	*	20022
<input type="checkbox"/>	<a href="#">SAS_SSL_SERVERAUTH_LISTENER_ADDRESS</a>	wtsc42.itso.ibm.com	9901
<input type="checkbox"/>	<a href="#">SOAP_CONNECTOR_ADDRESS</a>	wtsc42.itso.ibm.com	20020

Figure 2-10 Node agent ports immediately after federation

- Click each of the Port names in turn and set the correct value as shown for the CSIV2\_SSL\_MUTUALAUTH\_LISTENER\_ADDRESS in Figure 2-11.

**General Properties**

**Port Name**  
CSIV2\_SSL\_MUTUALAUTH\_LISTENER\_AD

\* Host  
wtsc42.itso.ibm.com

\* Port  
20130

Apply OK Reset Cancel

Figure 2-11 Correcting a node agent port

- When you have finished, the ports must all be as planned on the cell-planning spreadsheet, as shown in Figure 2-12.

<input type="checkbox"/>	<a href="#">BOOTSTRAP_ADDRESS</a>	wtsc42.itso.ibm.com	20021
<input type="checkbox"/>	<a href="#">CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS</a>	wtsc42.itso.ibm.com	20130
<input type="checkbox"/>	<a href="#">CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS</a>	wtsc42.itso.ibm.com	20131
<input type="checkbox"/>	<a href="#">DCS_UNICAST_ADDRESS</a>	*	20025
<input type="checkbox"/>	<a href="#">DRS_CLIENT_ADDRESS</a>	wtsc42.itso.ibm.com	20132
<input type="checkbox"/>	<a href="#">NODE_DISCOVERY_ADDRESS</a>	wtsc42.itso.ibm.com	20023
<input type="checkbox"/>	<a href="#">NODE_IPV6_MULTICAST_DISCOVERY_ADDRESS</a>	ff01::1	20033
<input type="checkbox"/>	<a href="#">NODE_MULTICAST_DISCOVERY_ADDRESS</a>	232.133.104.73	20024
<input type="checkbox"/>	<a href="#">ORB_LISTENER_ADDRESS</a>	*	20021
<input type="checkbox"/>	<a href="#">ORB_SSL_LISTENER_ADDRESS</a>	*	20022
<input type="checkbox"/>	<a href="#">SAS_SSL_SERVERAUTH_LISTENER_ADDRESS</a>	wtsc42.itso.ibm.com	20134
<input type="checkbox"/>	<a href="#">SOAP_CONNECTOR_ADDRESS</a>	wtsc42.itso.ibm.com	20020

Figure 2-12 Corrected node agent ports

- Save the changes and synchronize to the nodes. You must restart the Node Agent to pick up the port number change.

## 2.7.10 Create a personal certificate for the servant

The jobs that are generated by the zPMT tool to set up all the RACF user IDs, groups, profiles, keyrings, and certificates create a personal certificate for the controller region user ID but not for the servant region.

Our adjunct was running under the same user ID as the servant region, MDASRU. When security is enabled, and the active message engine is not in the same server, the servant needs to contact the active bus through the secure SSL port. A handshake must take place but this cannot succeed if the servant does not have a personal certificate.

We experienced the problem described in Section 4.2.17, “CWSIJ0063E: A network connection to host name x.x.x.x, port yyyyy cannot be established” on page 319 because of this. We resolved it by creating a personal certificate for the servant so it could initiate outbound SSL connections.

We created sample job MDCERT to create the certificate and connect it to the keyring of the servant region’s user ID, MDASRU. This JCL is provided in the additional material that accompanies this Redbooks publication, and is described in Appendix A, “Additional material” on page 351.

## 2.7.11 Back up the WebSphere configuration HFS datasets

Back up the deployment manager’s configuration ZFS data set and the Empty Node’s configuration ZFS data set after federating the node. We used the JCL shown in Example 2-25 on page 92, but changed the //DASD DD card to //DASD DD DSN=WASCFG.MDCELL.DUMP.POSTFED.

## 2.8 Configure the common database in DB2 for z/OS

**Note:** Skip this section if you have already generated all the DDL using createDB.sh.

We recommend that you use the createDB.sh to generate the DDL for WebSphere Process Server for z/OS because it is much easier than using DDL that is created during the execution of zWPSConfig.sh or the various wizards. The use of createDB.sh was described in 2.6, “Configure DB2 z/OS using createDB.sh” on page 79.

Nevertheless, in this section we outline how you can use the DDL generated by `zWPSConfig.sh` to create all the tables and indexes for the common components of WebSphere Process Server for z/OS. Then we describe how to execute it using `DSNTEP2`.

The steps to Configure the common database in DB2 for z/OS are as follows:

1. Create the common database.  
See page 99.
2. Copy the common DDL to z/OS sequential files.  
See page 99.
3. Edit and execute the common DDL.  
See page 100.
4. Grant table privileges to the J2C auth alias user ID.  
See page 100.

### **2.8.1 Create the common database**

We created sample job `MDCREDB` to create all the databases we needed. This included the database for the common components of WebSphere Process Server for z/OS.

You can obtain this sample JCL from Appendix A, “Additional material” on page 351.

The `MDCREDB` job is also used if you decide to generate DDL using `createDB.sh` and you can find some additional comments about it in Section 2.6.5, “Create the databases” on page 86.

### **2.8.2 Copy the common DDL to z/OS sequential files**

We created sample job `MDCPYCMN` to copy the DDL generated by `zWPSConfig.sh` into a series of z/OS sequential datasets. This sample JCL can be found in Appendix A, “Additional material” on page 351.

## 2.8.3 Edit and execute the common DDL

Perform the following steps to edit and execute the common DDL.

1. Edit the DDL while it is in the sequential files with variable record length. After editing the DDL and ensuring that all commands are within 80 columns, you can copy the DDL as members of a partitioned dataset that has an LRECL of 80. For example, we copied our customized DDL into a partitioned dataset called WASCFG.MDCELL.WPS.SQL.

The following list is a compilation of some of the changes you can expect to make to the DDL:

- Remove initial comments.
  - Split any lines longer than 80 characters so all the commands are within 80 columns.
  - Change buffer pools to match the ones you want to use.
  - Pay attention to the presence of 16 K buffer pool usage in Recovery DDL.
  - Set PRIQTY and SECQTY to -1 so that DB2 for z/OS manages physical space allocation for tablespaces and indexes.
2. Execute the DDL using DSNTEP2. We created sample job MDDDLCMN to do this. This sample JCL can be found in Appendix A, “Additional material” on page 351.

## 2.8.4 Grant table privileges to the J2C auth alias user ID

The J2C authentication alias user ID associated with the WBI\_Datasource must be granted privileges over the tables created by MDDDLCMN.

We created some GRANT commands and sample job MDGRANT to execute them. Refer to Section 2.6.8, “Grant table privileges to the J2C auth alias user ID” on page 88, where this task is described in detail following the creation of the DDL using createDB.sh.

That MDGRANT job executes GRANT statements for all components of WebSphere Process Server for z/OS, but if you did not use createDB.sh to create the DDL, you have only created the tables for the common components. You will need to comment out the GRANTxxx members on the SYSIN that refer to the other components.

## 2.9 Create a secondary node

Use the spreadsheet and zPMT to configure a secondary empty node up to the BBOMHFSB job but do not federate the node into the cell. That is, do not run the BBOWMNAN job. The node must be augmented to support WebSphere Process Server before it is federated in Section 2.9.7, “Federate the secondary empty node into the cell” on page 106.

If you plan to have a cluster with more than two cluster members, repeat the tasks in this section for each secondary node before continuing with 2.10, “Create WebSphere Process Server clusters” on page 108.

Perform the following steps to Create a secondary node.

1. Empty node: Run the zSMPInstall script.  
See page 102.
2. Empty node: Prepare ManagedDB2.rsp.  
See page 102.
3. Back up the WebSphere configuration HFS datasets.  
See page 104.
4. Empty node: Run zWPSCfg.sh.  
See page 104.
5. Back up the WebSphere configuration HFS datasets.  
See page 105.
6. Add DB2 libraries to the Servant and Adjunct JCL.  
See page 106.
7. Federate the secondary empty node into the cell.  
See page 106.
8. Update the secondary node's Node Agent ports.  
See page 107.
9. Back up the WebSphere configuration HFS datasets.  
See page 107.

## 2.9.1 Empty node: Run the zSMPInstall script

We created some JCL that executed all the necessary commands under BPXBATCH. MDNBINS. This job took about 10 minutes to run.

The JCL for MDNBINS is the same as that shown for the primary node in Example 2-23 on page 89, except that the node name must be changed from mdnodea to mdnodeb and the job is executed on the secondary node's LPAR, SC53.

## 2.9.2 Empty node: Prepare ManagedDB2.rsp

Perform the following steps to prepare ManagedDB2.rsp:

1. Copy the ManagedDB2.rsp response file from /<wps\_install\_root>/zos.config/ to a work directory. In our case, <wps\_install\_root> is /usr/lpp/zWPSMD/V6R1M2. The work directory is /var/WebSphere/home/MDCFG/wpswork.

For the secondary node of our cell named MDCELL, we named the response file mdnodebManagedDB2.rsp.

2. Customize the properties in the response file for your installation. We set up the values as shown in Example 2-24 on page 91. The user ID running the configuration jobs must have read access to this file.

The supplied ManagedDB2.rsp was missing the property DBJDBCCLASSPATH=/usr/lpp/db2/d9fg/db2910\_jdbc/classes so we added that by copying it from the mddmnodeDmgrDB2.rsp.

Note that the templatePath is split across two lines in Example 2-28 on page 103, but is coded on one line in the response file.

**Note:** If you try to save time by copying the response file that you used for the primary node, make sure you change all the necessary fields. In addition to changing the node name, you must change the cell name to the one used for the empty node on the new LPAR.



*Example 2-28 mdnodebManagedDB2.rsp for the second node*

---

```
JMSUSER=MDJMSU
JMSPASS=MDJMSU
DBPRODUCT=DB2UDBOS390_V9_1
DBLOCATION=DB9F
DBPROPERTIES=/etc/d9fg
DBJDBCCLASSPATH=/usr/lpp/db2/d9fg/db2910_jdbc/classes
DBUSER=MDDBU
DBPASS=MDDBU
DBHOSTNAME=wtsc53.itso.ibm.com
DBSERVERPORT=37893
SQLDB=MDWPSDB
SQLSTO=MDWPSSG
augment
profileName=default
profilePath=/wasv61config/mdcell/mdnodeb/AppServer/profiles/default
templatePath=/wasv61config/mdcell/mdnodeb/AppServer/profileTemplates/\
managed.wbiserver
cellName=mdemptyb
nodeName=mdnodeb
dbCreateNew=false
dbDelayConfig=true
configureScaSecurity=true
scaSecurityUserId=$JMSUSER
scaSecurityPassword=$JMSPASS
configureAppScheduler=false
dbName=$SQLDB
dbStorageGroup=$SQLSTO
dbType=$DBPRODUCT
dbConnectionLocation=$DBLOCATION
dbJDBCProperties=$DBPROPERTIES
dbJDBCClasspath=$DBJDBCCLASSPATH
dbUserId=$DBUSER
dbPassword=$DBPASS
dbHostName=$DBHOSTNAME
dbServerPort=$DBSERVERPORT
```

---

### 2.9.3 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and the Empty Node's configuration ZFS data set before running zWPSCfg.sh against the Empty Node. We used the JCL shown in Example 2-29.

*Example 2-29 Sample JCL to dump the DMGR and both nodes*

---

```
//MDDUMP JOB NOTIFY=&SYSUID,CLASS=A,MSGCLASS=A,
// REGION=OM,TIME=1440
//*
//* DUMP CONFIG HFS FOR DMGR AND BOTH NODES
//*
//DUMP EXEC PGM=ADRDSSU
//DASD DD DISP=(NEW,CATLG),DSN=WASCFG.MDCCELL.DUMP.PRECFGN,
// UNIT=3390,SPACE=(CYL,(1000,100),RLSE),VOL=SER=TST052
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DUMP DATASET( -
INCLUDE( -
WASCFG.MDCCELL.MDDMNODE.CONFIG.ZFS -
WASCFG.MDCCELL.MDNODEA.CONFIG.ZFS -
WASCFG.MDCCELL.MDNODEB.CONFIG.ZFS -
)
) -
OUTDD(DASD) -
TOL(ENQF) OPTIMIZE(4) COMPRESS
/*
```

---

### 2.9.4 Empty node: Run zWPSCfg.sh

After preparing the response file, we executed the shell script zWPSCfg.sh by submitting job MDNBCFG, as shown in Example 2-30 on page 105. This job took about three minutes to run.

The job ran under the WebSphere Administrator user ID to ensure that the updates can be made to the node's configuration. You must ensure that the response file (in /var/WebSphere/home/MDCFG/wpswork/mdnodebManagedDB2.rsp in Example 2-14 on page 77) is readable by the WebSphere Administrator user ID.

Example 2-30 MDNBCFG JCL to run zWPSConfig for empty node mdnodea

---

```
//MDNBCFG JOB (0),CLASS=A,MSGCLASS=A,REGION=OM,
// USER=MDADMIN,PASSWORD=MDADMIN,
// NOTIFY=SENIOR
/*JOBPARM SYSAFF=SC53
/*****
/* Configure WPS in empty node mdnodea */
/*****
/*****
/* Run zWPSConfig.sh */
/*****
//INSTO EXEC PGM=IKJEFT01,REGION=OM,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
    cd /wasv61config/mdcell/mdnodeb/AppServer/bin; +
    ./zWPSConfig.sh +
    -response /var/WebSphere/home/MDCFG/wpswork/+
    mdnodebManagedDB2.rsp +
    -augment +
    1> /tmp/mdnodebzWPSConfig_0511.out +
    2> /tmp/mdnodebzWPSConfig_0511.err;
/*
/*****
/* STEPS TO COPY THE OUTPUT THE JOB LOG */
/*****
//CPOUT EXEC PGM=IKJEFT01,REGION=OM
//SYSEXEC DD DISP=SHR,DSN=BBO6153.MDNODEB.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
    BBOHFSWR '/tmp/mdnodebzWPSConfig_0511.out'
    BBOHFSWR '/tmp/mdnodebzWPSConfig_0511.err'
/*
//
```

---

## 2.9.5 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and both of the the node configuration ZFS data sets before federating the secondary node.

We used the JCL shown in Example 2-29 on page 104, but changed the //DASD DD card to //DASD DD DSN=WASCFG.MDCCELL.DUMP.PREFEDB.

## 2.9.6 Add DB2 libraries to the Servant and Adjunct JCL

WebSphere Process Server accesses DB2 for z/OS using a Type 2 datasource, and the DB2 load libraries must be added to the STEPLIB of the servant regions (including the adjunct) if they are not defined in the system LINKLST.

1. Update the following members of SYS1.PROCLIB:
  - MDMGSRZ
  - MDCRAAZ
  - MDCRABZ
  - MDASRAZ
  - MDASRBZ
2. Add the highlighted SDSN\* libraries shown in Example 2-31:

*Example 2-31 SDSN libraries*

---

```
//STEPLIB DD DISP=SHR,DSN=BB46142.MDDMNODE.SBBOLD2
//          DD DISP=SHR,DSN=BB46142.MDDMNODE.SBBLOAD
//          DD DISP=SHR,DSN=BB46142.MDDMNODE.SBBGLOAD
//          DD DISP=SHR,DSN=DB9F9.SDSNEXIT
//          DD DISP=SHR,DSN=DB9F9.SDSNLOAD
//          DD DISP=SHR,DSN=DB9F9.SDSNLOD2
```

---

Note that the customized SDSNEXIT DD statement must occur above the SDSNLOAD in the concatenation.

3. Check that all the DB2 libraries are APF authorized by issuing the MVS command **D PROG,APF**.

Any libraries added to the WebSphere STEPLIBs must be authorized by APF and program-controlled by RACF. It is common to find that SDSNLOAD and SDSNEXIT were APF authorized when the DB2 subsystem was configured but that SDSNLOD2 was not.

## 2.9.7 Federate the secondary empty node into the cell

Perform the following steps to federate the secondary empty node into the cell.

1. Return to the CNTL dataset that contains the jobs for configuring the secondary empty node. Locate the BBOMNAN job, and edit it.
2. Add the user ID and password of the WebSphere administrator user to the job card. For example, the job card when we federated our primary node into the MDCELL was as shown in Example 2-32 on page 107.

*Example 2-32 Job card when we federated our primary node into the MDCELL*

```
//BBOWMNAN JOB (0),'WPS SETUP',CLASS=A,REGION=OM,  
//          MSGCLASS=H,NOTIFY=&SYSUID,  
//          USER=MDADMIN,PASSWORD=MDADMIN  
/*JOBPARM SYSADFF=SC53
```

**Note:** Use a /\*JOBPARM SYSAFF card to ensure that the BBOWMNAN job runs on the LPAR, which is the home of the secondary node you are federating.

If you chose to start the node agent after federation, the BBOWMNAN job will issue a start command for the node agent. Therefore, BBOWMNAN must run on the LPAR where the secondary node's node agent is started.

### 3. Submit BBOMNAN.

If you chose to start the node agent during federation, the node agent is started by job BBOWMNAN just before it ends.

If there are problems with the job and you submitted it using the cell's administrator user ID, check the logs of the deployment manager controller and servant for error messages and the wsadmin.traceout file in the node's <was\_home>/profiles/default/logs

## 2.9.8 Update the secondary node's Node Agent ports

As with the primary node, some of the ports belonging to the secondary node's node agent are assigned default values. Set these to the port numbers that were planned for the cluster in the spreadsheet, as described in Section 2.7.9, "Update the primary node's Node Agent ports" on page 95.

## 2.9.9 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and both of the the node configuration ZFS data sets before federating the node.

We used the JCL shown in Example 2-29 on page 104, but changed the //DASD DD card to //DASD DD DSN=WASCFG.MDCELL.DUMP.POSTFEDB.

## 2.10 Create WebSphere Process Server clusters

Creating a cluster is easier in WebSphere Application Server for z/OS V6.1 than it was in V6.0 because the wizard prompts you for the Cluster short name and for the short names of the cluster members. This means you no longer have to manually update the cluster and server configurations to set the correct short names. Nevertheless, in this section we show you how to review the cluster after it has been created so you become familiar with the panels where all the WebSphere Process Server for z/OS configuration wizards are located.

At this stage your target topology becomes important.

We created one cell with prefix MD for a single-cluster topology. Our single cluster topology used a cluster called mdcl01\_WPS.

In a second cell called MP we created the two-clusters topology. This used one cluster for Support and Messaging functions, called mpcl02.WPS\_SandM and a second cluster for application functions called mpcl02.AppTarget.

There are few differences between these two scenarios except when it comes to choosing the deployment target for resources or applications. Where the two-clusters topology differs from the single-cluster topology, those differences are high-lighted.

If you decide to configure some other topology then you will need to adapt these instructions to deploy the various components of WebSphere Process Server for z/OS in the correct cluster.

Perform the following steps to Create WebSphere Process Server clusters.

1. Creating a cluster.  
See page 109.
2. Define additional clusters.  
See page 115.
3. Familiarize yourself with the cluster.  
See page 115.
4. Update the server's ports.  
See page 117.
5. Back up the WebSphere configuration HFS datasets.  
See page 119.

## 2.10.1 Creating a cluster

Perform the following steps to create a cluster.

1. Log in to the WebSphere Integrated Solutions Console and navigate to **Servers** → **Clusters**.
2. Click **New** to launch the wizard as shown in Figure 2-13.



Figure 2-13 Create a new cluster

3. In the Step 1: Enter basic cluster information window of the wizard (Figure 2-14), enter the cluster long name and the cluster short name.

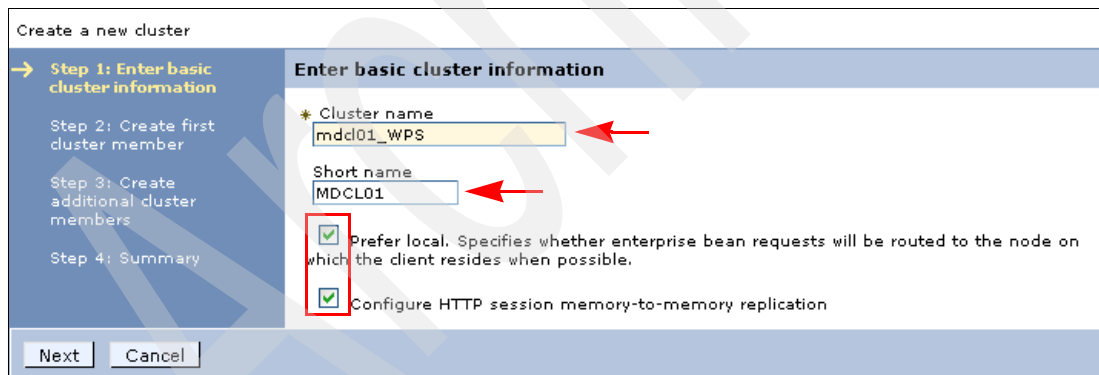


Figure 2-14 Enter the names for the cluster and check both boxes

Our convention was to use lowercase names for long names and uppercase names for short names. In order to make the purpose of the cluster clearer when looking at topology through the Integrated Solutions Console, we added a descriptive string (for example, \_WPS or SandM) to the long name of clusters and servers. The names you choose at this point will depend to some extent on the topology you have chosen to configure.

When we deployed across two clusters we decided to choose the cluster names shown in Table 2-4.

*Table 2-4 Cluster names when deployed across two clusters*

Cluster Usage	Short name	Long name
WebSphere Process Server Support and Messaging cluster	MPCL01	mpcl01.WPS_SandM
BPC, Fabric and Application cluster	MPCL02	mpcl02.AppTarget

For the two clusters topology, we created the first cluster, mpcl01.WPS\_SandM, in this step. In Section 2.10.2, “Define additional clusters” on page 115, we created the second cluster, mpcl02.AppTarget.

4. Select the Prefer local and Configure HTTP Sessions check boxes, as shown in Figure 2-14 on page 109. It is easier to set up HTTP replication when creating the cluster than it is to do it later.
5. In the Step 2: Create first cluster member window of the wizard (Figure 2-15 on page 111), enter the long and short names of the first cluster member.

Be sure to choose the correct node from the Select node list box, as shown in Figure 2-15 on page 111.

It does not matter whether you select the Generate unique HTTP ports check box or not because you will set all the correct port in the cluster members later in Section 2.10.4, “Update the server’s ports” on page 117.

6. Click **Create the member using an application server template**. Choose **defaultProcessServerOS** from the menu. This is important because if you do not do this the cluster panels will not contain all the options to configure WebSphere Process Server for z/OS.

When the panel is complete, click **Next**.



Create a new cluster

Step 1: Enter basic cluster information

→ Step 2: Create first cluster member

Step 3: Create additional cluster members

Step 4: Summary

Create first cluster member

The first cluster member determines the server settings for the cluster members. A server configuration template is created from the first member and stored as part of the cluster data. Additional cluster members are copied from this template.

\* Member name  
mdsr01a\_WPS

Select node  
mdnodea(ND 6.1.0.17)

Short name  
MDSR01A

\* Weight  
2 (0..20)

☐ Generate unique HTTP ports

Select basis for first cluster member:

☒ Create the member using an application server template.  
defaultESBServerZOS  
defaultESBServerZOS  
defaultProcessServerZOS  
defaultZOS

☐ Existing application server as a template.

☐ Create the member by converting an existing application server.  
(none)

☐ None. Create an empty cluster.

Previous

Next

Cancel

Figure 2-15 Enter the names of the first cluster member

7. In the Step 3: Create additional cluster members window of the wizard, notice that the first cluster member is listed in the table at the bottom of the panel.

Enter the long and short names of the second cluster member.

Be sure you choose the correct node from the drop-down list box.

As for the first cluster member, it does not matter whether you select the Generate unique HTTP ports check box because you will set the correct port numbers later.

Click **Add member** as shown in Figure 2-16.

Create a new cluster

Step 1: Enter basic cluster information

Step 2: Create first cluster member

→ Step 3: Create additional cluster members

Step 4: Summary

### Create additional cluster members

Enter information about this new cluster member, and click Add Member to add this cluster member to the member list. A server configuration template is created from the first member and stored as part of the cluster data. Additional cluster members are copied from this template.

\* Member name  
mdsr01b\_WPS

Select node  
mdnodeb(ND 6.1.0.17)

Short name  
MDSR01B

\* Weight  
2 (0..20)

☐ Generate unique HTTP ports

**Add Member**

Use the Edit function to edit the properties of a cluster member that is already included in this list. Use the Delete function to remove a cluster member from this list. You are not allowed to edit or remove the first cluster member or an already existing cluster member.

Edit Delete

Select	Member name	Nodes	Version	Weight
	mdsr01a_WPS	mdnodea	Business Process Choreographer 6.1.2.0 ND 6.1.0.17 WPS 6.1.2.0	2

Previous Next Cancel

Figure 2-16 Enter the names of the second cluster member

The wizard presents another panel giving you the opportunity to define a third cluster member. At the bottom of the panel there are two cluster members listed, as shown in Figure 2-17 on page 113.

Create a new cluster

Step 1: Enter basic cluster information

Step 2: Create first cluster member

→ Step 3: Create additional cluster members

Step 4: Summary

Create additional cluster members

Enter information about this new cluster member, and click Add Member to add this cluster member to the member list. A server configuration template is created from the first member and stored as part of the cluster data. Additional cluster members are copied from this template.

\* Member name

Select node

mdnodeb(ND 6.1.0.17)

Short name

\* Weight

2

(0..20)

☐ Generate unique HTTP ports

Add Member

Use the Edit function to edit the properties of a cluster member that is already included in this list. Use the Delete function to remove a cluster member from this list. You are not allowed to edit or remove the first cluster member or an already existing cluster member.

Edit

Delete

☒

☐

Select	Member name	Nodes	Version	Weight
<input type="checkbox"/>	mdsr01a_WPS	mdnodea	Business Process Choreographer 6.1.2.0 ND 6.1.0.17 WPS 6.1.2.0	2
<input type="checkbox"/>	mdsr01b_WPS	mdnodeb	Business Process Choreographer 6.1.2.0 ND 6.1.0.17 WPS 6.1.2.0	2

Previous

Next

Cancel

Figure 2-17 Two cluster members listed

- Verify that the cluster members are in the correct nodes.

Because we had only two nodes configured, no more cluster members were needed. Click **Next**.

**Note:** Section 1.1.4, “WebSphere Process Server for z/OS scalability and availability” on page 5 explained that a production topology ideally will have clusters with at least three cluster members (the rule of three). If you followed that advice you would add another cluster member at this point by clicking **Add Member** again.

9. In the Step 4: Summary window of the wizard, shown in Figure 2-18, review the data entered into the wizard for accuracy. Click **Finish**.

Create a new cluster

Step 1: Enter basic cluster information

Step 2: Create first cluster member

Step 3: Create additional cluster members

→ Step 4: Summary

Summary

Summary of actions:

Options	Values
Cluster Name	mdd01_WPS
Short name	MDCL01
Core Group	DefaultCoreGroup
Node group	DefaultNodeGroup
Prefer local	true
Configure HTTP session memory-to-memory replication	true
Server name	mdsr01a_WPS
Node	mdnodea(Business Process Choreographer 6.1.2.0 ND 6.1.0.17 WPS 6.1.2.0)
Short name	MDSR01A
Weight	2
Clone Template	defaultProcessServerZOS
Clone Type	default
Generate unique HTTP ports	false
Server name	mdsr01b_WPS
Node	mdnodeb(Business Process Choreographer 6.1.2.0 ND 6.1.0.17 WPS 6.1.2.0)
Short name	MDSR01B
Weight	2
Clone Template	defaultProcessServerZOS
Clone Type	default
Generate unique HTTP ports	true

Previous

Finish

Cancel

Figure 2-18 Review the summary

10. Save the configuration changes and synchronize to the nodes.

## 2.10.2 Define additional clusters

**Note:** If you are configuring the single-cluster topology, skip this step.

If you intend to deploy WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS in the two-clusters topology, then in Section 2.10.1, “Creating a cluster” on page 109 you just configured the Support and Message cluster. Repeat the tasks in Section 2.10.1, “Creating a cluster” on page 109 to create the AppTarget cluster.

## 2.10.3 Familiarize yourself with the cluster

Perform the following steps to familiarize yourself with the cluster.

1. Navigate to **Servers** → **Clusters** (Figure 2-19) and click the cluster link.

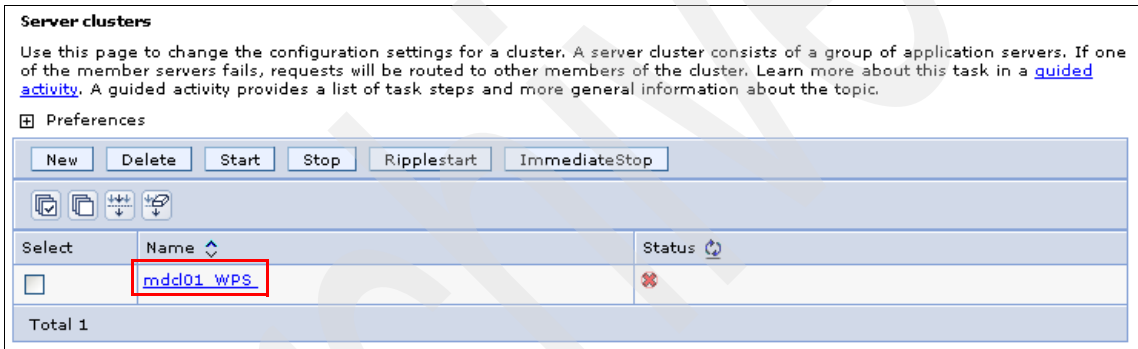


Figure 2-19 Click the link to review the cluster configuration

Figure 2-20 shows that the cluster short name has been set already.

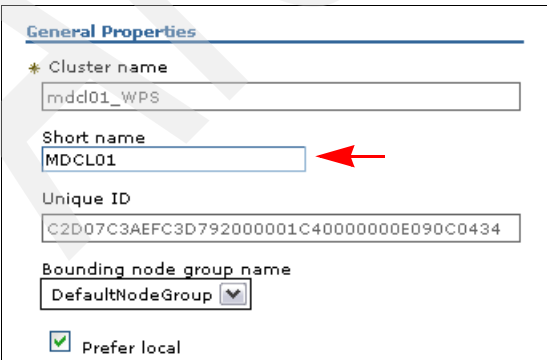


Figure 2-20 Check the cluster short name

- Under Additional properties, expand **Cluster members** to see the two cluster members (Figure 2-21). Click **Details** to see a panel that provides links to each cluster member. From here there are options to alter the weighting of the workload to each cluster member. However on z/OS this weighting is not used for workload balancing so leave the weightings equal.

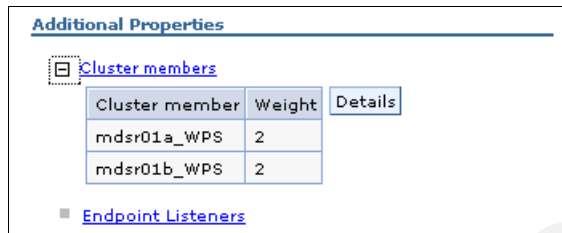


Figure 2-21 Review the cluster members

Note the location of the BPC and CEI configuration wizards in the section headed **Business Integration** (Figure 2-22).



Figure 2-22 Business Integration configuration on the cluster panel

- Click the Topology tab and expand the tree to view the cluster topology. It will look like Figure 2-23.

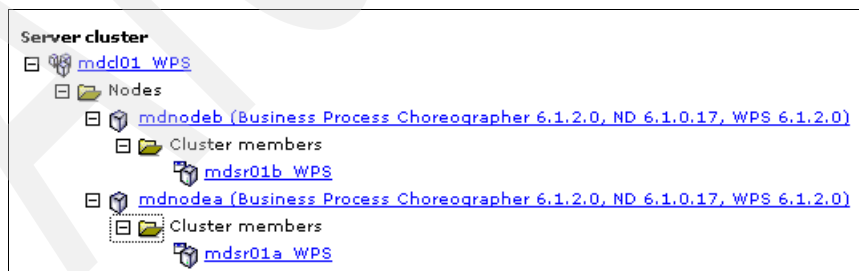


Figure 2-23 Review the cluster topology

## 2.10.4 Update the server's ports

Perform the following steps to update the server's ports.

1. Navigate to **Servers** → **Application Servers** to list the cluster members, as shown in Figure 2-24.

**Application servers**

Use this page to view a list of the application servers in your environment and the status of each of these servers. You can also use this page to change the status of a specific application server.

⊞ Preferences

New Delete Templates... Start Stop ImmediateStop Terminate

Select	Name ↕	Node ↕	Version ↕	Cluster Name ↕	Status ↕
<input type="checkbox"/>	<a href="#">mdsr01a_WPS</a>	mdnodea	Business Process Choreographer 6.1.2.0 ND 6.1.0.17 WPS 6.1.2.0	mdcl01_WPS	✖
<input type="checkbox"/>	<a href="#">mdsr01b_WPS</a>	mdnodeb	Business Process Choreographer 6.1.2.0 ND 6.1.0.17 WPS 6.1.2.0	mdcl01_WPS	✖

Total 2

Figure 2-24 List the cluster members

2. Click the link to one of the servers and then click **Ports** under **Communications**. You will see that default numbers have been assigned.

The ports can be changed through the Integrated Solutions Console but it is easier and less error-prone to use a script. Techdoc *Creating new Application Servers in WAS V6.1 for z/OS*, TD10466 provides a jython script called `updNewServer.py` that allows you to specify a starting port range and then assigns port numbers according to the convention in the spreadsheets provided in Techdoc PRS131. You can obtain the `updNewServer.py` at the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TD104066>

3. Create some JCL to invoke the updNewServer.py for each of the cluster members. Sample job MDNAPORT, which updates the ports of server mdsr01a\_WPS in node mdnodea, is shown in Example 2-33.

*Example 2-33 Sample JCL MDNAPORT to update the ports of a server*

---

```
//MDNAPORT JOB (0),'MDADMIN',CLASS=A,MSGCLASS=H,REGION=OM,
// NOTIFY=SENIOKJ,USER=MDADMIN,PASSWORD=MDADMIN
/*JOBPARM SYSAFF=SC42
//*****
/* Run this job using an administrator userid. */
//*****
//UPDPORT EXEC PGM=IKJEFT01,REGION=OM,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
  export PATH=.:$PATH; +
  cd /wasv61config/mdcell/mdnodea/AppServer; +
  cd profiles/default/bin; +
  wsadmin.sh -user MDADMIN -password MDADMIN +
  -lang jython +
  -f /var/WebSphere/home/MDCFG/wpswork/updNewServer.py +
  mdsr01a_WPS mdnodea 20040 +
  1> /tmp/updNewServer_1000.out +
  2> /tmp/updNewServer_1000.err ;
//*****
/* STEPS TO COPY THE OUTPUT THE JOB LOG */
//*****
//CPOUT EXEC PGM=IKJEFT01,REGION=OM
//SYSEXEC DD DISP=SHR,DSN=BB06142.MDNODEA.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
  BBOHFSWR '/tmp/updNewServer_1000.out'
  BBOHFSWR '/tmp/updNewServer_1000.err'
```

---



Unfortunately the updNewServer.py does not update all the ports. We used the Integrated Solutions Console to update the CSiv2 and SAS SSL ports as shown in Figure 2-25.

**Application servers > mdsr01a.WPS > Ports**

Specifies the TCP/IP ports this server uses for connections.

Preferences

New Delete

Select	Port Name	Host	Port
<input type="checkbox"/>	<a href="#">BOOTSTRAP ADDRESS</a>	wtsc42.itso.ibm.com	20041
<input type="checkbox"/>	<a href="#">CSIV2 SSL MUTUALAUTH LISTENER ADDRESS</a>	wtsc42.itso.ibm.com	20054
<input type="checkbox"/>	<a href="#">CSIV2 SSL SERVERAUTH LISTENER ADDRESS</a>	wtsc42.itso.ibm.com	20055
<input type="checkbox"/>	<a href="#">DCS UNICAST ADDRESS</a>	wtsc42.itso.ibm.com	20043
<input type="checkbox"/>	<a href="#">ORB LISTENER ADDRESS</a>	*	20041
<input type="checkbox"/>	<a href="#">ORB SSL LISTENER ADDRESS</a>	*	20042
<input type="checkbox"/>	<a href="#">SAS SSL SERVERAUTH LISTENER ADDRESS</a>	wtsc42.itso.ibm.com	20056
<input type="checkbox"/>	<a href="#">SIB ENDPOINT ADDRESS</a>	*	20050

Figure 2-25 Update CSiv2 and SAS ports

4. Save the changes and synchronize to the nodes.
5. If you have defined more than one cluster, repeat these steps for the servers in the second cluster using a different port range.

The planning spreadsheet described in Section 2.2.5, “Planning using a spreadsheet” on page 33 contains port assignments for two clusters.

For our two-clusters topology, the servers in cluster mpcl01.WPS\_SandM used ports starting from 22040, while those in the mpcl02.AppTarget cluster used ports starting from 22060.

## 2.10.5 Back up the WebSphere configuration HFS datasets

Back up the deployment manager’s configuration ZFS data set and both of the node configuration ZFS data sets after creating the cluster.

We used the JCL shown in Example 2-29 on page 104 but changed the //DASD DD card to //DASD DD DSN=WASCFG.MDCELL.DUMP.POSTCLUS.

## 2.11 Define HTTP servers or proxies

When you have a cluster, it is usually necessary to define one or more HTTP Servers, or some kind of proxy, in front of the cluster. The HTTP Servers run the WebSphere plugin, which is responsible for distributing work and maintaining affinity between a client and a cluster member. Alternatively, you can use the On-Demand Router (ODR) that is part of WebSphere Extended Deployment for z/OS Virtual Enterprise, or perhaps use some other third-party proxy. IBM White Paper *WebSphere z/OS - Comparing Front End HTTP Options*, PRS2663, will help you understand the differences between these solutions. You can obtain that paper from the following Web page:

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

When you run HTTP servers on z/OS you have a choice between the older IBM HTTP Server for z/OS (based on Lotus® Domino® Go technology) and the newer IBM HTTP Server (based on Apache). You can also choose to run the HTTP Server on some other platform. We configured the IBM HTTP Server for z/OS based on Lotus Domino Go and we refer to this HTTP Server as “IBM HTTP Server for z/OS (LGW)” hereafter.

HTTP Servers were configured on the LPARs hosting the nodes of our MD and MP cells. We do not include step-by-step instructions on configuring an HTTP Server in this Redbooks publication. Instead, we describe the two z/OS HTTP Server options briefly, and provide references to existing step-by-step guides for each of these options.

### 2.11.1 IBM HTTP Server for z/OS powered by Apache

The newer HTTP Server is shipped as one component of the WebSphere Application Server for z/OS optional materials and is usually found in the file system at `/usr/lpp/zWebSphere_OM/V6R1/HTTP/Server`. The IBM HTTP Server for z/OS Powered by Apache is FMID JIWO610, Component IDs 5655I3510 and 5655I3511.

IBM White Paper WP101170, *The IBM HTTP Server for z/OS Powered by Apache*, describes how to configure it. You can download that White Paper from the following Web page:

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

The service level of the HTTP Server powered by Apache needs to be kept in step with the underlying WebSphere Application Server for z/OS. You can find details of the corresponding service levels at the following Web page:

<http://www.ibm.com/support/docview.wss?&uid=swg27009131>

### 2.11.2 IBM HTTP Server for z/OS (LGW)

The IBM HTTP Server for z/OS (LGW) has been shipped as part the z/OS base operating system for many years. There are three FMIDs, HIMW530, JIMW53B and JIMW531, and it is typically found in the file system at `/usr/lpp/internet`.

You can find a step-by-step guide to configuring the IBM HTTP Server for z/OS (LGW) in IBM Redpaper *WebSphere Process Server for z/OS: Configuring a Network Deployment Environment*, REDP-4388. Although that paper discusses WebSphere Process Server for z/OS V6.0.2, the instructions are largely the same for V6.1.2.

## 2.12 Service Component Architecture

This section describes the steps to generate and execute the DDL that sets up the tables for Service Component Architecture (SCA) in DB2 for z/OS if that was not done using `createDB.sh` in Section 2.6, “Configure DB2 z/OS using `createDB.sh`” on page 79. Then it describes how to use the Service Component Architecture configuration wizard that is located under Business Integration on the cluster configuration panel shown in Figure 2-26 on page 125.

Perform the following steps to generate and execute the DDL:

1. Generate DDL for all SIBs using `sibDDLGenerator.sh`.  
See page 122.
2. Edit and execute the SIB DDL.  
See page 124.
3. Run the SCA wizard.  
See page 125.
4. Verify SCA Modules and SCA runtime.  
See page 128.
5. Back up the WebSphere configuration HFS datasets.  
See page 128.

## 2.12.1 Generate DDL for all SIBs using sibDDLGenerator.sh

**Note:** Skip this step if you already created all the DDL using the createDB.sh.

If you did not use createDB.sh to create and execute the SIB DDL, you can run the sibDDLGenerator.sh to generate DDL for all the SIBs you will use. It is a good idea to prepare the DDL and execute it before using the wizards or scripts to configure the buses that are used by SCA, CEI, and BPC.

Generate the SIB DDL. We created sample job MDSIBGEN (Example 2-34) to execute sibDDLGenerator.sh so it generated DDL for four SIBs: two for SCA, one for BPC, and one for CEI. The generated DDL was then copied to z/OS sequential files where it can be reviewed and edited with the ISPF editor.

### *Example 2-34 Sample job MDSIBGEN to generate SIB DDL*

```
//MDSIBGEN JOB (0),'MDADMIN',CLASS=A,REGION=OM,NOTIFY=&SYSUID
/*JOBPARM S=SC42
/*****
/* Make sure you edit this file with CAPS OFF. */
/* Run this job using an administrator userid or UID(0) */
/*****
/*****
/* Run sibDDLGenerator.sh to create DLL for 4 SIBs */
/*****
//SIBGEN EXEC PGM=IKJEFT01,REGION=OM,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
cd /wasv61config/mdcell/mdnodea/AppServer/profiles/default/bin/; +
sibDDLGenerator.sh +
-system db2 -version 8.1 -platform zos -schema MD01A +
-user MDDBU -create -database MD01ADB -storagegroup MDWPSSG +
-bufferpool BP3 +
-statementend ";" +
1> /tmp/MDSIBAPP.ddl +
2> /tmp/sibDDLGen_APP.err; +
sibDDLGenerator.sh +
-system db2 -version 8.1 -platform zos -schema MD01B +
-user MDDBU -create -database MD01BDB -storagegroup MDWPSSG +
-bufferpool BP3 +
-statementend ";" +
1> /tmp/MDSIBBPC.ddl +
2> /tmp/sibDDLGen_BPC.err; +
sibDDLGenerator.sh +
-system db2 -version 8.1 -platform zos -schema MD01C +
-user MDDBU -create -database MD01CDB -storagegroup MDWPSSG +
-bufferpool BP3 +
```

```

        -statementend ";" +
1> /tmp/MDSIBCEI.ddl +
2> /tmp/sibDDLGen_CEI.err; +
sibDDLGenerator.sh +
-system db2 -version 8.1 -platform zos -schema MD01S +
-user MDDBU -create -database MD01SDB -storagegroup MDWPSSG +
-bufferpool BP3 +
        -statementend ";" +
1> /tmp/MDSIBSCA.ddl +
2> /tmp/sibDDLGen_SCA.err;
/*
/*****
/*  STEPS TO COPY THE OUTPUT THE JOB LOG                                */
/*****
//CPOUT EXEC PGM=IKJEFT01,REGION=OM
//SYSEXEC DD DISP=SHR,DSN=BB06142.MDDMNODE.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
        BBOHFSWR '/tmp/sibDDLGen_APP.err'
        BBOHFSWR '/tmp/sibDDLGen_BPC.err'
        BBOHFSWR '/tmp/sibDDLGen_CEI.err'
        BBOHFSWR '/tmp/sibDDLGen_SCA.err'
/*
/*****
/*  COPY DDL from UNIX to zOS                                            */
/*****
//COPY EXEC PGM=IKJEFT01,REGION=OM,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
OGET '/tmp/MDSIBAPP.ddl' 'WASCFG.MDCELL.WPS.SQL.MD01A'
OGET '/tmp/MDSIBBPC.ddl' 'WASCFG.MDCELL.WPS.SQL.MD01B'
OGET '/tmp/MDSIBCEI.ddl' 'WASCFG.MDCELL.WPS.SQL.MD01C'
OGET '/tmp/MDSIBSCA.ddl' 'WASCFG.MDCELL.WPS.SQL.MD01S'
/*
/*  Clean up /tmp
//DELTMP EXEC PGM=IKJEFT01,REGION=OM,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
        rm /tmp/MDSIBAPP.ddl; +
        rm /tmp/MDSIBBPC.ddl; +
        rm /tmp/MDSIBCEI.ddl; +
        rm /tmp/MDSIBSCA.ddl; +
        rm /tmp/sibDDLGen_APP.err; +
        rm /tmp/sibDDLGen_BPC.err; +
        rm /tmp/sibDDLGen_CEI.err; +
        rm /tmp/sibDDLGen_SCA.err;
/*

```

---

## 2.12.2 Edit and execute the SIB DDL

Perform the following steps to edit and execute the SIB DDL.

**Note:** Skip this step if you have already created all the DDL using the createdB.sh and executed it.

1. Review the generated DDL and edit it if required. The SIB DDL is fully qualified so you do not need to be careful about setting the correct SQLID before executing the DDL.
2. After editing it, copy the DDL from the sequential files into a partitioned dataset with LRECL=80 to execute it using DSNTEP2
3. Execute the SIB DDL. We created sample job MDDDLISIB, shown in Example 2-35, to run DSNTEP2 and execute the SIB DDL.

*Example 2-35 Sample job MDDDLISIB to execute the SIB DDL*

---

```
//MDDDLISIB JOB (0), 'MDADMIN', CLASS=A, REGION=OM, NOTIFY=&SYSUID
/*JOBPARM S=SC42
/*-----*/
/* This job executes SIB DDL generated by the shell */
/* script sibDDLGenerator.sh. */
/* See job MDDDL for JCL that executes SIB DDL generated */
/* by createdB.sh. */
/*-----*/
//JOB LIB DD DISP=SHR, DSN=DB9F9.SDSNEXIT
// DD DISP=SHR, DSN=DB9F9.SDSNLOAD
/*
/* The userid on the SET CURRENT SQLID statement must
/* have DB2 SYSADM authority and EXECUTE on PLAN DSNTEP2
/*
//CREATE EXEC PGM=IKJEFT01, DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//CREATE EXEC PGM=IKJEFT01, DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
DSN SYSTEM(D9FG)
RUN PROGRAM(DSNTEP2) PLAN(DSNTEP91) LIB('DB9FU.RUNLIB.LOAD')
//SYSIN DD *
// DD DISP=SHR, DSN=WASCFG.MDCELL.WPS.SQL(MD01A)
// DD DISP=SHR, DSN=WASCFG.MDCELL.WPS.SQL(MD01B)
// DD DISP=SHR, DSN=WASCFG.MDCELL.WPS.SQL(MD01C)
// DD DISP=SHR, DSN=WASCFG.MDCELL.WPS.SQL(MD01S)
```

---

**Note:** The SIB DDL generated by sibDDLGenerator.sh also includes GRANT statements that grant table privileges on the SIB tables to the J2C authentication alias user ID. Therefore, unlike other components of WebSphere Process Server for z/OS, there is no need to issue separate GRANT statements for SIB tables,

### 2.12.3 Run the SCA wizard

Perform the following steps to run the SCA wizard.

1. Log in to the WebSphere Administration console and navigate to **Servers** → **Clusters**.

**Two-clusters topology:** When you are configuring the two-clusters topology, SCA is defined in the Support and Messaging cluster. In our MP cell, for example, we selected the cluster called mpcl02.WPS\_SandM before launching the SCA configuration wizard.

2. Click the link to the cluster. Then, under the Business Integration head, click **Service Component Architecture** (Figure 2-26).

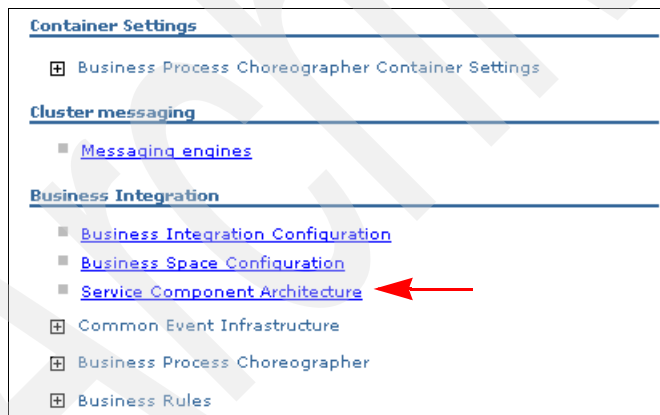


Figure 2-26 Launch the SCA wizard

3. Select the Support the Service Component Architecture components check box and select the Local radio button (Figure 2-27). Selecting Local will causes the SCA buses to be configured in the local cluster.

**General Properties**

☒ Support the Service Component Architecture components

**Bus Member Location**

☒ Local

☐ Remote

New

Figure 2-27 Check the box and select a Local Bus Member location

4. Change the schema values for the System SCA and Application SCA buses.

Further down the panel are two tables that contain the settings for the data stores used by the System SCA and Application SCA buses. Most of the fields already contain the values you specified in the `mddmnodeDmgrDB2.rsp` file, but you will probably need to change the schema values for these buses.

According to our naming convention, the schema of the System bus is MD01S and the schema of the application bus is MD01A, so we set those values (Figure 2-28). When you have the two-clusters topology, the SCA buses in the Support and Messaging cluster are Local.

**System Bus Member**

System bus destinations support the asynchronous communication of Service Oriented Architecture applications and components with each other.

Edit ... Test Connection

Database Instance	Schema	Create Tables	User name	Password	Server
DB9F	MD01S	<input type="checkbox"/>	MDDBU	*****	wtsc42.itso.ibm.com

**Application Bus Member**

Application bus destinations support the asynchronous communication of WebSphere Business Integration Adapter Architecture components.

☒ Enable the WebSphere Business Integration Adapter components

Edit ... Test Connection

Database Instance	Schema	Create Tables	User name	Password	Server
DB9F	MD01A	<input type="checkbox"/>	MDDBU	*****	wtsc42.itso.ibm.com

Figure 2-28 Setting the SCA data store properties including the schemas



- Click the **Edit ...** button inside each table to set the storage group name and buffer pools. These values are used to customize DDL.

**Note:** The Database Instance field in Figure 2-34 is shown set to the DB2 z/OS Location name. This is correct when you want to create type 2 data sources that connect to DB2 for z/OS. In WebSphere Process Server for z/OS the Database name field of type 2 data sources is set to the DB2 Location name rather than the Database name.

However, specifying the DB2 Location name in the Database instance field of the wizard means that the wizard will customize SCA bus DDL with DB9F as the database name. If you have not yet created the DDL using createDB.sh, you may prefer to have the wizard customize the SCA DDL with the correct database name, and then change the Database name on the data sources to the DB2 subsystem's Location name.

After completing the panel, taking particular care to set the correct schema names, click **OK** and then save the configuration changes.

- Expand Service Integration and click **Buses** (Figure 2-29).

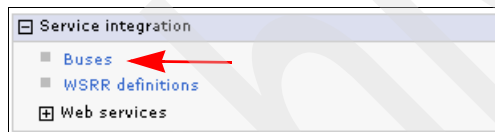


Figure 2-29 Expand Service Integration and click Buses

You see two SCA buses listed as shown in Figure 2-30.

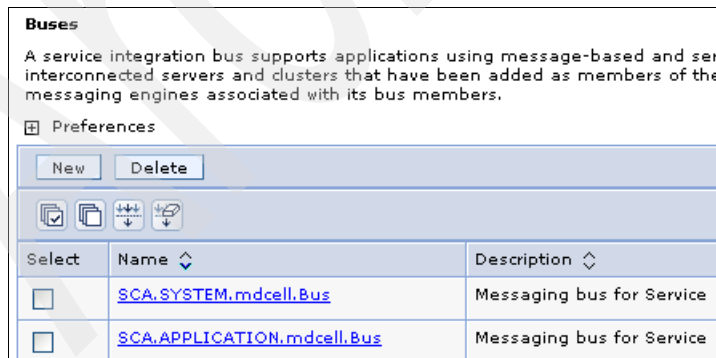


Figure 2-30 The SCA system and Application buses

SCA is now configured. You can verify that it as installed and working correctly.

## 2.12.4 Verify SCA Modules and SCA runtime

To verify SCA, we deployed a simple web service and mediation module ZOSBookOrderApp as described in White Paper WP101218, which you can obtain from the following Web page:

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

## 2.12.5 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and both of the node configuration ZFS data sets. We used the JCL shown in Example 2-29 on page 104 but changed the //DASD DD card to //DASD DD DSN=WASCFG.MDCELL.DUMP.POSTSCA.

## 2.13 Remote SCA in AppTarget cluster

**Note:** If you are configuring a single-cluster topology, skip this section. It applies only if SCA is in a different cluster from BPC.

When the BPC and Human Task containers are configured in a different cluster from the one where SCA is configured, it is necessary to configure remote SCA in the cluster hosting BPC before running the BPC wizard to configure BPC. The BPC configuration wizard checks that SCA, or remote SCA, has been configured and will not continue if it does not find one of these in the cluster where you intend to configure BPC.

1. Log in to the Integrated Solutions Console and navigate to **Servers** → **Clusters**. Click the link to the **AppTarget** cluster.
2. Under the Business Integration, click **Service Component Architecture**.
3. Select the Support the Service Component Architecture components check box and the Remote radio button, as shown in Figure 2-31.

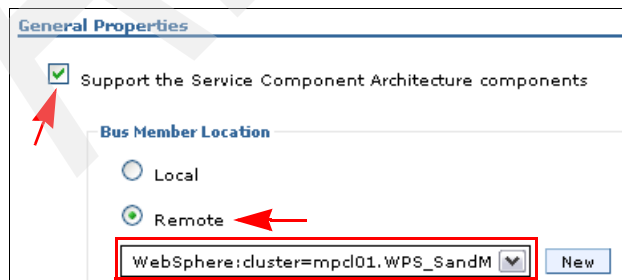


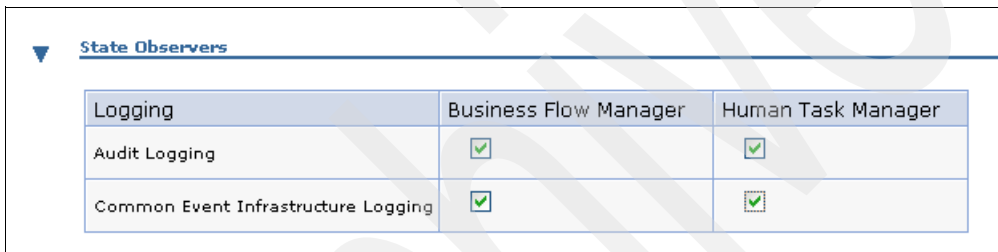
Figure 2-31 Select Remote and then choose the Support and Messaging cluster

4. Click **OK**. Do not change any fields related to the database configuration.
5. Save the configuration changes.

## 2.14 Configuring CEI using the Integrated Solutions Console

Common Event Infrastructure (CEI) is used by the BPC Observer. If you plan to configure BPC Observer as part of configuring the BPC container, configure CEI before configuring BPC.

Later, when you configure the BPC and Human Task containers, you need to check the boxes that enable CEI event logging and auditing so that events that the Event Collector collects for use by the BPC Observer can be written to CEI. See Figure 2-32.



State Observers		
Logging	Business Flow Manager	Human Task Manager
Audit Logging	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Common Event Infrastructure Logging	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 2-32 Setting auditing and event logging when configuring BPC

If you do not intend to use the BPC Observer, you can leave configuring CEI until later. If your applications do not use CEI, you can skip this section because configuring CEI is optional.

You can configure CEI using the Integrated Solutions Console or using shell scripts. We decided to use the wizards provided in the Integrated Solutions Console.

The wizard that configures the CEI does nearly all the work for you, but there are some manual changes required to complete the configuration. CEI is part of the support infrastructure so in the two-clusters topology it is configured in the Support and Messaging cluster.

Perform the following steps to configure CEI using the Integrated Solutions Console:

1. Run the CEI configuration wizard.  
See page 131.
2. Update the CEI bus authentication alias.  
See page 135.
3. Update the CEI bus connector role.  
See page 136.
4. Check the database name on the CEI data sources.  
See page 138.
5. Review the CEI DDL.  
See page 139.
6. Edit and execute the CEI Event database DDL.  
See page 142.
7. Grant table privileges on CEI tables.  
See page 143.
8. Define CEI roles to RACF.  
See page 143.
9. Restart the cluster.  
See page 144.
10. Verify the CEI configuration and CBE Browser.  
See page 146.
11. Back up the configuration ZFS datasets.  
See page 146 .

## 2.14.1 Run the CEI configuration wizard

Perform the following steps to run the CEI configuration wizard.

1. Navigate to **Servers** → **Clusters**.
2. Click the cluster where you want to configure CEI.
3. Under the Business Integration head, expand **Common Event Infrastructure** and click **Common Event Infrastructure Server**, as shown in Figure 2-33.

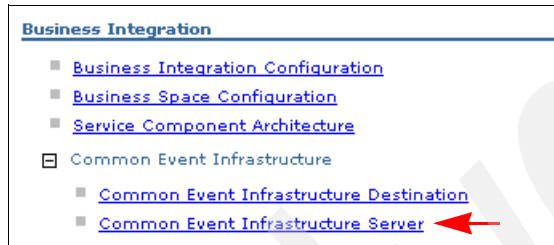
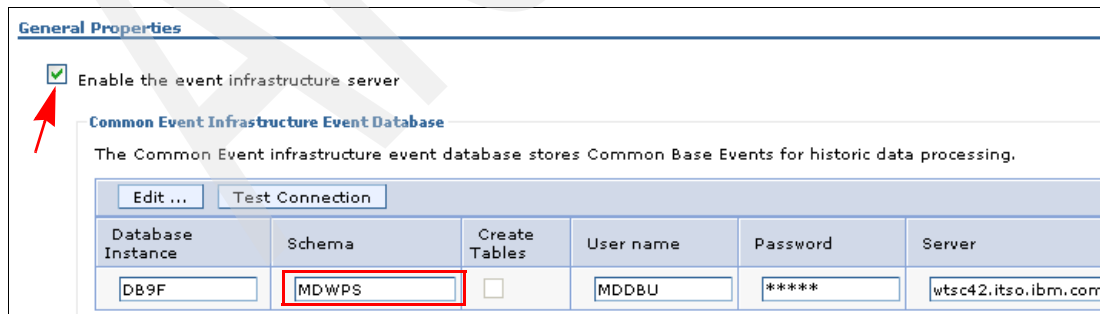


Figure 2-33 Launch the CEI configuration wizard

4. Select the Enable the event infrastructure server check box. Review the contents of the two tables that set the properties used to access the Event database and the CEI message engine's data store.

Most of the fields already contain the values you entered in the mddmnodeDmgr.rsp file when you ran zWPSCfg.sh against the deployment manager's node. However, we found that we had to re-enter the correct schema name as shown in Figure 2-34.

As with SCA, the Database Instance field is set to the DB2 Location name in order to correctly create the type 2 data sources with the Location name in their Database name fields.



Database Instance	Schema	Create Tables	User name	Password	Server
DB9F	MDWPS	<input type="checkbox"/>	MDDBU	*****	wtsc42.itso.ibm.com

Figure 2-34 Setting the database properties for the CEI Event database

**Notes:** In WebSphere Process Server for z/OS, the Database name field of type 2 data sources is set to the DB2 Location name rather than the Database name.

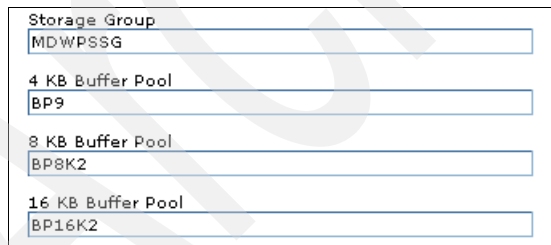
The Database Instance field in Figure 2-34 is set to the DB2 z/OS Location name, which will correctly configure the CEI event and event\_catalog data sources with the location name in the Database name field.

However, specifying the DB2 Location name in the Database instance field of the wizard means that the wizard will use the Location name as the database name in the CEI DDL that it creates. This does not matter if you created all the DDL earlier using the createDB.sh. If you did not use createDB.sh, and expect to use the DDL generated by the CEI wizard, you will need to edit the generated DDL to set the correct database name.

5. Click **Edit** in the top table to set more detailed properties for the Event data source, as shown in Figure 2-35.

**Note:** Setting the correct values here is necessary if you want the wizard to correctly customize the CEI DDL it will generate.

If you have already created the CEI DDL using createDB.sh, do not worry about these values. However, we suggest that you set the correct Storage Group name and the buffer pools here so that the values recorded in the Integrated Solutions Console match the real values.



The screenshot shows a configuration form with four input fields. The first field is labeled 'Storage Group' and contains the text 'MDWPSSG'. The second field is labeled '4 KB Buffer Pool' and contains 'BP9'. The third field is labeled '8 KB Buffer Pool' and contains 'BP8K2'. The fourth field is labeled '16 KB Buffer Pool' and contains 'BP16K2'.

Storage Group	MDWPSSG
4 KB Buffer Pool	BP9
8 KB Buffer Pool	BP8K2
16 KB Buffer Pool	BP16K2

Figure 2-35 Setting the storage group name and bufferpools for CEI

6. Verify that the Create Tables check box is cleared.

At the bottom of the panel shown in Figure 2-36, a message box tells you where the CEI DDL is generated.

Click **OK** to return the General properties window.

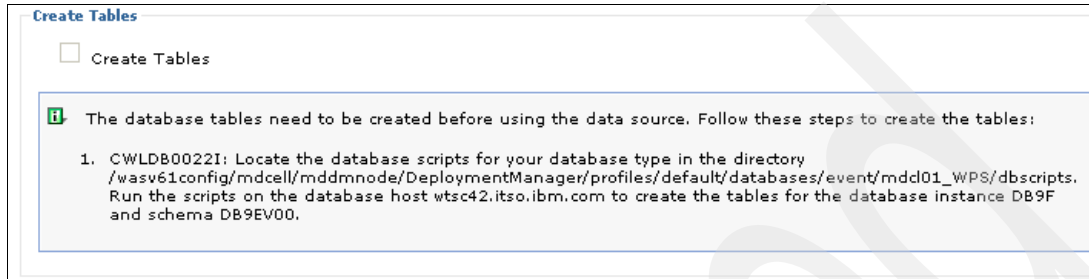


Figure 2-36 Do not Create Tables and note where the DDL is created

7. Select the Local radio button to configure the CEI bus in the local cluster, as shown in Figure 2-37.

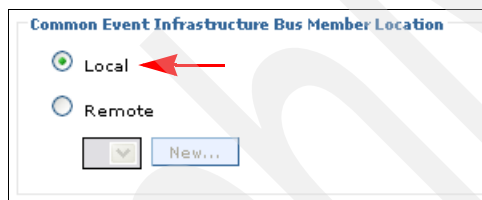


Figure 2-37 Choose to define CEI in a local cluster

8. Set the Database Instance to the DB2 Location name and set the appropriate schema, as shown in Figure 2-38.

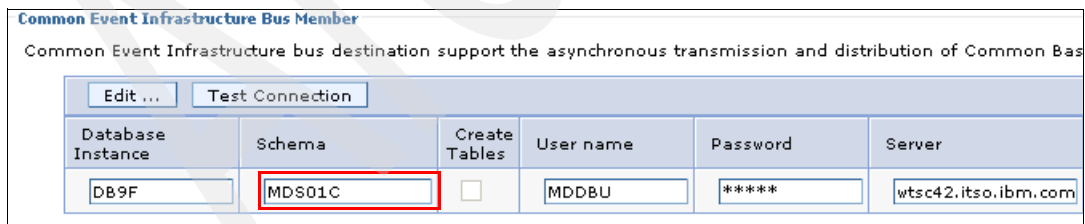
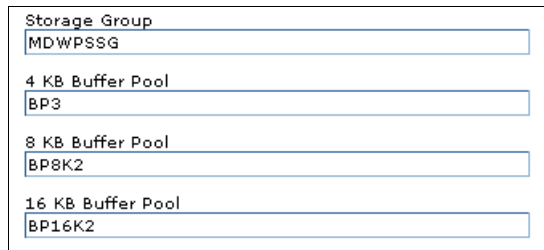


Figure 2-38 Database configuration for the CEI message engine data store

9. Click **Edit** in the bottom table to set more detailed properties for the CEI message engine's data source, as shown in Figure 2-39.

Set the correct Storage Group name and buffer pools as before.

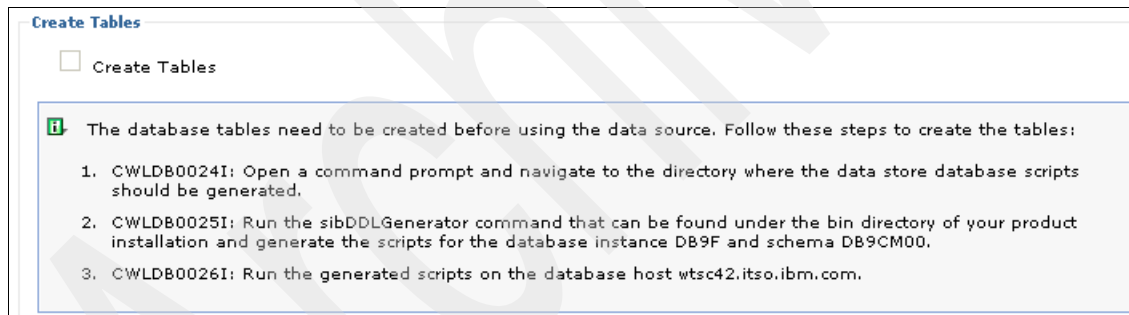


Storage Group	MDWPSSG
4 KB Buffer Pool	BP3
8 KB Buffer Pool	BP8K2
16 KB Buffer Pool	BP16K2

Figure 2-39 Set the database properties for the CEI message engine

10. Verify that the Create Tables check box is cleared.

The message box shown in Figure 2-40 displays the location of the DDL for the Event database and tells you to run `sibDDLGenerator.sh` to generate DDL for CEI Service Integration Bus (SIB). However, this should already have been done in Section 2.12.1, “Generate DDL for all SIBs using `sibDDLGenerator.sh`” on page 122.



**Create Tables**

☐ Create Tables

**i** The database tables need to be created before using the data source. Follow these steps to create the tables:

1. CWLDB0024I: Open a command prompt and navigate to the directory where the data store database scripts should be generated.
2. CWLDB0025I: Run the `sibDDLGenerator` command that can be found under the bin directory of your product installation and generate the scripts for the database instance DB9F and schema DB9CM00.
3. CWLDB0026I: Run the generated scripts on the database host wtsc42.itso.ibm.com.

Figure 2-40 Note the instruction to run `sibDDLGenerator.sh`

11. Click **OK** to return to the General properties window. Click **OK**.
12. Save the configuration changes and synchronize to the nodes.



## 2.14.2 Update the CEI bus authentication alias

The values you set in the wizard are used for defining resources to WebSphere Process Server for z/OS and are used to substitute variables in the skeleton DDL for CEI. The wizard does not prompt you for details of the J2C authentication alias user ID that is used to authenticate when connecting to the CEI bus.

A default value of CEI is used as the user ID in the J2C authentication alias called `CommonEventInfrastructureJMSAuthAlias` that is used by the CEI message engine. Also, that user ID CEI is permitted to connect to the bus. If you want to use a user ID other than CEI, you must update both the J2C authentication alias and the CEI bus security role settings.

For our cell we planned to authorize the user ID `MDJMSU` to connect to the CEI bus. We updated the J2C authentication alias that was created by the wizard, (`CommonEventInfrastructureJMSAuthAlias`), so it contained the user ID and password of `MDJMSU`. We updated the bus security to permit `MDJMSU` to connect to the CEI bus.

Perform the following steps to update the CEI bus authentication alias:

1. Navigate to **Resources** → **JDBC** → **Datasources** (Figure 2-41). You will see the three new datasources for CEI.

<input type="checkbox"/>	<a href="#">CEI ME data source</a>	<code>jdbc/com.ibm.ws.sib/mdd01_WPS-CommonEventInfrastructure_Bus</code>	<code>Cluster=mdd01_WPS</code>
<input type="checkbox"/>	<a href="#">event</a>	<code>jdbc/cei</code>	<code>Cluster=mdd01_WPS</code>
<input type="checkbox"/>	<a href="#">event_catalog</a>	<code>jdbc/eventcatalog</code>	<code>Cluster=mdd01_WPS</code>

Figure 2-41 CEI datasources

2. Click **CEI ME data source**. Then, under Related items, click the **JAAS - J2C authentication data** link. Notice in Figure 2-42 that the user ID set for the `CommonEventInfrastructureJMSAuthAlias` J2C authentication alias is set to the user ID `CEI`.

<input type="checkbox"/>	<a href="#">CommonEventInfrastructureJMSAuthAlias</a>	<code>CEI</code>	Authentication alias for the Common Event Infrastructure JMS Topics and Queues
--------------------------	---	------------------	--

Figure 2-42 A default user ID is set for `CommonEventInfrastructureJMSAuthAlias`

- Click **CommonEventInfrastructureJMSAuthAlias** and set the correct alias user ID and password.

For our MD cell, the alias user ID was MDJMSU, and the password was MDJMSU. See Figure 2-43.

**General Properties**

\* Alias  
CommonEventInfrastructureJMSAuthAlias

\* User ID  
MDJMSU

\* Password  
\*\*\*\*\*

Description  
Authentication alias for the C

Apply OK Reset Cancel

Figure 2-43 Set the correct JMS alias user ID and password

- Click **OK** and then save the configuration changes.

### 2.14.3 Update the CEI bus connector role

By default, the user CEI is permitted to connect to the CEI bus. You will probably need to change this to a different user ID or group. We removed CEI from the role and added group MDJMSG as described below.

- Navigate to **Service Integration** → **Buses**. There is a new bus called **CommonEventInfrastructure\_Bus** as shown in Figure 2-44.

<input type="checkbox"/>	<a href="#">CommonEventInfrastructure_Bus</a>	CommonEventInfrastructure Bus	<a href="#">Enabled</a>
--------------------------	---	-------------------------------	-------------------------

Figure 2-44 CEI bus

- Click **CommonEventInfrastructure\_Bus**, then click **Security** under Additional properties (Figure 2-45).

**Additional Properties**

■ [Custom properties](#)

■ [Security](#)

■ [Web service gateway instances](#)

Figure 2-45 Link to CEI bus security

3. Click **Users and groups in the bus connector role** under Additional properties (Figure 2-46).



Figure 2-46 Link to CEI bus role assignments

4. Select the CEI check box, and click **Delete** to delete the old role permission (Figure 2-47).

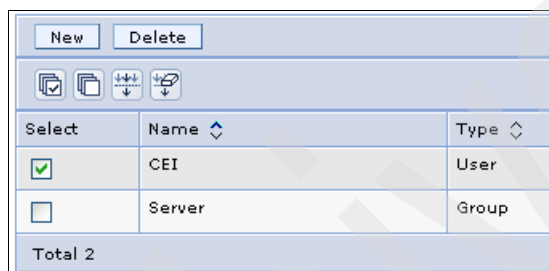


Figure 2-47 Delete user CEI from the role

5. Click **New** to authorize a new group or user to connect to the bus.

You can choose to permit a group or a user ID to the role. We decided to permit the default group of the user ID MDJMSU. This was the user ID we planned to use as the JMS Authentication Alias user ID, as shown in Figure 2-48. We permitted the group so it was possible to allow other users to connect to the CEI bus by connecting them to group MDJMSG.

Buses > CommonEventInfrastructure\_Bus > Security for bus  
CommonEventInfrastructure\_Bus > Users and groups in the bus connector role > New

Create a user or group in the bus connector role.

Configuration

**General Properties**

**Bus Connector Role**

☒ Group name

☐ User name

☐ Server - Allow servers to connect to the bus

☐ All Authenticated - Allow all authenticated users to connect to the bus

☐ Everyone - Allow unauthenticated users to connect to the bus

OK Reset Cancel

Figure 2-48 Permit the JMS authentication alias user ID's group to the role

6. Click **OK** and then save the configuration changes.

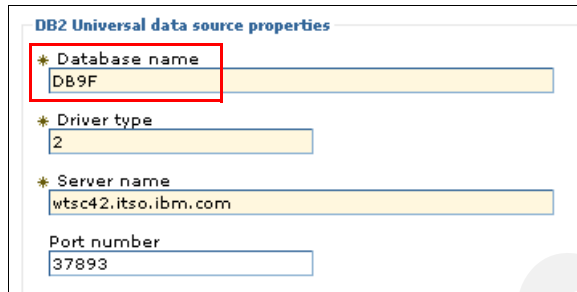
#### 2.14.4 Check the database name on the CEI data sources

If you entered the CEI database name in the Database Instance field of the CEI configuration wizard so that CEI DDL was customized with the correct database name, you must correct the CEI data sources.

On type 2 data sources event and event\_catalog, the Database name field must be updated to match the Location name of the DB2 subsystem that holds the CEI tables.

1. Navigate to: **JDBC providers** → **DB2 Universal JDBC Driver Provider** → **Data sources**.
2. Click the **CEI ME data source** data source.

3. Scroll down to the bottom of the CEI ME data source panel and change the Database name field to the DB2 z/OS Location name. If you do not set the Location name as shown in Figure 2-49, the bus will fail to start with a -551 SQLCODEN551.



DB2 Universal data source properties

\* Database name  
DB9F

\* Driver type  
2

\* Server name  
wtsc42.itso.ibm.com

Port number  
37893

Figure 2-49 Set DB2 location name in the Database name field

4. Click **OK**, but do not save the configuration changes yet.
5. Click the event data source and make the same change to set the Database name field to the DB2 location name.
6. Click the event\_catalog data source and make the same change to set the Database name field to the DB2 location name.
7. For the event and event\_catalog data sources, set either the currentSchema or currentSQLID custom properties to the value of the schema you are using for CEI in DB2 for z/OS. We recommend that you use the currentSchema property discussed in Section 2.2.10, “Choosing currentSchema or currentSQLID” on page 48.
8. Save the configuration changes.

### 2.14.5 Review the CEI DDL

**Note:** Skip this step if you created the CEI DDL using the createDB.sh.

Unlike the other components of WebSphere Process Server for z/OS, the Data Definition Language (DDL) statements that create the CEI tables are generated in EBCDIC, so there is no need to convert from ASCII.

We copied the DDL from its location in the UNIX file system into z/OS sequential files so we can check it and edit it using the ISPF editor.

We created sample job MDCPYCEI, shown in Example 2-36, to perform the copy.

*Example 2-36 Sample job MDCPYCEI to copy CEI DDL to z/OS sequential files*

```
//MDCPYCEI JOB (0),'MDADMIN',CLASS=A,REGION=0M,NOTIFY=&SYSUID
/*JOBPARM S=SC42
/*****
/* Make sure you edit this file with CAPS OFF. */
/* Run this job using an administrator userid or UID(0) */
/* Make sure the admin userid can create the zOS datasets */
/* */
/* !!WARNING!! If you are running non-default z/OS EBCDIC */
/* code page (default is IBM-037) then you must specify */
/* the correct code page translation table on the CONVERT */
/* option of the oget command instead of CONVERT(YES) */
/*****
/*****
/* COPY DDL from UNIX to zOS */
/*****
//COPY EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
cd /wasv61config/mdcell/mddmnode/DeploymentManager/; +
cd profiles/default/databases/event/mdcl01_WPS/+
dbscripts/db2zos/ddl; +
cp 'cr_db.ddl' /tmp/MDEVTDB +
1> /tmp/ceisqlcopy_1029.out +
2> /tmp/ceisqlcopy_1029.err; +
cp 'cr_db_catalog.ddl' /tmp/MDEVTCDB +
1> /tmp/ceisqlcopy_1030.out +
2> /tmp/ceisqlcopy_1030.err; +
cp 'cr_tbl.ddl' /tmp/MDEVTTB +
1> /tmp/ceisqlcopy_1031.out +
2> /tmp/ceisqlcopy_1031.err; +
cp 'cr_tbl_catalog.ddl' /tmp/MDEVTCTB +
1> /tmp/ceisqlcopy_1032.out +
2> /tmp/ceisqlcopy_1032.err; +
cp 'ins_metadata.ddl' /tmp/MDEVTINS +
1> /tmp/ceisqlcopy_1033.out +
2> /tmp/ceisqlcopy_1033.err; +
cp 'catalogSeed.ddl' /tmp/MDEVTSEE +
1> /tmp/ceisqlcopy_1034.out +
2> /tmp/ceisqlcopy_1034.err;
```

```

/*
//*****
/* STEPS TO COPY THE OUTPUT THE JOB LOG */
//*****
//CPOUT EXEC PGM=IKJEFT01,REGION=0M
//SYSEXEC DD DISP=SHR,DSN=BB06142.MDDMNODE.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
    BBOHFSWR '/tmp/ceisqlcopy_1029.out'
    BBOHFSWR '/tmp/ceisqlcopy_1029.err'
    BBOHFSWR '/tmp/ceisqlcopy_1030.out'
    BBOHFSWR '/tmp/ceisqlcopy_1030.err'
    BBOHFSWR '/tmp/ceisqlcopy_1031.out'
    BBOHFSWR '/tmp/ceisqlcopy_1031.err'
    BBOHFSWR '/tmp/ceisqlcopy_1032.out'
    BBOHFSWR '/tmp/ceisqlcopy_1032.err'
    BBOHFSWR '/tmp/ceisqlcopy_1033.out'
    BBOHFSWR '/tmp/ceisqlcopy_1034.err'
    BBOHFSWR '/tmp/ceisqlcopy_1034.out'
    BBOHFSWR '/tmp/ceisqlcopy_1034.err'
/*
//*****
/* COPY DDL from UNIX to zOS */
//*****
//COPY EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
    OGET '/tmp/MDEVTDDB' 'WASCFG.MDCELL.SQL.MDEVTDDB'
    OGET '/tmp/MDEVTCDB' 'WASCFG.MDCELL.SQL.MDEVTCDB'
    OGET '/tmp/MDEVTTB' 'WASCFG.MDCELL.SQL.MDEVTTB'
    OGET '/tmp/MDEVTCTB' 'WASCFG.MDCELL.SQL.MDEVTCTB'
    OGET '/tmp/MDEVTINS' 'WASCFG.MDCELL.SQL.MDEVTINS'
    OGET '/tmp/MDEVTSEE' 'WASCFG.MDCELL.SQL.MDEVTSEE'
/*
/* Clean up /tmp
/*
//DELTMP EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
    BPXBATCH SH +
        rm /tmp/MDEV* ; +
        rm /tmp/ceisqlcopy*.out ; +
        rm /tmp/ceisqlcopy*.err ;
/*

```

## 2.14.6 Edit and execute the CEI Event database DDL

**Note:** Skip this step if you created the CEI DDL using the createDB.sh.

After copying the CEI DDL to z/OS sequential datasets we reviewed the DDL and performed some minor editing.

We commented out the CREATE DATABASE command because we had already created a database called MDEVTDDB for CEI using job MDCREDB. Apart from this change, the rest of the DDL was correctly formatted in columns 1–72 and contained the correct buffer pools settings. Therefore no further edits were necessary.

The DDL was copied from the sequential files to the partitioned dataset we used to hold the rest of the DDL, WASCFG.MDCELL.WPS.SQL.

Sample job MDDDLCEI, shown in Example 2-37, was created to execute the CEI DDL. The order of the SYSIN concatenation is important. You cannot insert data into tables until after they have been created.

*Example 2-37 Sample JCL to execute the CEI DDL*

```
//MDDDLCEI JOB (0),'MDADMIN',CLASS=A,REGION=OM,NOTIFY=&SYSUID
/*JOBPARM S=SC42
/*****
/* Run this job using a userid with DB2 SYSADM authority */
/* or grant DBADM to the <sqlid> on SET CURRENT SQLID. */
/* Set the SYSAFF to the LPAR for this node. */
/*****
//JOBLIB DD DISP=SHR,DSN=DB9F9.SDSNEXIT
// DD DISP=SHR,DSN=DB9F9.SDSNLOAD
//*
/* Before running this:
/* The userid on the SET CURRENT SQLID statement must
/* have DB2 SYSADM authority and EXECUTE on PLAN DSNTIA91
/*
/* Create WPS Common Tables and prime the relationships table
/*
//CREATE EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
```



```

DSN SYSTEM(D9FG)
RUN PROGRAM(DSNTEP2) PLAN(DSNTEP91) LIB('DB9FU.RUNLIB.LOAD')
//SYSIN      DD *
SET CURRENT SQLID = 'MDWPS' ;
//          DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDEVTDB)
//          DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDEVTCDB)
//          DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDEVTTB)
//          DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDEVTCTB)
//          DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDEVTINS)
//          DD DISP=SHR,DSN=WASCFG.MDCELL.WPS.SQL(MDEVTSEE)

```

---

## 2.14.7 Grant table privileges on CEI tables

**Note:** Skip this step if you already created all the DDL using the createDB.sh and have therefore already granted table privileges in Section 2.6.8, “Grant table privileges to the J2C auth alias user ID” on page 88.

If you did not create all the tables in Section 2.6, “Configure DB2 z/OS using createDB.sh” on page 79, and you are specifying the schema for CEI on the currentSchema property of the CEI data sources, you will need to grant table privileges to the J2C authentication alias user ID now.

We created GRANT commands in our WASCFG.MDCELL.WPS.SQL library and sample job MDGRANT to execute them. Review, modify, and submit this job or execute equivalent GRANT statements using SPUFI.

**Note:** The job MDGRANT job will grant table privileges for all the WebSphere Process Server for z/OS tables. You will need to comment out all but the GRANT statements that apply to CEI on the SYSIN DD card.

## 2.14.8 Define CEI roles to RACF

When using RACF to perform EJBROLE authorization checking, there are several CEI roles that must be defined as RACF profiles in class EJBROLE.

RACF definitions for CEI roles were included in the sample jobs MDEJBROL and GEJBROLE, which you executed in Section 2.3.3, “Create RACF user IDs, groups, and EJBROLE profiles” on page 62. If you ran the MDEJBROL or GEJBROLE jobs at that time you will not need to define roles for CEI now. However, we review some key points about defining these roles and deciding on the access list.

You need to decide for each role whether to make it available to all authenticated users, by defining the EJBROLE profile as UACC(READ), or to define the profile as UACC(NONE) and then PERMIT a suitable group to the access list.

We defined the roles as follows:

- ▶ UACC(NONE):
  - CEI eventAdministrator
  - catalogAdministrator
- ▶ UACC(READ):
  - eventCreator
  - eventConsumer
  - eventUpdater
  - catalogReader

When you verify SCA and CEI using the ZOSBookOrderApp, that application does not prompt for a login, and so runs as UNAUTHENTICATED. When your WebSphere Process Server for z/OS cell is using a RACF user registry, running as UNAUTHENTICATED means that requests run under the guest user ID (for our MD cell: MDGUEST). Because the guest user ID is created as a RESTRICTED user ID in RACF, it must be explicitly permitted any EJBROLE profiles that are UACC(READ). We did this in the sample job MDEJBROL.

### 2.14.9 Restart the cluster

After executing the CEI DDL and creating the RACF EJBROLE profiles, the cluster is restarted. The CEI bus will not start properly until the cluster is restarted because it uses a data source that references a new J2C authentication alias and the RACF-related control blocks for an authentication alias are built during server initialization.

You can delay starting the cluster until other components such as BPC and Business Space have been configured, but we suggest that you start the WebSphere Process Server for z/OS cluster at this point to verify that the buses start and that there are no errors related to communication with DB2.

1. Navigate to **Servers** → **Clusters**, select the check box next to the cluster, and then click **Start** as shown in Figure 2-50.

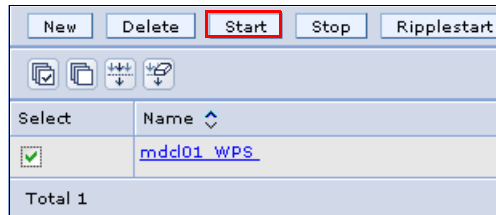


Figure 2-50 Start the cluster

2. Check the logs of the adjunct and the servant for errors.

**Shortcut:** The **find exception: 1 30** command is useful for quickly locating exceptions in the servant logs.

During restart, look in the WebSphere Process Server for z/OS servant logs for any errors. Some errors and their solutions are described in Chapter 4, “Troubleshooting WebSphere Process Server for z/OS” on page 305.

Check that the messages in the logs state that the SCA buses and the CEI bus start normally. Check any FFDC logs being written for any errors related to DB2. Review the state of the buses using the WebSphere Integrated Solutions Console.

3. Log in to the Integrated Solutions Console and navigate to **Service Integration** → **Buses**.
4. Click **CommonEventInfrastructure\_Bus**, and then the Topology tab.
5. Expand the topology. You see the CEI message engine started as shown in Figure 2-51. If it is not started, look in the servant log(s) for errors.

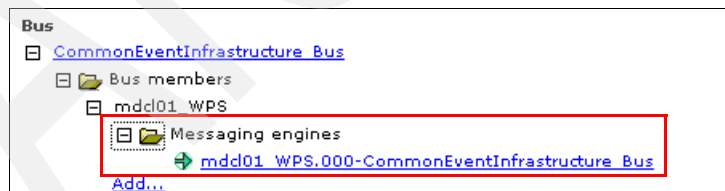


Figure 2-51 Verify that the CEI bus is started

6. Check the SCA system and SCA application buses in the same way. If you are configuring more than one cluster, we recommend that you leave the verification of remote access to SCA from the second cluster until you have configured BPC in the second cluster.

## 2.14.10 Verify the CEI configuration and CBE Browser

To verify CEI and CBE Browser, we followed the procedure described in White Paper *Performing Installation Verification for WPS on z/OS V6.1*, WP101218, which you can download from the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101218>

Ensure that the user ID that you use to test the CBE Browser has at least the eventCreator role. On our system we connected the BPE Administrator user ID, MDBPEADM, to the group MDEVTCRG. MDEVTCRG was the group we had permitted to the RACF EJBROLE profile MD.eventCreator.

## 2.14.11 Back up the configuration ZFS datasets

Back up the deployment manager's configuration ZFS data set and both of the node's configuration ZFS data sets after configuring CEI. We used the JCL shown in Example 2-9 on page 73 but changed the //DASD DD card to //DASD DD DSN=WASCFG.MDCELL.DUMP.POSTCEI.

## 2.15 Business Process and Human Task containers

Unlike WebSphere Process Server on non-z/OS platforms it is not possible to configure the Business Process and Human Task Container using the Deployment Environments wizard on the WebSphere Administration console. Instead, you use the wizards provided under Container Settings on the Cluster configuration panel.

This section describes how to configure Business Process Container in both a single-cluster topology and the two-clusters topology. The examples and screenshots are from the single-cluster topology. Differences for two-clusters are highlighted.

The process to configure Business Process Container in both a single-cluster topology and the two-clusters topology is as follows:

1. "Execute the BPC configuration wizard" on page 147.
2. "Set the schema name for the BPC message engine" on page 152.
3. "Set currentSchema on the BPC datasource" on page 153.
4. "Verify that the applications are present" on page 154.
5. "Restart the cluster hosting BPC" on page 154.
6. "Back up the WebSphere configuration HFS datasets" on page 154.

**Note:** When SCA has been configured in a different cluster from the one where you are configuring BPC, you must configure remote SCA in the cluster that is home to BPC before you configure BPC.

If you intend to use the BPC Observer, be sure to configure CEI before configuring the Business Process Containers.

## 2.15.1 Execute the BPC configuration wizard

Perform the following steps to execute the BPC configuration wizard.

1. Log in to the Integrated Solutions Console and navigate to **Servers** → **Clusters**.

**Note:** When configuring the two-clusters topology, the common components of WebSphere Process Server for z/OS, along with SCA and CEI are configured in the Support and Messaging cluster. In our scenario, for example, these were configured in cluster mpcl01.WPS\_SandM.

The BPC and the Human Task containers are configured in the second AppTarget cluster, which was mpcl02.AppTarget in our two-clusters topology.

2. Click the link to the cluster. For our single-cluster scenario we selected cluster mdcl01\_WPS. For our two-clusters scenario we selected mpcl02.AppTarget.
3. Under Container Settings, expand **Business Process Choreographer Container Settings** and click **Business Process Choreographer Containers** as shown in Figure 2-52.

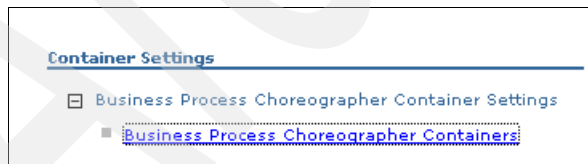


Figure 2-52 Launch the BPC containers wizard

4. In the Data Source section (Figure 2-53), many of the fields are populated with the values you specified in the mddmnodeDmgrDB2.rsp response file when you ran zWPSConfig.sh against the deployment manager's node. Set the Database Instance field to the DB2 location name and set the Schema Name used by BPC tables as shown in Figure 2-53.

Database Instance	Schema Name	Create Tables	User Name	Password	Server	Provider
DB9F	MDWPS	<input type="checkbox"/>	MDDBU	*****	wtsc42.itso	DB2 for z/OS v9

Figure 2-53 Set the Database Instance and Schema Name

**Note:** When using a type 2 data source to connect to DB2 for z/OS, the Database Instance field is set to the location name of the DB2 for z/OS system. This will configure the type 2 data source correctly.

However, if you want the wizard to generate customized DDL for BPC that contains the correct database name, you may prefer to enter the BPC database name in the Database Instance field so the DDL is correctly customized. When you have executed the wizard, you will need to change the Database Name field on the BPC datasource so it contains the DB2 location name rather than the database name.

5. Clear the Enable e-mail service check box (Figure 2-54), unless you have a mail server available and need to enable this service.

Human Task Manager Mail Session	
<input type="checkbox"/>	Enable e-mail service
Mail transport host	
Mail transport user	
MDADMIN	
Mail transport password	
Confirm mail transport password	
Business Process Choreographer Explorer URL	

Figure 2-54 Un-check Enable e-mail service

6. Expand the Security section and set the user IDs and groups that you planned to permit to the BPESystemAdministrator and BPESystemMonitor roles. See Section 2.2.6, “Planning security” on page 38.

Set the user IDs and passwords that are used for the JMS JAAS Authentication Alias and for the two runAs role user IDs, JMS API authorization and Escalation user as shown in Figure 2-55.

The screenshot shows the 'Security' configuration page. It contains two main sections. The first section is a table for role assignments:

Role	User	Group	Description
Administrator	MDBPEADM	MDBPEADG	User name(s) and/or group name(s) assigned to this role have administrative access to the system.
Monitor	MDBPESM	MDBPESMG	User name(s) and/or group name(s) assigned to this role can monitor the system.

The second section is a table for authentication settings:

Authentication	User	Password	Confirm Password
JMS Authentication	MDJMSU	*****	*****
JMS API Authentication	MDJMSAPI	*****	*****
Escalation User Authentication	MDESCAL	*****	*****

In both tables, the user and group names in the first table, and the user names and passwords in the second table, are highlighted with red boxes.

Figure 2-55 Customizing security settings for BPC

7. If you plan to use BPC Observer, select the check boxes in the State Observers section as shown in Figure 2-56. This will cause events to be emitted that the BPC Event Collector will collect and make available to the BPC Observer.

The screenshot shows the 'State Observers' configuration page. It contains a table with logging options:

Logging	Business Flow Manager	Human Task Manager
Audit Logging	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Common Event Infrastructure Logging	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

The checkboxes for 'Audit Logging' and 'Common Event Infrastructure Logging' in both columns are highlighted with red boxes.

Figure 2-56 Choose to enable State Observers if using BPC Observer

The SCA bindings section (Figure 2-57) do not need changing, so scroll to the next section of the wizard.

SCA Bindings		
Host	Context Root	Relative Path
http://host:port	/BFMIF_md	/sca/com/ibm/bpe/spi/sca/BFMWS
http://host:port	/HTMIF_mc	/sca/com/ibm/bpe/spi/sca/HTMWS

Figure 2-57 The SCA binding do not need changing

**Note:** The gold topology for non z/OS WebSphere Process Server configurations has all the message engines running in a separate cluster. Because a WebSphere Application Server for z/OS server runs message engines in a separate adjunct address space, this separation is not necessary. We configured the SCA and CEI buses in the same cluster as the WebSphere Process Server common support components. We called this our Support and Messaging cluster.

When configuring the BPC and Human Task containers in a different cluster from the other WebSphere Process Server for z/OS components, you can choose to configure the BPC bus in either the Support and Messaging cluster, or in the AppTarget cluster.

For performance reasons we suggest that you define the BPC bus Locally in the cluster that hosts BPC.

8. Select the Use the default configuration check box for the BPC bus, as shown in Figure 2-58. This will create the BPC bus in the same cluster as BPC.

▼

Bus

☒ Use the default configuration

Figure 2-58 Check the box to use the default configuration for the BPC bus



9. Click **Finish**. The configuration takes some time, so be patient (Figure 2-59).

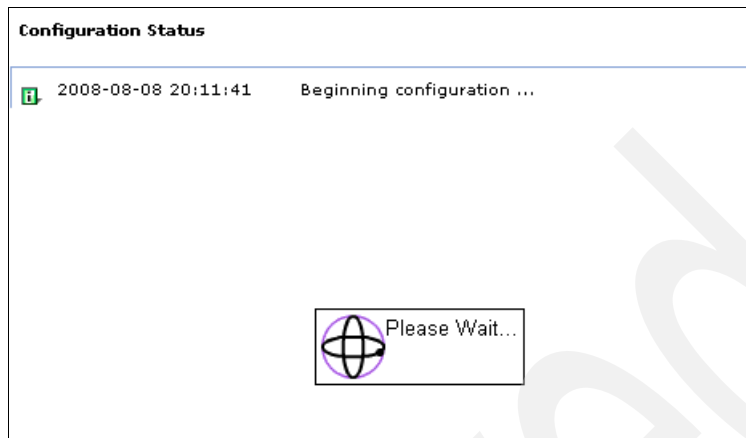


Figure 2-59 Be patient while waiting for the wizard to do its work

10. Check that you see the successful deployment messages and then click **Save Changes** as shown in Figure 2-60.

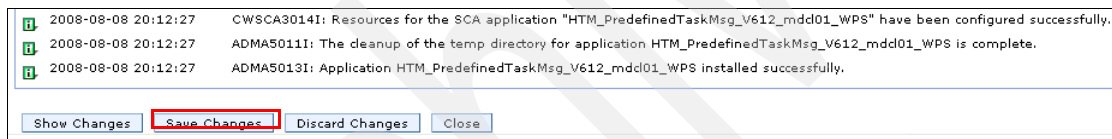


Figure 2-60 Check the messages for errors then click **Save Changes**

11. After the changes have been synchronized to the nodes, click **OK**. You see messages (like those in Figure 2-61) indicating that the BPC container and the Human Task container have been successfully installed.

We delayed restarting the server until after we had updated the Schema on the BPC bus in Section 2.15.2, "Set the schema name for the BPC message engine" on page 152.

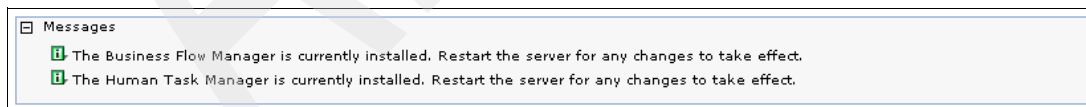


Figure 2-61 Messages confirm successful BPC and HT configuration

## 2.15.2 Set the schema name for the BPC message engine

We found that the schema name was not set correctly on the message data stores for the BPC bus.

For example, when we navigated to **Buses** → **BPC.mdcell.Bus** → **Messaging engines** → **mdcl01\_WPS.000-BPC.mdcell.Bus** → **Message store** we saw that the schema was set to DB9BM00 instead of MD01B, as shown in Figure 2-62.

The screenshot shows the configuration page for the BPC message engine. The breadcrumb navigation at the top is: **Buses** > **BPC.mdcell.Bus** > **Messaging engines** > **mdcl01\_WPS.000-BPC.mdcell.Bus** > **Data store**. Below the breadcrumb, there is a description: "The persistent store for messages and other state managed by the messaging engine." The "Configuration" tab is selected. The "General Properties" section contains the following fields: "UUID" with value "7E1AC62AD4F21E63", "Data source JNDI name" with value "jdbc/com.ibm.ws.sib/mdcl01\_WPS-BPC.mdcell.Bus", "Schema name" with value "DB9BM00" (highlighted with a red box), "Authentication alias" with value "BPCME\_00\_Auth\_Alias", and a "Create tables" checkbox. The "Number of tables for permanent objects" and "Number of tables for temporary objects" are both set to "1". The "Related Items" section on the right shows a link to "JAAS - J2C authentication data". At the bottom, there are buttons for "Apply", "OK", "Reset", and "Cancel".

Figure 2-62 Incorrect schema set for the BPC message engine

Perform the following steps to set the schema name for the BPC message engine.

1. Set the correct schema (MD01B for our MD cell) as shown in Figure 2-63) and click **OK**.

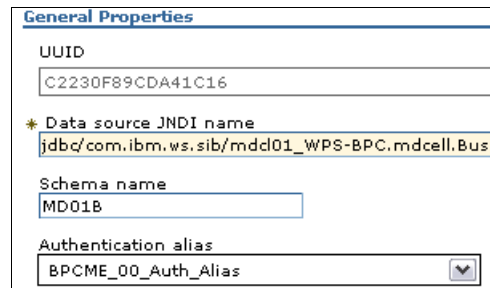


Figure 2-63 Correcting the schema for the BPC message engine

2. Save the configuration changes and restart the cluster hosting BPC.

### 2.15.3 Set currentSchema on the BPC datasource

The currentSQLID or currentSchema properties must be set manually on the BPC datasource. We suggest that you use currentSchema as discussed in Section 2.2.10, “Choosing currentSchema or currentSQLID” on page 48.

Perform the following steps to set currentSchema on the BPC datasource.

1. Click **Resources** and then expand **JDBC**.
2. Click **Data Source** → **Business Process Choreographer data source** → **Custom Properties** → **currentSchema**.
3. Enter the database schema in the value field as shown in Figure 2-64.

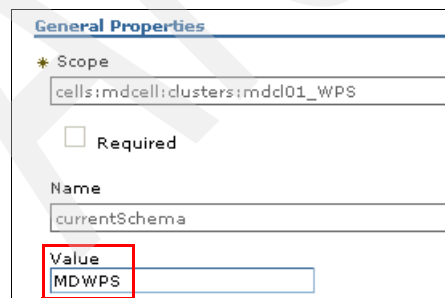


Figure 2-64 Setting the currentSchema on the BPC datasource

## 2.15.4 Verify that the applications are present

After configuring SCA and BPC you see the list of applications shown in Figure 2-65.

Their status will show as unknown when the cluster is not yet running.

<input type="checkbox"/>	<a href="#">AppScheduler</a>
<input type="checkbox"/>	<a href="#">BPEContainer mdd01_WPS</a>
<input type="checkbox"/>	<a href="#">HTM_PredefinedTaskMsg_V612_mdd01_WPS</a>
<input type="checkbox"/>	<a href="#">HTM_PredefinedTasks_V612_mdd01_WPS</a>
<input type="checkbox"/>	<a href="#">RemoteAL61</a>
<input type="checkbox"/>	<a href="#">TaskContainer mdd01_WPS</a>
<input type="checkbox"/>	<a href="#">persistentLkMqr</a>
<input type="checkbox"/>	<a href="#">sca.sib.mediation</a>
<input type="checkbox"/>	<a href="#">wpsFEMqr 6.1.2</a>
Total 9	

Figure 2-65 SCA and BPC applications

## 2.15.5 Restart the cluster hosting BPC

After checking the bus and data source schemas, stop and restart the cluster hosting BPC and check the logs for any errors.

Verify that the new BPC bus starts correctly in the adjunct.

## 2.15.6 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and both of the node's configuration ZFS data set. We used the JCL shown in Example 2-29 on page 104 but changed the //DASD DD card to: //DASD DD DSN=WASCFG.MDCELL.DUMP.POSTBPC.

## 2.16 BPC Explorer

The BPC Explorer can be configured using the Integrated Solutions Console or a jacl script called clientconfig.jacl. We used the wizard provided in the Integrated Solutions Console.

The BPC Explorer is considered part of the Support Infrastructure. Currently, however, it is only possible to deploy the BPC Explorer in the same cluster as BPC when you use the wizard in the Integrated Solutions Console.

### 2.16.1 Install BPC Explorer using the Integrated Solutions Console

This section describes how to deploy the BPC Explorer using the WebSphere Integrated Solutions Console.

1. Log in to the WebSphere Integrated Solutions Console.
2. Navigate to **Servers** → **Clusters**.
3. Click the link to the cluster where BPC is configured.
4. Under Business Integration, expand **Business Process Choreographer**. Click **Business Process Choreographer Explorer** as shown in Figure 2-66.

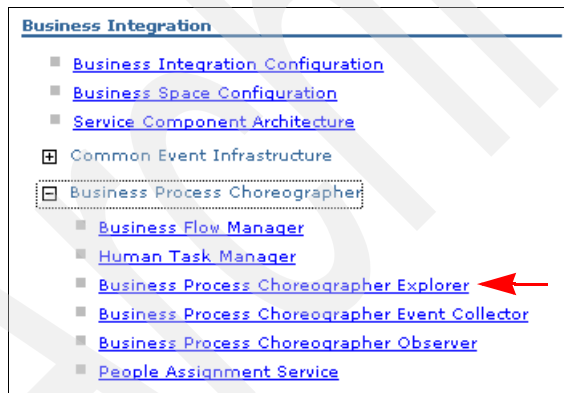
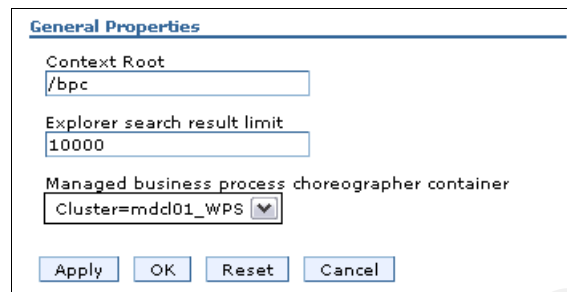


Figure 2-66 Launch wizard to configure BPC Explorer

5. Click **Add**.

6. Confirm that the context root is the one you want to use and that the Managed business process choreographer container field specifies the cluster where you want to deploy the BPC Explorer (Figure 2-67).



**General Properties**

Context Root  
/bpc

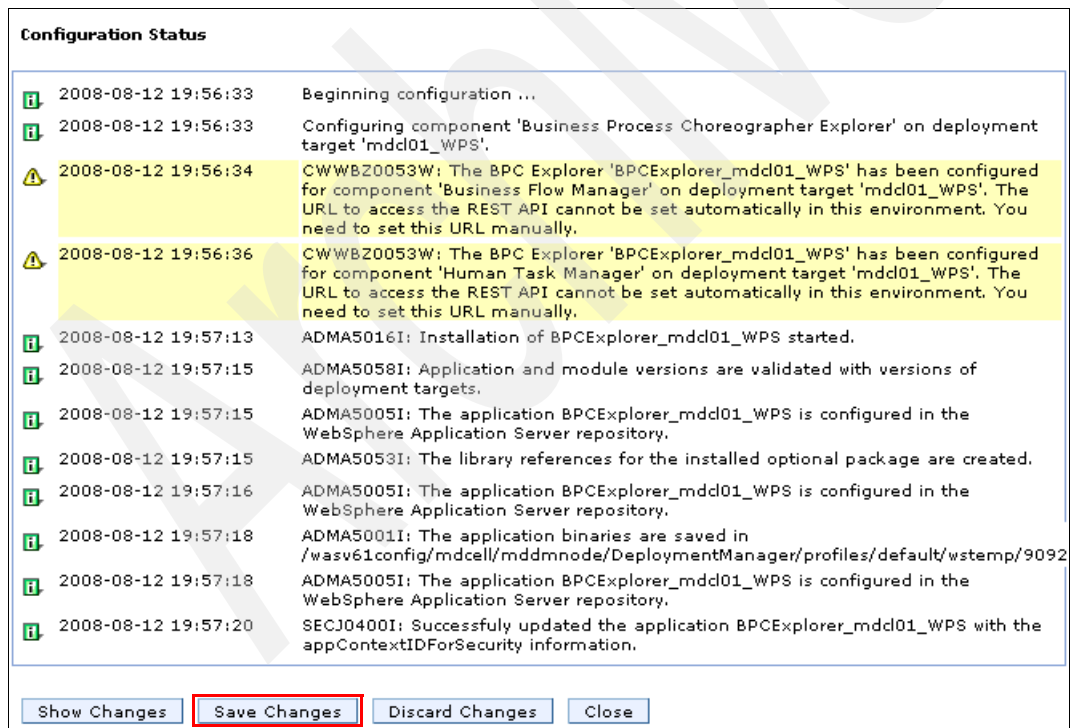
Explorer search result limit  
10000

Managed business process choreographer container  
Cluster=mdd01\_WPS

Apply OK Reset Cancel

Figure 2-67 Review the context root

7. Click **OK**.
8. Confirm that the BPC Explorer is deployed successfully as shown in Figure 2-68. Click **Save changes**.



**Configuration Status**

2008-08-12 19:56:33	Beginning configuration ...
2008-08-12 19:56:33	Configuring component 'Business Process Choreographer Explorer' on deployment target 'mdd01_WPS'.
2008-08-12 19:56:34	CWWBZ0053W: The BPC Explorer 'BPCEXplorer_mdd01_WPS' has been configured for component 'Business Flow Manager' on deployment target 'mdd01_WPS'. The URL to access the REST API cannot be set automatically in this environment. You need to set this URL manually.
2008-08-12 19:56:36	CWWBZ0053W: The BPC Explorer 'BPCEXplorer_mdd01_WPS' has been configured for component 'Human Task Manager' on deployment target 'mdd01_WPS'. The URL to access the REST API cannot be set automatically in this environment. You need to set this URL manually.
2008-08-12 19:57:13	ADMA5016I: Installation of BPCEXplorer_mdd01_WPS started.
2008-08-12 19:57:15	ADMA5058I: Application and module versions are validated with versions of deployment targets.
2008-08-12 19:57:15	ADMA5005I: The application BPCEXplorer_mdd01_WPS is configured in the WebSphere Application Server repository.
2008-08-12 19:57:15	ADMA5053I: The library references for the installed optional package are created.
2008-08-12 19:57:16	ADMA5005I: The application BPCEXplorer_mdd01_WPS is configured in the WebSphere Application Server repository.
2008-08-12 19:57:18	ADMA5001I: The application binaries are saved in /wasv61/config/mdcell/mddmnode/DeploymentManager/profiles/default/wstemp/9092
2008-08-12 19:57:18	ADMA5005I: The application BPCEXplorer_mdd01_WPS is configured in the WebSphere Application Server repository.
2008-08-12 19:57:20	SECJ0400I: Successfully updated the application BPCEXplorer_mdd01_WPS with the appContextIDForSecurity information.

Show Changes **Save Changes** Discard Changes Close

Figure 2-68 BPC Explorer deployment messages

9. When using RACF to perform EJBROLE authorization checking, define the role WebClientUser to RACF. We had already done this as part of the job MDEJBROL. The applicable commands are shown in Example 2-38.

*Example 2-38 Extract of sample job MDEJBROL showing the commands to define EJBROLE WebClientUser for BPC Explorer*

---

RDEFINE EJBROLE	MD.WebClientUser	+
	UACC(NONE)	+
	OWNER(WASCFG)	+
	DATA('MDCELL WebClientUser role')	+
	APPLDATA('MDWCU')	
PERMIT	MD.WebClientUser	+
	CLASS(EJBROLE)	+
	RESET	
PERMIT	MD.WebClientUser	+
	CLASS(EJBROLE)	+
	ID(MDWCUG,	+
	MDBPESMG,MDBPEADG,MDHTSMG,MDHTADMG) +	
	ACCESS(READ)	

---

If you prefer to use GEJBROLE profiles, see job GEJBROLE instead. The use of GEJBROLE profiles rather than EJBROLE profiles is discussed in Section 2.2.7, “Managing EJB roles using RACF GEJBROLE profiles” on page 41.

Decide whether to make the use of BPC Explorer available to all authenticated users by making the EJBROLE profile UACC(READ), or make the profile UACC(NONE) and PERMIT a suitable group to the access list.

As shown in Example 2-38, in job MDEJBROL we permitted all the groups with administrator roles in the BPC and Human Task Containers (MDBPESMG,MDBPEADG,MDHTSMG,MDHTADMG) to the role WebClientUser.

In addition, in job MDRACF we defined a specific group called MDWCUG (WCU=WebClientUser) to which other user IDs can be connected in order to be given the role WebClientUser. Therefore, the group MDWCUG was on the access list for the profile.

If you prefer to use GEJBROLE see job GEJBROLE which assigned the WebClientUser role to the profile that contained administrator roles.

10. Select the BPCEXplorer application check box, and click **Start**, as shown in Figure 2-69. Note that its display name includes its cluster name.

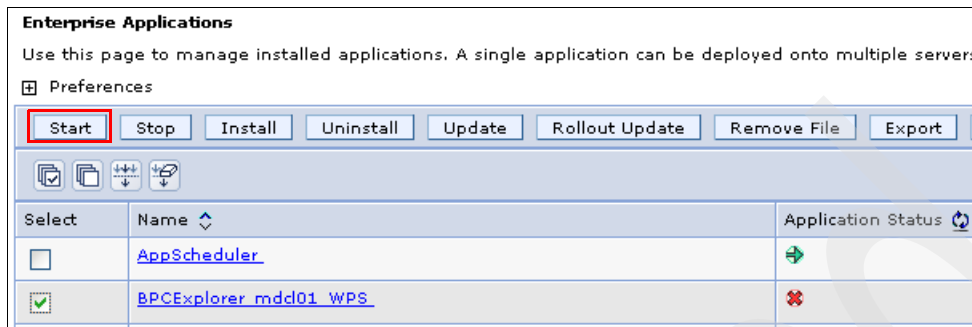


Figure 2-69 Start the BPCEXplorer application

11. Launch the BPC Explorer using a URL similar to the following:

`http://<hostname>:<port>/bpc`

**Note:** It is suggested to use a different browser from the one you may have connected to the Integrated Solutions Console. If you use a new tab on that browser to connect to BPC, the LTPA token of the WebSphere administrative user may be accepted and you will end up logged into BPC Explorer with the WebSphere administrative user ID rather than the one you want to use.

12. Log in using the user ID and password of the BPC Administrator user ID. This is the user ID you configured as the BPE Administrator in Figure 2-55 on page 149. For our MD cell this was MDBPEADM.

**Note:** Remember that when RACF APPL class is active you will need to permit all user IDs, including the BPE Administrator user ID, to the APPL profile for the cell's security domain.



13.If the login is successful you will see the BPC Explorer menu (Figure 2-70).  
You will use this later when verifying the installation.

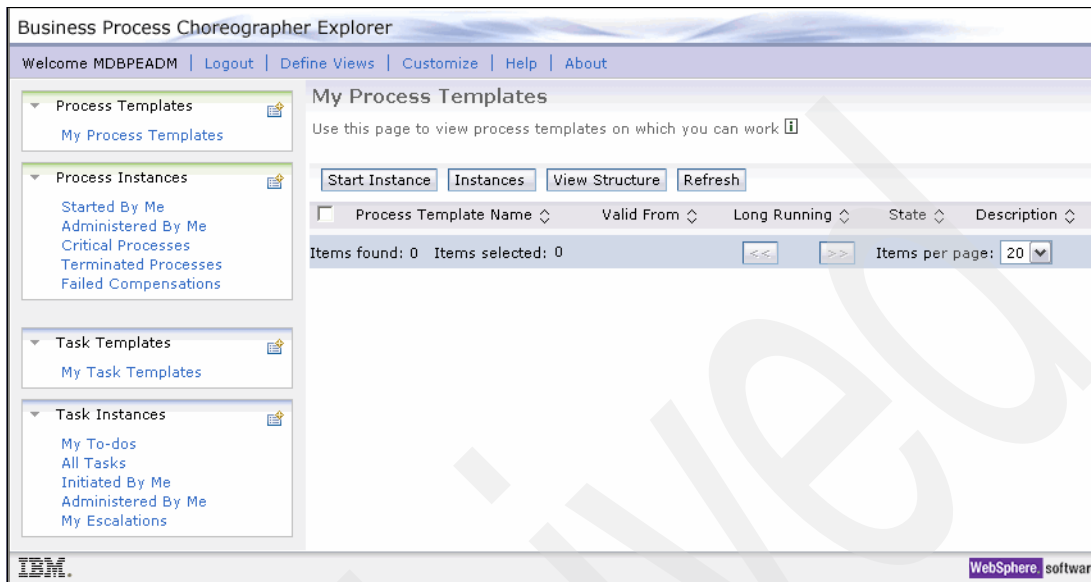


Figure 2-70 The BPC Explorer menu

There is not much you can do with BPC Explorer at this point but it will be used later once some applications have been deployed.

## 2.17 Configuring BPC Event Collector and BPC Observer

The BPC Event Collector and the BPC Observer can be configured using the Integrated Solutions Console or a by running shell scripts. We describe how to use the wizards provided in the Integrated Solutions Console.

The BPC Observer depends on the BPC Event Collector, which must be configured first.

BPC Event Collector and BPC Observer are part of the support infrastructure so when configuring the two-clusters topology you would configure them both in the Support and Messaging cluster.

Perform the following steps to configure BPC Event Collector and BPC Observer:

1. Install Event Collector using the Integrated Solutions Console.  
See page 160.
2. Installing Event Collector and BPC Observer using scripts.  
See page 164.
3. Install BPC Observer using the Integrated Solutions Console.  
See page 164.

### 2.17.1 Install Event Collector using the Integrated Solutions Console

Perform the following steps to install Event Collector using the Integrated Solutions Console.

1. Log in to the Integrated Solutions Console and navigate to **Servers** → **Clusters** → **cluster\_name**.  
  
In the two-clusters topology, choose the Support and Messaging cluster because BPC Event Collector and BPC Observer are considered part of the support infrastructure.
2. Under Business Integration, expand **Business Process Choreographer**, and click **Business Process Choreographer Event Collector** (Figure 2-71).

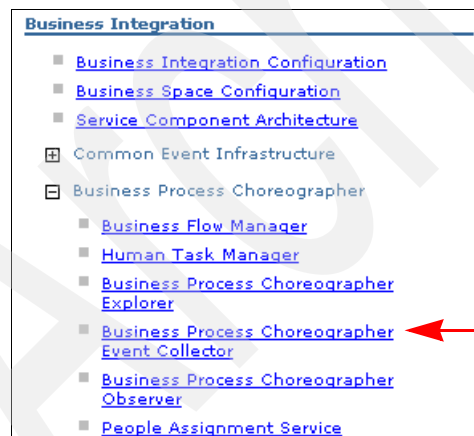
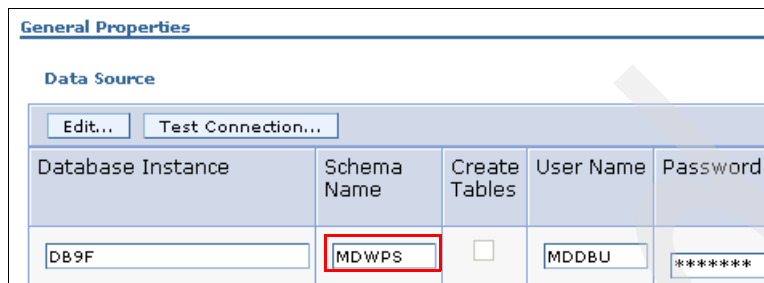


Figure 2-71 Launch the BPC event collector wizard

3. In the Data Source table, set the correct schema name and ensure that the Database Instance field is set to the location name of your DB2 for z/OS system (Figure 2-72).



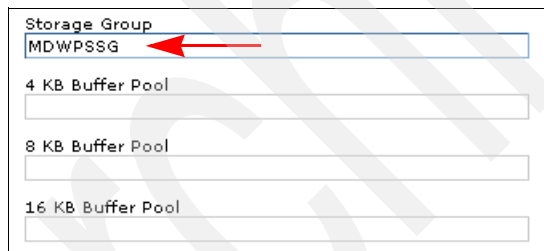
The figure shows a 'General Properties' dialog box with a 'Data Source' tab. It contains a table with five columns: Database Instance, Schema Name, Create Tables, User Name, and Password. The Database Instance field contains 'DB9F', the Schema Name field contains 'MDWPS' (highlighted with a red box), the Create Tables checkbox is unchecked, the User Name field contains 'MDDBU', and the Password field contains '\*\*\*\*\*'. There are 'Edit...' and 'Test Connection...' buttons at the top of the table.

Database Instance	Schema Name	Create Tables	User Name	Password
DB9F	MDWPS	<input type="checkbox"/>	MDDBU	*****

Figure 2-72 Set the schema name for BPC Event Collector

4. Click **Edit**, set the correct Storage Group name, and click **OK** as shown in Figure 2-73.

You cannot set buffer pools here. The storage group name is only used in the generated DDL. If you have already created all the databases and tables using `createdB.sh`, the only value in setting the storage group name here is for documentation purposes.



The figure shows a 'Storage Group' dialog box. It has a 'Storage Group' field containing 'MDWPSSG' (highlighted with a red arrow). Below it are three fields for buffer pools: '4 KB Buffer Pool', '8 KB Buffer Pool', and '16 KB Buffer Pool', all of which are empty.

Storage Group	MDWPSSG
4 KB Buffer Pool	
8 KB Buffer Pool	
16 KB Buffer Pool	

Figure 2-73 Set the storage group name

5. Make a selection from the radio buttons listed under Observation Target. Click **OK**. We decided to observe all events in the Business Process Container but you can specify an event group to limit the events that the Observer monitors (Figure 2-74).

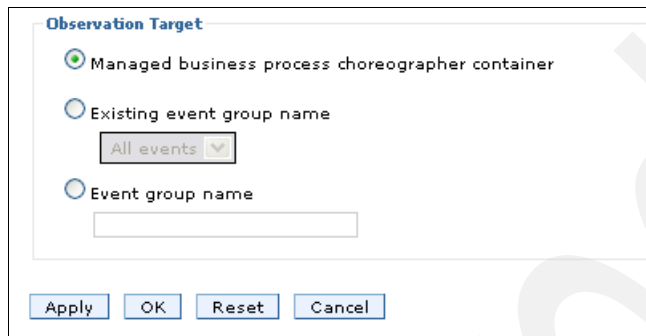
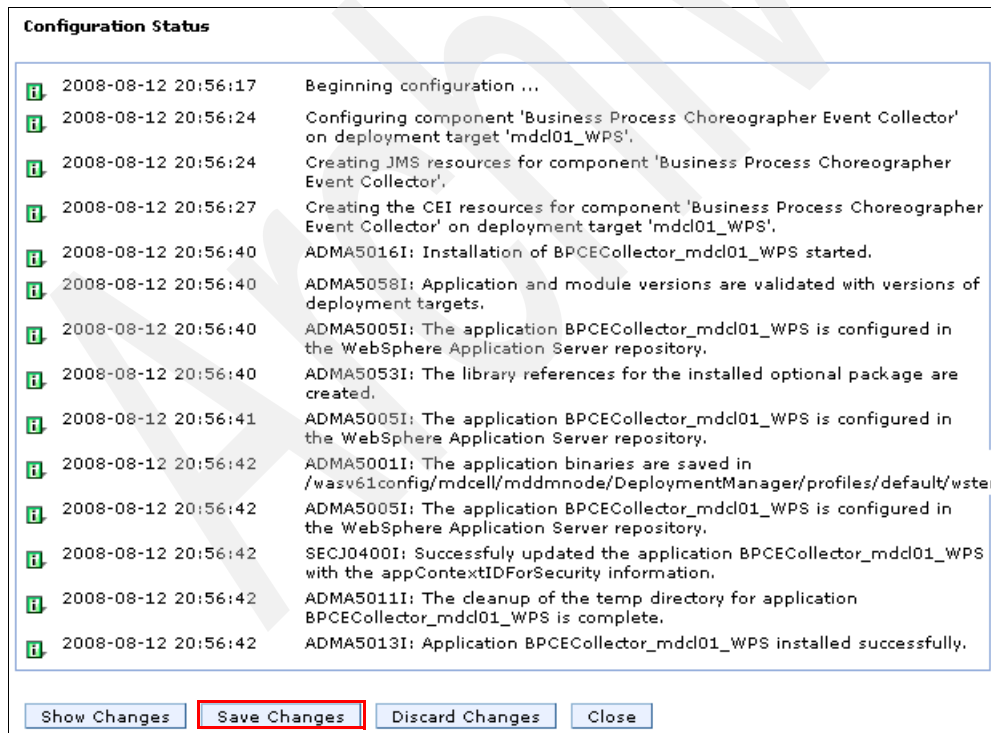


Figure 2-74 Set the Observation target

6. Check that the Event Collector deploys with no errors as shown in Figure 2-75 and click **Save Changes**.



Timestamp	Message
2008-08-12 20:56:17	Beginning configuration ...
2008-08-12 20:56:24	Configuring component 'Business Process Choreographer Event Collector' on deployment target 'mdd01_WPS'.
2008-08-12 20:56:24	Creating JMS resources for component 'Business Process Choreographer Event Collector'.
2008-08-12 20:56:27	Creating the CEI resources for component 'Business Process Choreographer Event Collector' on deployment target 'mdd01_WPS'.
2008-08-12 20:56:40	ADMA5016I: Installation of BPCECollector_mdd01_WPS started.
2008-08-12 20:56:40	ADMA5058I: Application and module versions are validated with versions of deployment targets.
2008-08-12 20:56:40	ADMA5005I: The application BPCECollector_mdd01_WPS is configured in the WebSphere Application Server repository.
2008-08-12 20:56:40	ADMA5053I: The library references for the installed optional package are created.
2008-08-12 20:56:41	ADMA5005I: The application BPCECollector_mdd01_WPS is configured in the WebSphere Application Server repository.
2008-08-12 20:56:42	ADMA5001I: The application binaries are saved in /wasv61config/mdcell/mddmnode/DeploymentManager/profiles/default/wster
2008-08-12 20:56:42	ADMA5005I: The application BPCECollector_mdd01_WPS is configured in the WebSphere Application Server repository.
2008-08-12 20:56:42	SECJ0400I: Successfully updated the application BPCECollector_mdd01_WPS with the appContextIDForSecurity information.
2008-08-12 20:56:42	ADMA5011I: The cleanup of the temp directory for application BPCECollector_mdd01_WPS is complete.
2008-08-12 20:56:42	ADMA5013I: Application BPCECollector_mdd01_WPS installed successfully.

Figure 2-75 Check the deployment messages for errors and then click Save Changes

7. Set the currentSQLID property or currentSchema property in the Custom properties of the BPC Event Collector data source datasource.

Navigate to **Resources** → **JDBC** → **Data sources** → **Business Process Choreographer Event Collector data source** and click **Custom properties** (Figure 2-76).

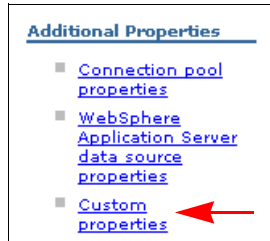


Figure 2-76 Click Custom properties

8. Scroll and locate the property currentSQLID or currentSchema then click it. Figure 2-77 shows currentSQLID being selected.

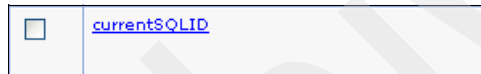


Figure 2-77 Click property currentSQLID

9. Set the database schema name as the value in currentSQLID or currentSchema as shown in Figure 2-78, and then click **OK**.

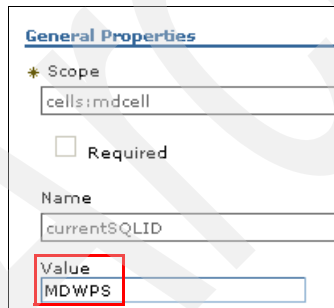


Figure 2-78 Set the schema in currentSQLID

10. Stop and restart the cluster.

The Event Collector application cannot be started using the Integrated Solutions Console at this point because the changes to the data source do not take affect until the servers are restarted. The Event Collector application starts normally when the servers next restart.

## 2.17.2 Installing Event Collector and BPC Observer using scripts

Although we used the wizards provided in the Integrated Solutions Console to configure Event Collector, we had to use the `setupEventCollector.sh` shell script to reduce the thresholds that controlled when event collector collects events. This was necessary in order to verify the function of BPC Event Collector and BPC Observer.

We do not describe in detail the use of the `setupEventCollector.sh` for configuring Event Collector. However, you can read about how to invoke `setupEventCollector.sh`, and how to use it, in Section 2.19.1, “Change Event Collector thresholds” on page 177.

The BPC Observer can be configured using a shell script `setupObserver.sh`, which operates in a similar way to `setupEventCollector`.

## 2.17.3 Install BPC Observer using the Integrated Solutions Console

Perform the following steps to install BPC Observer using the Integrated Solutions Console.

1. Log in to the Integrated Solutions Console and navigate to **Servers** → **Clusters**.
2. Click the cluster where you want to configure BPC Observer.

In the two-clusters topology, click the Support and Messaging cluster because BPC Event Collector and BPC Observer are considered part of the support infrastructure.

3. Under Business Integration, expand **Business Process Choreographer**, and click **Business Process Choreographer Observer** as shown on Figure 2-79.

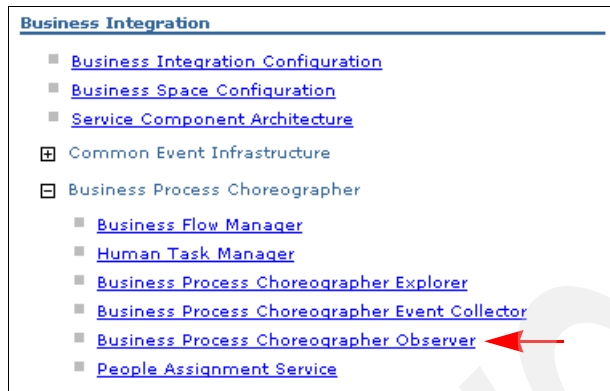


Figure 2-79 launch the BPC Observer wizard

4. Set the context root or accept the default as shown in Figure 2-80.  
Map the BPC Observer to the correct cluster by choosing the cluster in the list box. Click **OK**.

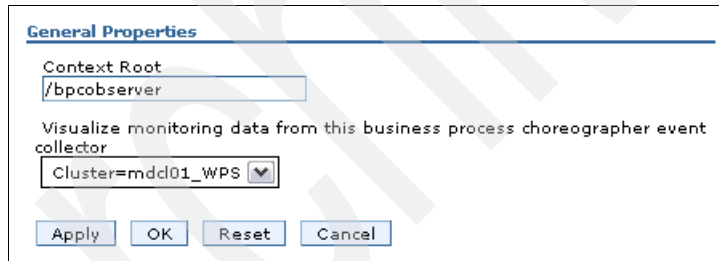


Figure 2-80 Set BPC Observer context root and map to a cluster

5. Check that the BPC Observer deploys with no errors, and click **Save Changes** as shown in Figure 2-81.

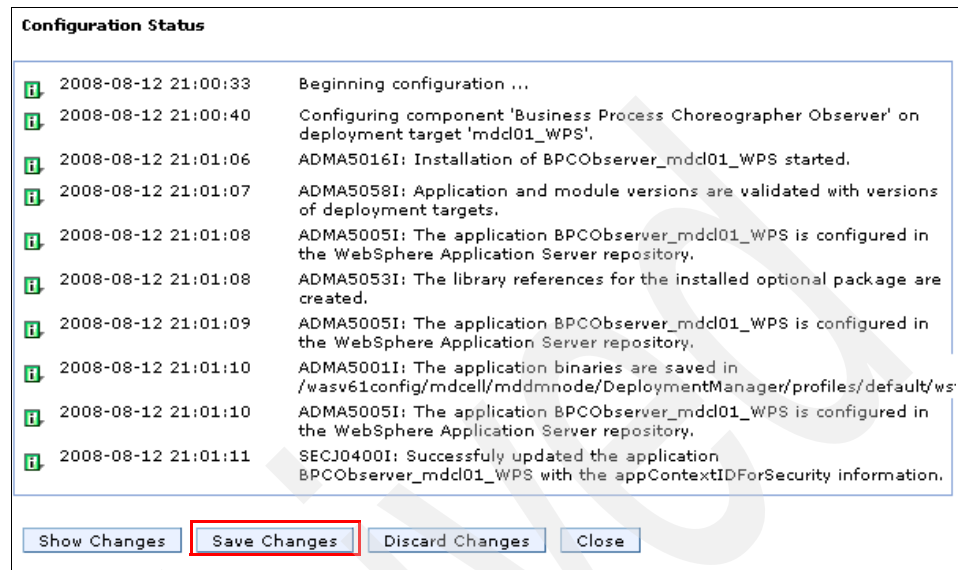


Figure 2-81 Check the deployment messages for errors and then click Save Changes

6. Start the BPC Observer application (Figure 2-82).

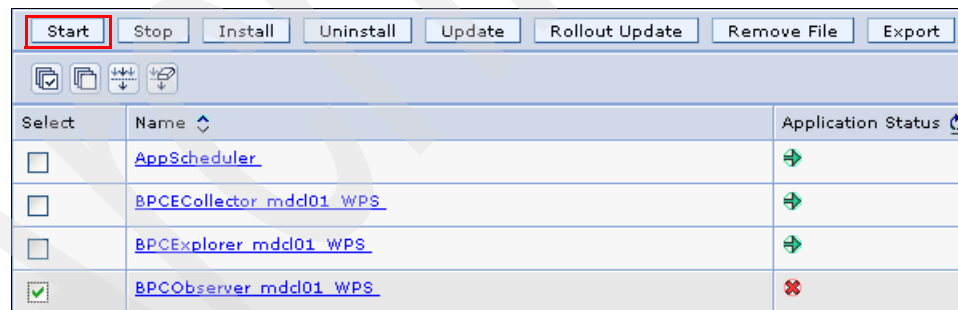


Figure 2-82 Start the BPC Observer

7. When using RACF to perform EJBROLE authorization checking, you must define the role ObserverUser to RACF.

Decide whether to make the use of BPC Observer available to all authenticated users by making the EJBROLE profile UACC(READ), or make the profile UACC(NONE) and PERMIT a suitable group to the access list.



As shown in Example 2-38 on page 157, in job MDEJBROL we permitted a specific group called MDOBSUG to which other user IDs can be connected in order to be given the role ObserverUser. Therefore the group MDOBSUG is on the access list for the profile.

*Example 2-39 Extract of sample job MDEJBROL showing the commands to define EJBROLE ObserverUser for BPC Observer*

---

```

/* Define BPC Observer User role */
RDELETE EJBROLE MD.ObserverUser
RDEFINE EJBROLE MD.ObserverUser
                                UACC(NONE)
                                OWNER(WASCFG)
                                DATA('MD BPC Observer user')
PERMIT MD.ObserverUser
                                CLASS(EJBROLE)
                                RESET
PERMIT MD.ObserverUser
                                CLASS(EJBROLE)
                                ID(MDOBSUG)
                                ACCESS(READ)

```

---

If you prefer to use GEJBROLE profiles, see job GEJBROLE which assigned the ObserverUser role to the profile protecting administrator roles.

## 2.18 Verify BPC

After the BPC and Human Task containers have been configured, perform a quick installation verification test (IVT).

We used the bpcivt and the ZOS Book Order applications for testing. The Installation Verification Test procedure is described in the White Paper *Performing Installation Verification for WPS on z/OS V6.1*, WP101218, which you can download from the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101218>

White Paper WP101218 describes how to install and run the ZOSBookOrderApp. Therefore that information is not repeated here. However, the White Paper does not include a procedure for verifying the BPC Event Collector and BPC Observer.

In this section, then, we briefly describe how to install and run bpcivt, noting some issues we experienced. Then, in Section 2.19, “Verify BPC Event Collector and BPC Observer” on page 176, we describe how to verify that the Event Collector and BPC Observer are working properly.

Perform the following steps to install and run bpcivt.

1. Install the bpcivt application.  
See page 168.
2. Enable CSiv2 Asserted Identity.  
See page 171.
3. Run bpcivt and ZOSBookOrderApp.  
See page 175.

### 2.18.1 Install the bpcivt application

You can deploy the application taking all of the defaults. There may be some warning messages issued during the deployment process that relate to missing activation specification definitions but these can be ignored.

1. Navigate to **Applications** → **Install New Application** and click **Remote file system**, as shown in Figure 2-83.

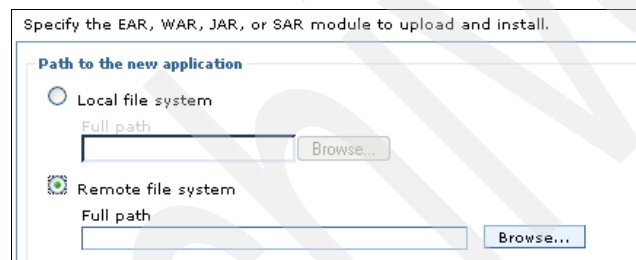


Figure 2-83 Select to install from Remote file system

- Click the deployment manager's node directory, navigate up to the root directory, /, and then through the z/OS UNIX file system to locate the InstallableApps directory under the deployment manager's node.

Click the radio button to the left of bpcivt.ear as shown in Figure 2-84, and then click **OK**.



Figure 2-84 Locate and select the bpcivt.ear

- Click **Next**.
- In the Step 1: Select installation options window, click **Next**.
- In the Step 2: Map modules to servers window, check that the two modules are mapped to the WebSphere Process Server cluster, and then click **Next**.

Because only the WebSphere Process Server cluster has been created in the cell so far, the mapping will be correct (Figure 2-85).

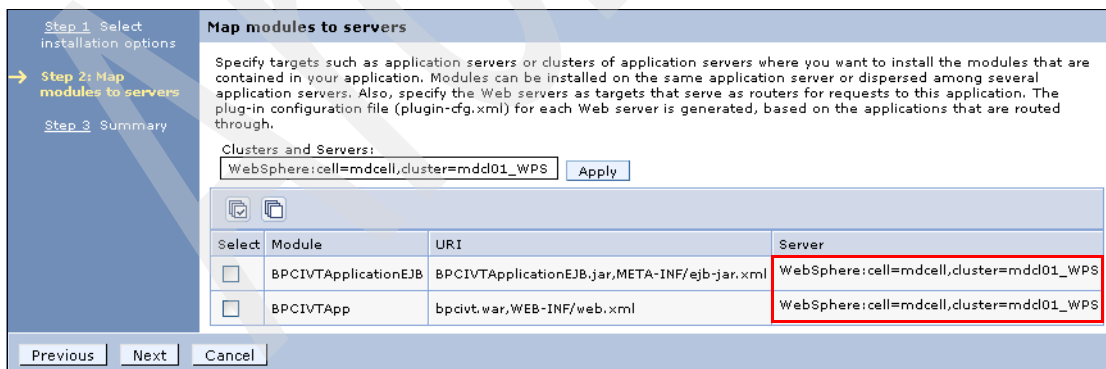


Figure 2-85 Check that the modules are mapped to the correct cluster

- In the Step 3: Summary window, click **Finish**.

7. Check that the application deploys successfully (Figure 2-86), and click **Save**.

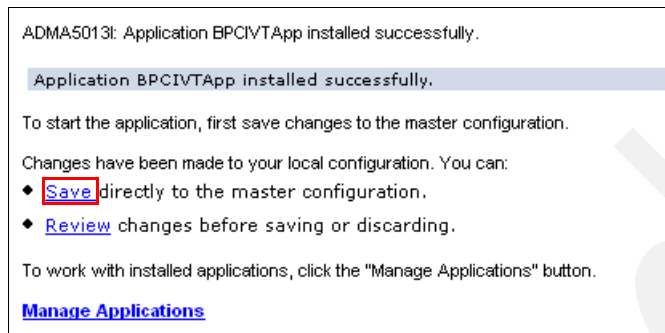


Figure 2-86 Check the application deploys successfully and click **Save**

8. Navigate to **Applications** → **Enterprise applications**, select the bpcivt application as shown in Figure 2-87, and click **Start**.

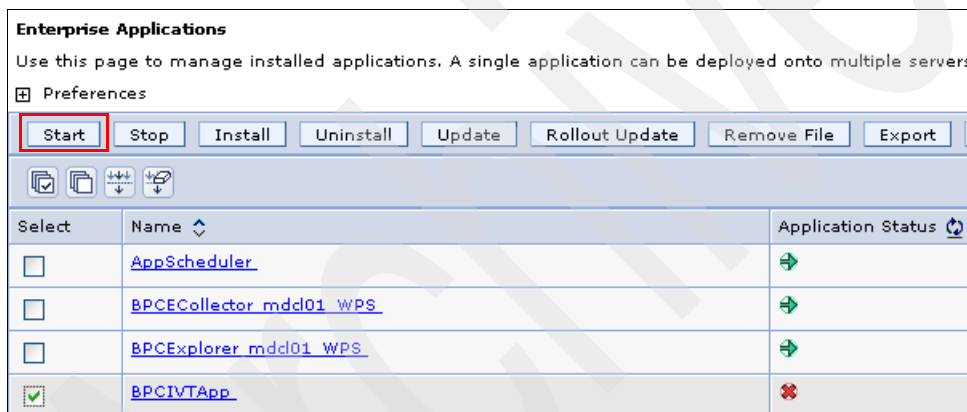


Figure 2-87 Start the bpcivt application

The application will start successfully as shown in Figure 2-88.

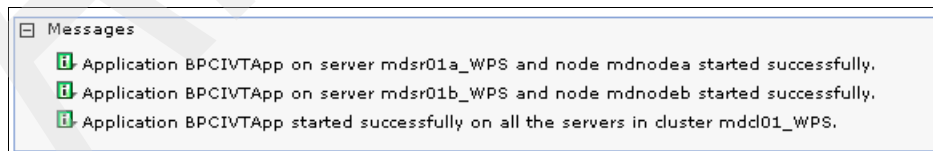


Figure 2-88 Successful start of bpcivt

If you receive errors starting the bpcivt application, check the logs of the WebSphere Process Server servant regions for error messages.

The application list looks like Figure 2-89. Also, if you list the SCA Modules you see the BPCIVTApplication listed and started.

Name 	Application Status 
<a href="#">AppScheduler</a>	
<a href="#">BPCECollector mdc01 WPS</a>	
<a href="#">BPCEExplorer mdc01 WPS</a>	
<a href="#">BPCIVTApp</a>	
<a href="#">BPCEObserver mdc01 WPS</a>	
<a href="#">BPCEContainer mdc01 WPS</a>	
<a href="#">HTM_PredefinedTaskMsg V612 mdc01 WPS</a>	
<a href="#">HTM_PredefinedTasks V612 mdc01 WPS</a>	
<a href="#">RemoteAL61</a>	
<a href="#">TaskContainer mdc01 WPS</a>	
<a href="#">persistentLkMqr</a>	
<a href="#">sca.sib.mediation</a>	
<a href="#">wpsFEMqr 6.1.2</a>	

Figure 2-89 List of applications all showing started status

## 2.18.2 Enable CSlv2 Asserted Identity

**Note:** If you are configuring a single-cluster topology skip this step. CSlv2 only needs to be configured if you are not running in a single cluster.

When running the bpcivt in a multi-cluster scenario we received a JNDI lookup error caused by the fact that CSlv2 asserted identity was not enabled. This problem is described in Section 4.3.2, “NMSV0610I: NamingException, CORBA.NO\_PERMISSION, The WSCredential does not contain a forwardable token ... enable Identity Assertion” on page 321.

Perform the following steps to enable CSiv2 Asserted Identity:

1. Log in to the Integrated Solutions Console and navigate to **Security** → **Secure administration, applications, and infrastructure**.
2. Expand RMI/IIOP security under Authentication and click **CSiv2 inbound authentication** as shown in Figure 2-90.



Figure 2-90 Select CSiv2 inbound authentication

3. Select the Identity Assertion check box and click **OK**. The other fields shown in Figure 2-91 will already be selected as shown.

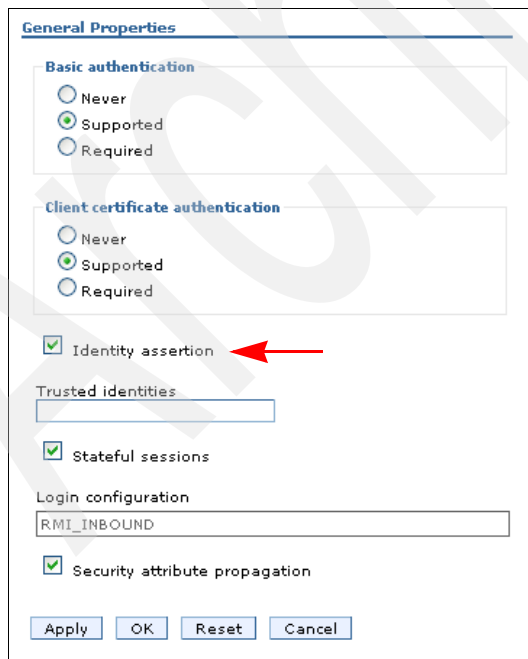


Figure 2-91 Enabling CSiv2 inbound authentication

4. Expand **RMI/IIOP security** under Authentication and click **CSlv2 inbound authentication** again.
5. Click **z/OS Additional Settings** under Additional properties on the right.
6. Select the SAF identity assertion check box as shown in Figure 2-92.

**Note:** SAF identity assertion is appropriate for a cell that is using a SAF-based user registry such as RACF.

If you are using an LDAP user registry you would need to choose Distinguished name identity assertion or Certificate identity assertion.

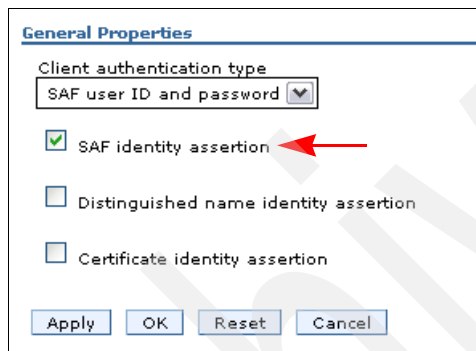


Figure 2-92 Setting the authentication type

7. Click **CSlv2 outbound authentication** as shown in Figure 2-93.



Figure 2-93 Click CSlv2 outbound authentication

8. Select the Identity assertion check box as shown in Figure 2-94.

The **Use server trusted identity** option is already selected. This is probably what you will want to choose.

**General Properties**

**Basic authentication**

☐ Never  
☒ Supported  
☐ Required

**Client certificate authentication**

☒ Never  
☐ Supported  
☐ Required

☒ Identity assertion →

☒ Use server trusted identity  
☐ Specify an alternative trusted identity

Trusted identity

Password

Confirm password

☒ Stateful sessions

Login configuration

☐ Custom outbound mapping

☒ Security attribute propagation

Trusted target realms

Apply OK Reset Cancel

Figure 2-94 Enabling CSiv2 outbound authentication

9. Click **OK**. Save the configuration changes and restart the clusters.

**Note:** In this section we showed the minimum steps to make bpcivt work. You may need to consider other CSiv2 options in your environment.



## 2.18.3 Run bpcivt and ZOSBookOrderApp

Detailed instructions for running bpcivt can be found in the WebSphere Process Server for z/OS Information Center and in White Paper WP101218. Here we summarize the process and highlight some issues we experienced.

1. Launch the bpcivt application: `http://<host>:<port>/bpcivt`.

The `<host>` variable applies to the BPC cluster and `<port>` is the `wc_default` port. For our MD cell the URL was `http://wpsplex.itso.ibm.com:20048/bpc`.

**Note:** If you receive: Error 403: AuthorizationFailed, see the explanation and solution in Section 4.3, “Errors when launching bpcivt” on page 320.

2. Log in when prompted with a user that has been permitted to the role BPCIVTUser. The bpcivt program will issue the messages shown in Example 2-40 if the test is successful.

---

### *Example 2-40 Successful test messages*

---

```
Looking up the HumanTaskManager API EJB...
Querying the originating task template...
Creating the input message...
Creating and starting the originating task...
Querying the participating task...
Participating task not yet created, waiting 1s...
Participating task not yet created, waiting 1s...
. .
. .
Participating task not yet created, waiting 1s...
Claiming the participating task...
Checking the input data...
Completing the participating task...
Checking the output data...
Output message of starter oTask not yet available, waiting 1s...
Output message of starter oTask not yet available, waiting 1s...
Deleting the originating task ...
Passed
Done.
```

---

**Notes:**

1. It is possible that if your system is suffering poor performance then the first time you run bpcvlt it may fail to create, claim, or finish a task within the timeout period in the application.

If you rerun the bpcvlt a second time, less time is taken loading the execution environment and the test completes successfully.

2. When security is enabled using RACF and you are testing using an application like ZOSBookOrder that is not secure (it does not prompt you to log in when you invoke it), the test will execute under the unauthenticated guest user ID.

The guest user ID is defined to RACF as a RESTRICTED user ID, which means you must explicitly permit that user ID to any profiles that are UACC(READ). For example, we had to explicitly permit MDGUEST to the profile MD.eventCreator.

## 2.19 Verify BPC Event Collector and BPC Observer

This section describes how to verify that the BPC Event Collector and BPC Observer are working properly. The BPC Event Collector component provides events to the BPC Observer, so when you verify the BPC Observer you are also verifying the Event Collector.

In order to verify BPC Observer, perform the following steps:

1. “Change Event Collector thresholds” on page 177.
2. “Enable CEI Event Logging” on page 179.

In order for the CEI events to be created you must have enabled CEI trace for WBILocationMonitor.CEI.SCA.\* and WBILocationMonitor.CEI.bpc.\*.

3. “Start WBILocationMonitor.CEI traces” on page 180.

If using the Java User-defined functions, start the DB2 stored procedure address space that was the one set up to run the java user-defined function.

4. “Stop and start the BPCECollector application” on page 182.
5. “Generate some events using a sample application” on page 182.
6. “Run the BPC Observer in a browser” on page 183.
7. “Back up the WebSphere configuration HFS datasets” on page 200

## 2.19.1 Change Event Collector thresholds

Before you can verify that the Event Collector and BPC Observer are functioning correctly, you will need to run the `setupEventCollector.sh` to reduce some of the thresholds that control when events are made available to the BPC Observer. If you do not reduce the thresholds, it is probable that no events are listed when you get to the point where you test the BPC Observer.

We logged in to a telnet session with the WebSphere administrator user ID, MDADMIN and then executed `setupEventCollector.sh`. The log of the telnet session is shown in Example 2-41. The initial commands to set up the environment and invoke `setupEventCollector.sh` are high-lighted in bold. Our responses to the shell's prompts are high-lighted in reverse text.

*Example 2-41 Telnet log of using `setupEventCollector` to reduce thresholds*

```
MDADMIN @ SC42:/SC42/var/WebSphere/home/MDCFG>\
export WAS_HOME=/wasv61config/mdcell/mddmnode/DeploymentManager/profiles/default
MDADMIN @ SC42:/SC42/var/WebSphere/home/MDCFG>\
cd /wasv61config/mdcell/mddmnode/DeploymentManager/ProcessChoreographer/config
MDADMIN @ SC42:/wasv61config/mdcell/mddmnode/DeploymentManager/ProcessChoreographer/config>\
./setupEventCollector.sh -conntype SOAP -user MDADMIN -password MDADMIN -cluster mdc101_WPS
WASX7209I: Connected to process "dmgr" on node mddmnode using SOAP connector; The type of
process is: DeploymentManager
WASX7303I: The following options are passed to the scripting environment and are available as
arguments that are stored in the argv variable: "[-cluster, mdc101_WPS]"
-----
Welcome to the WebSphere Business Process Choreographer Event Collector setup!
-----
Logfile is
'/wasv61config/mdcell/mddmnode/DeploymentManager/profiles/default/logs/setupEventCollector.log'.
Initializing ...
Commands Menu

    1) Prepare a database for the Event Collector and Observer
    2) Install the Event Collector application
    3) Remove the Event Collector application and related objects
    4) Change configuration settings of an installed Event Collector
    5) Drop the database schema of the Event Collector and Observer
    6) Administer Observer related user-defined functions

    0) Exit Menu

Your selection: 4
==> Change configuration settings of an installed Event Collector

Select the deployment target where the application is installed:

    1) Cluster 'mdc101_WPS'
    0) Exit Menu

Your selection: [1]
==> Cluster 'mdc101_WPS'

Retrieving the application configuration settings of application 'BPCECollector_mdc101_WPS' ...

Enter the number of the configuration parameter you want to change/display:

    1) BPCEventTransformerEventCount
```

```

2) BPCEventTransformerMaxWaitTime
3) BPCEventTransformerToleranceTime
4) ObserverCreateTables
5) ObserverSchemaName

0) Exit Menu

Your selection: 1

Edit a configuration parameter.

Configuration parameter: BPCEventTransformerEventCount
Description:             The number of events after which the Event
                          Collector sends a notification to the Transformer.
Data type:               Integer
Unit:                   Events
Current value is:        '500'

Enter a new value. Press 'Enter' to exit.
Your selection: 1

==> Value for 'BPCEventTransformerEventCount' set to '1'.

Enter the number of the configuration parameter you want to change/display:

1) BPCEventTransformerEventCount
2) BPCEventTransformerMaxWaitTime
3) BPCEventTransformerToleranceTime
4) ObserverCreateTables
5) ObserverSchemaName

0) Exit Menu

Your selection: 2

Edit a configuration parameter.

Configuration parameter: BPCEventTransformerMaxWaitTime
Description:             Time in minutes after the Transformer is notified
                          although the number of events is not reached.
Data type:               Integer
Unit:                   Minutes
Current value is:        '10'

Enter a new value. Press 'Enter' to exit.
Your selection: 1

==> Value for 'BPCEventTransformerMaxWaitTime' set to '1'.

Enter the number of the configuration parameter you want to change/display:

1) BPCEventTransformerEventCount
2) BPCEventTransformerMaxWaitTime
3) BPCEventTransformerToleranceTime
4) ObserverCreateTables
5) ObserverSchemaName

0) Exit Menu

Your selection: 3

Edit a configuration parameter.

Configuration parameter: BPCEventTransformerToleranceTime
Description:             Time in minutes while the Transformer ignores the
                          events in the database.
Data type:               Integer

```

```

Unit:                               Minutes
Current value is:                    '10'

Enter a new value. Press 'Enter' to exit.
Your selection: 1

==> Value for 'BPCEventTransformerToleranceTime' set to '1'.

Enter the number of the configuration parameter you want to change/display:

1) BPCEventTransformerEventCount
2) BPCEventTransformerMaxWaitTime
3) BPCEventTransformerToleranceTime
4) ObserverCreateTables
5) ObserverSchemaName

0) Exit Menu

Your selection: 0

Do you want to save the changes?
y) yes
n) no
Your selection: [y] Y
==> yes
Updating the application configuration settings ...

Note: To activate the changes, you must restart the application BPCECollector_mdcl01_WPS.
Information: Please run the node synchronization (run syncNode.bat|.sh) to activate the changes.

```

---

## 2.19.2 Enable CEI Event Logging

If you did not enable CEI logging when configuring the BPC and Human Tasks Containers you need to do it now to create the events that the event collector is to collect and make available to BPC Observer.

If you configured BPC before CEI, even though you may think that CEI logging is enabled, it may not be. Enable CEI logging for both BPC and Human Task containers.

1. Perform the following steps to enable CEI logging using the Integrated Solutions Console.
  - a. Log in to the WebSphere Administration Console  
<http://zt01.mop.tec.com:20218/ibm/console>.
  - b. Navigate to **Servers** → **Clusters** and click the cluster that is hosting BPC. For our single-cluster topology that was mdcl01\_WPS. For our two-clusters topology it was mpcl02.AppTarget.
  - c. Under Container settings, expand **Business Process Choreographer Container Settings**, and click **Business Process Choreographer Containers**.

- d. Confirm that CEI logging is enabled under State Observers as shown in Figure 2-95.

State Observers		
Logging	Business Flow Manager	Human Task Manager
Audit Logging	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Common Event Infrastructure Logging	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 2-95 Verifying that CEI logging is enabled for BPC and Human Tasks

- e. If logging is not enabled, select the check boxes to enable CEI logging and click **OK**.
- f. Save the configuration change and synchronize to the nodes.

**Note:** If you configured BPC before configuring CEI you may not succeed in having BPC event logged by checking these boxes after BPC is configured. We found that we had to uninstall the BPC Event Collector and re-install it using the `setupEventCollector`. During the re-installation of the Event Collector we were prompted to enable Event logging. After doing this, Event Collector started to collect events.

See Section 4.4.2, “No events displayed in BPC Observer; no errors in logs” on page 327 for details of re-installing `setupEventCollector`.

### 2.19.3 Start WBILocationMonitor.CEI traces

You need to enable the WBI traces to cause CEI events to be emitted from various trace points if the sample application does not use CEI itself. The `bpcivt` application itself does not use CEI, for example.

**Tip:** When you have a cluster, it is easier to verify Event Collector and Observer if you only have one cluster member running because you only need to enable traces in one cluster member.

Perform the following steps to start WBILocationMonitor.CEI traces.

1. Log in to the Integrated Solutions Console and perform the following steps to enable tracing for WBILocationMonitor.CEI.bpc.\*.
  - a. Navigate to **Troubleshooting** → **Logs and trace**.
  - b. Click the link to the first cluster member.
  - c. Click **Change Log Detail Levels**.
  - d. Click the Runtime tab. You will have to wait a few moments for the component list to initialize.

You should still have the trace for WBILocationMonitor.CEI.SCA.\* started from when you verified SCA.

- e. Locate, then click the component **WBILocationMonitor.CEI.bpc.\***. Select **All** as shown in Figure 2-96.

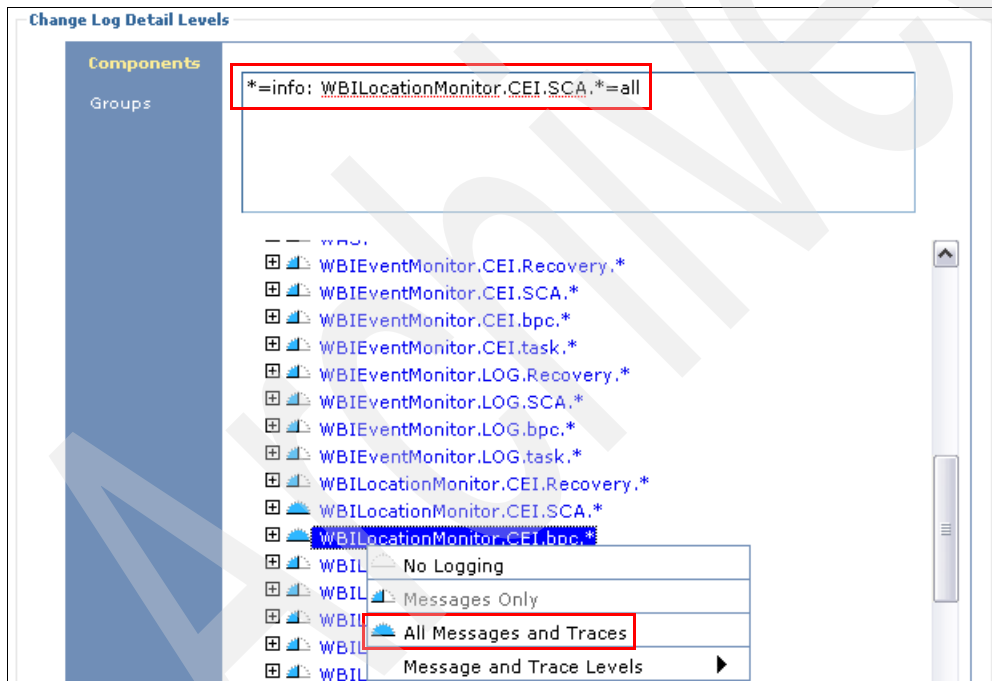


Figure 2-96 Enabling WBILocationMonitor traces

The trace entry is added into the box at the top of the window and resembles the following text:

```
*=info: WBILocationMonitor.CEI.SCA.*=all: WBILocationMonitor.CEI.bpc.*=all
```

- f. Click **OK** to start the trace.

2. By default, Event Collector uses a non-Java user defined function. If you decided to use the Java versions instead, start the DB2 stored procedure address space that supports them.

**Note:** For more information about choosing between the Java and non-Java user-defined function, see the WebSphere Process Server for z/OS Information Center article *Selecting between Java and SQL user-defined functions*, available at the following Web page.

[http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.websphere.bpc.z.612.doc/doc/bpc/t2config\\_obs\\_udfs.html](http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.websphere.bpc.z.612.doc/doc/bpc/t2config_obs_udfs.html)

There is also a detailed description of using the java user-defined functions in the WebSphere Process Server for z/OS V6.0.2 Redpaper *WebSphere Process Server for z/OS: Configuring a Network Deployment Environment*, REDP-4388. Much of this information still applies to V6.1.

## 2.19.4 Stop and start the BPCECollector application

The previous changes to enable CEI logging are not picked up by the Event Collector unless it is restarted.

1. Log in to the Integrated Solutions Console and navigate to **Applications** → **Enterprise Applications**.
2. Select the BPCECollector\_mdcl01\_WPS check box. Click **Stop**.
3. When the application has stopped, select the BPCECollector\_mdcl01\_WPS check box. Click **Start**.

## 2.19.5 Generate some events using a sample application

We tried to use the bpcivt application to generate events that BPC Observer can report on but we found that enabling the trace WBILocationMonitor.CEI.bpc.\*=all caused duplicate events to be created.

Therefore, we used the sample Fabric application described in Section 3.6, “Install and test the sample application” on page 282.

1. Follow the process described in Section 3.6.10, “Test the sample application” on page 300 by listing and instantiating a Business Process instance of the TestLoanProcess.

Creating a Business Process instance with traces enabled will cause CEI events to be logged and picked up by the BPC Event Collector.

2. Check the logs to verify that no errors are occurring when you initiate the process instance.



## 2.19.6 Run the BPC Observer in a browser

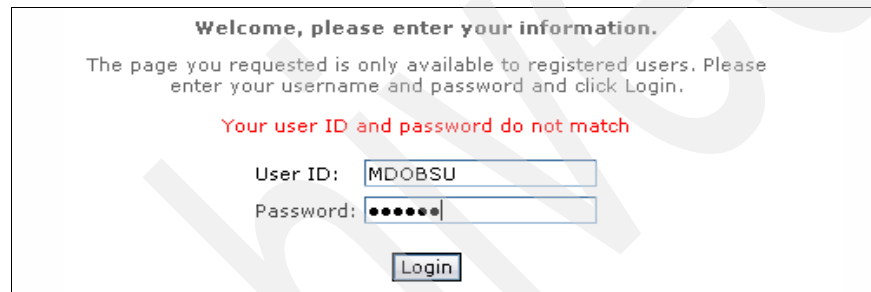
Perform the following steps to run the BPC Observer in a browser.

1. Open a browser and launch the BPCObserver.

The URL for our single-cluster topology was  
<http://wpsplex.itso.ibm.com:20048/bpcobserver>.

2. Log in with a user ID that is authorized to the ObserverUser role. We had created user ID MDOBSU and permitted its default group, MDOBSUG, to the EJBROLE profile MD.ObserverUser. We logged in with MDOBSU.

If you receive a “Your user ID and password do not match” message (Figure 2-97) when logging in to BPC Observer, it can be because you have RACF APPL class active but have not permitted the user MDOBSU or group MDOBSUG to the APPL profile.



Welcome, please enter your information.

The page you requested is only available to registered users. Please enter your username and password and click Login.

Your user ID and password do not match

User ID:

Password:

Figure 2-97 User ID and password do not match

3. If events have been collected, the Total number of events on the Welcome window will not be 0.

**Note:** If you see zero events in BPC Observer refer to article *Troubleshooting BPC Observer* in the WebSphere Process Server for z/OS Information Center. This article is available from the following Web page:

[http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=/com.ibm.websphere.bpc.z.612.doc/doc/bpc/r7observer\\_troubleshooting.html](http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=/com.ibm.websphere.bpc.z.612.doc/doc/bpc/r7observer_troubleshooting.html)

Also see the problems and solutions in Section 4.4, “Errors relating to BPC Event Collector and Observer” on page 325.

4. In Figure 2-98 on page 184, the menu on the left offers Lists, Charts, Process Reports and Activity Reports. The following sections describe how to verify these functions.

## Verify the BPC Observer Lists function

Perform the following steps to verify the BPC Observer Lists function.

1. On the Lists menu, click **Processes**, as shown in Figure 2-98.

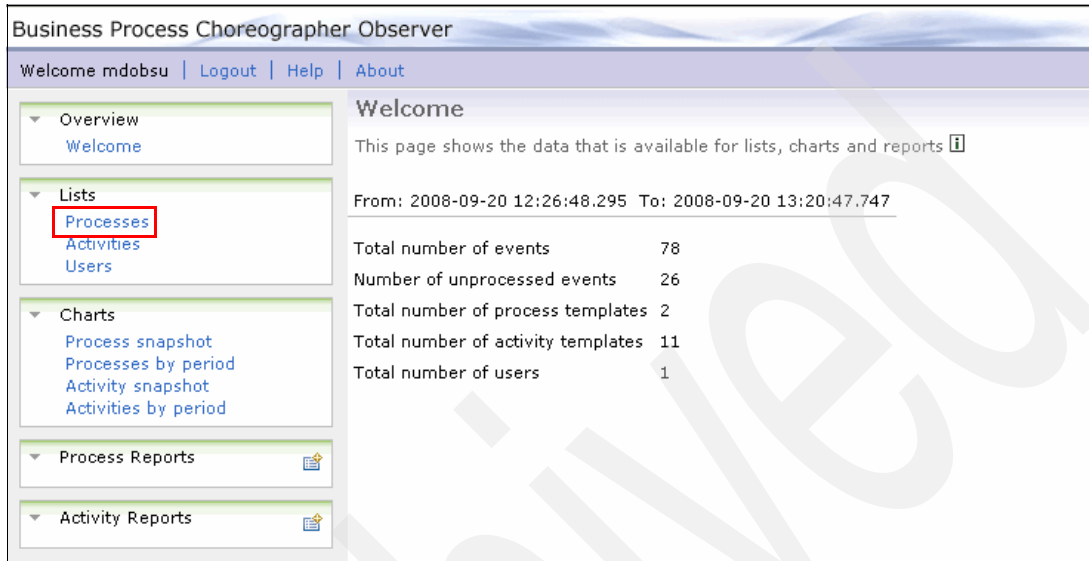
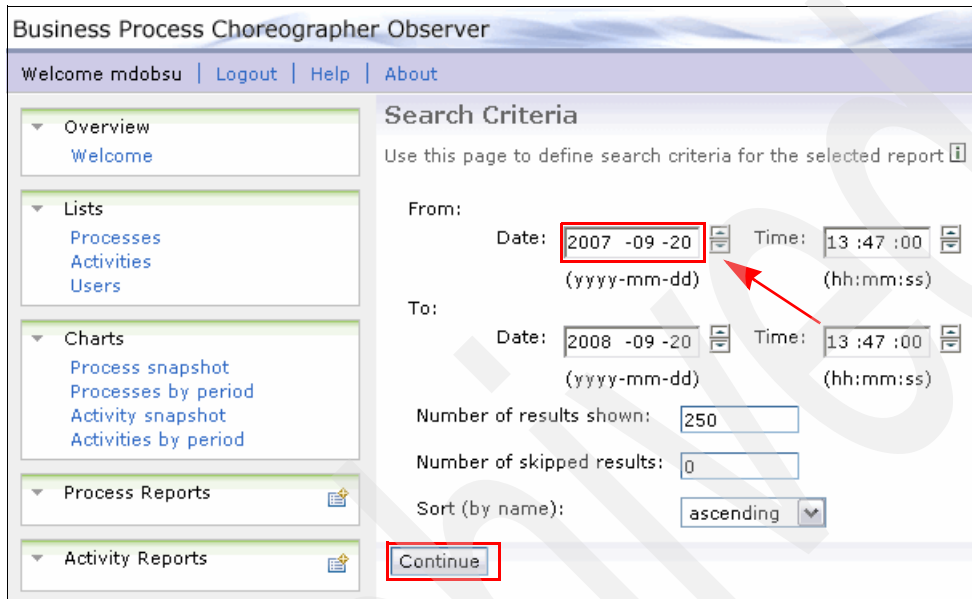


Figure 2-98 BPC Observer Welcome panel showing captured events

2. Set the search criteria by clicking the down arrow so the From date changes to 2007, as shown in Figure 2-99. Click **Continue**.

This is a much longer time range than you use when searching for real process events in a busy system, but it is a quick way to verify the function when there are only a few events.



The screenshot shows the 'Business Process Choreographer Observer' interface. On the left is a navigation menu with sections: Overview (Welcome), Lists (Processes, Activities, Users), Charts (Process snapshot, Processes by period, Activity snapshot, Activities by period), Process Reports, and Activity Reports. The main area is titled 'Search Criteria' and contains the text 'Use this page to define search criteria for the selected report'. It features 'From' and 'To' date and time pickers. The 'From' date is '2007 -09 -20' and the 'To' date is '2008 -09 -20'. Both times are '13 :47 :00'. Below these are input fields for 'Number of results shown' (250) and 'Number of skipped results' (0). A 'Sort (by name):' dropdown is set to 'ascending'. A 'Continue' button is at the bottom, highlighted with a red box. A red arrow points to the 'From' date field.

Figure 2-99 Changing From date to set the Search data range

One or more process templates is found and listed.

3. Select the check box next to one of these (we checked TestLoadProcess) and click **Instances snapshot**, as shown in Figure 2-100.

**Process Templates**

This page displays a list of process templates according to the search criteria [\[i\]](#)

Search Criteria: Time frame: 2007-09-20 13:47:00 - 2008-09-20 13:47:00; Sort: ascending; Skipped templates: 0; Max. total:250

Number of Process Templates found: 2

► Refine your Search Criteria

**Instances Snapshot**

<input type="checkbox"/>	Template Name	Valid From	instances	instances started	instances ended	avg duration in seconds
<input type="checkbox"/>	NewLoanProcess	2008-02-05 20:34:47.000	3	3	3	5
<input checked="" type="checkbox"/>	TestLoanProcess	2008-07-21 20:28:09.000	3	3	3	9

Items found: 2 Items selected: 1 << Page 1 of 1 >> Items per page: 20 ▼

<<Criteria

Figure 2-100 Selecting a process and requesting an Instances Snapshot™

The next window is a wide display with a number of tabs and a horizontal scroll bar. Only the left-hand part of the window is shown in Figure 2-101.

4. All the TestLoadProcesses are finished, so click the Finished (3) tab as shown in Figure 2-101. Click one of the Process Instance IDs.

**Process Instances**

This page displays information about process instances grouped by their most recent event. The list is sorted of an instance. [\[i\]](#)

Search Criteria: Time frame: 2007-09-20 13:47:00 - 2008-09-20 13:47:00; Sort: ascending; Skipped templ

► Refine your Search Criteria

Running (0)	Suspended (0)	<b>Finished (3)</b>	Failed (0)	Terminated (0)	Compensated (0)	Compensation Failed (0)	Comp												
<table border="1"> <thead> <tr> <th>Instance ID</th> <th>Template Name</th> <th>Valid from</th> </tr> </thead> <tbody> <tr> <td><b>PI:1003011c.7f4c235a.cbfbf3f6.2bb80198</b></td> <td>TestLoanProcess</td> <td>2008-07-21 20:28:09.000</td> </tr> <tr> <td>_PI:1003011c.7f78ad16.cbfbf3f6.2bb80219</td> <td>TestLoanProcess</td> <td>2008-07-21 20:28:09.000</td> </tr> <tr> <td>_PI:1003011c.7f7d7731.cbfbf3f6.2bb80264</td> <td>TestLoanProcess</td> <td>2008-07-21 20:28:09.000</td> </tr> </tbody> </table>								Instance ID	Template Name	Valid from	<b>PI:1003011c.7f4c235a.cbfbf3f6.2bb80198</b>	TestLoanProcess	2008-07-21 20:28:09.000	_PI:1003011c.7f78ad16.cbfbf3f6.2bb80219	TestLoanProcess	2008-07-21 20:28:09.000	_PI:1003011c.7f7d7731.cbfbf3f6.2bb80264	TestLoanProcess	2008-07-21 20:28:09.000
Instance ID	Template Name	Valid from																	
<b>PI:1003011c.7f4c235a.cbfbf3f6.2bb80198</b>	TestLoanProcess	2008-07-21 20:28:09.000																	
_PI:1003011c.7f78ad16.cbfbf3f6.2bb80219	TestLoanProcess	2008-07-21 20:28:09.000																	
_PI:1003011c.7f7d7731.cbfbf3f6.2bb80264	TestLoanProcess	2008-07-21 20:28:09.000																	

Items found: 3 << Page 1 of 1 >> Items per page: 20 ▼

<<Process Templates Export

Figure 2-101 Select a process from the list of finished processes

The details of the process are displayed as shown in Figure 2-102.

Process Instance Detail

This page displays information about a single process instance.

Number of Activities found: 3

Instance Name or ID: PI:1003011c.7f4c235a.cbfbf3f6.2bb80198

Process Template Name: TestLoanProcess

Valid From: 2008-07-21 20:28:09.000

State: Finished

Started: 2008-09-20 12:26:48.295

Completed: 2008-09-20 12:27:01.703

Work Time in seconds: 12

Elapsed Time in seconds: 13

Activity Name or ID	Timestamp	state	events
ReplyLoanReq	2008-09-20 12:27:01.659	Finished	2
InvokeOrignianlLoanProcess	2008-09-20 12:27:01.576	Finished	2
ReceiveLoanReq	2008-09-20 12:26:48.808	Finished	2

Items found: 3

<<<

Page 1 of 1

>>>

Items per page: 20

<<Instances Snapshot

Figure 2-102 The process details display

If you can successfully list process instances and display their details then you have successfully verified the Lists function of BPC Observer.

## Verify the BPC Observer chart function

Perform the following steps to verify the BPC Observer chart function.

1. In the Charts menu, click **Process snapshot** as shown in Figure 2-103.

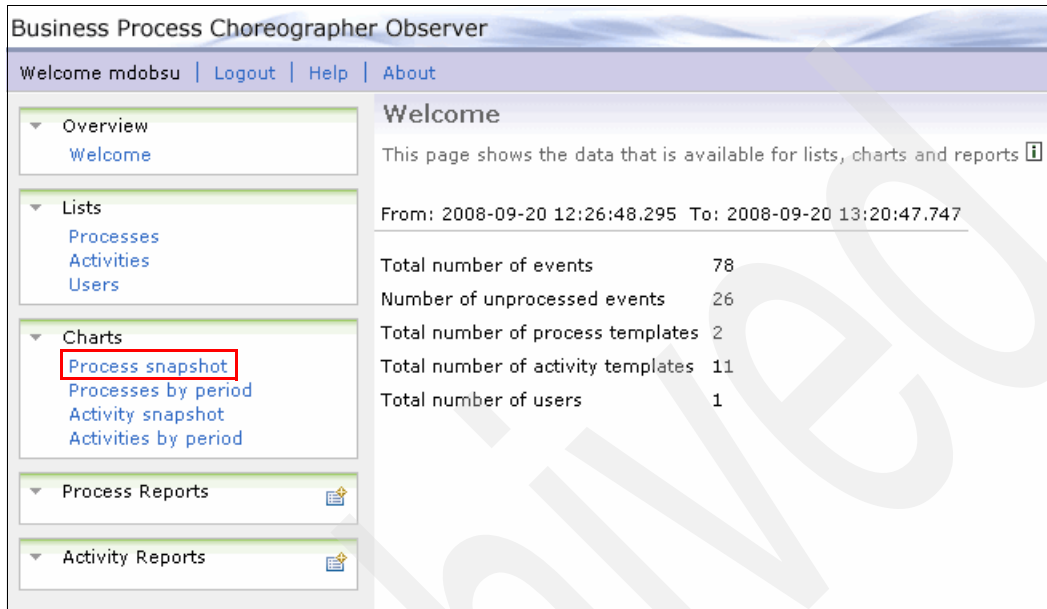
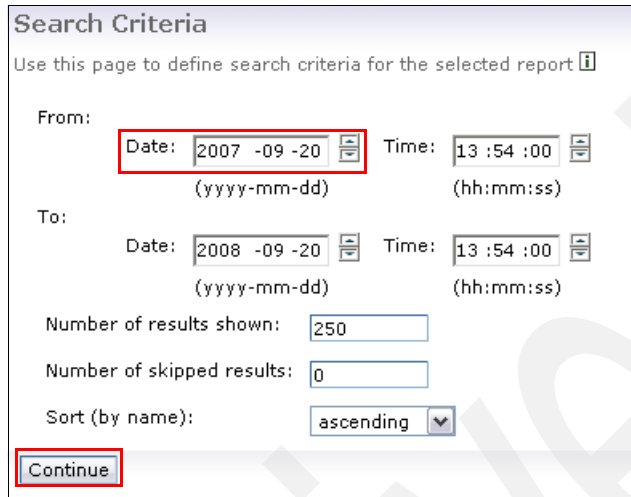



Figure 2-103 Click Process snapshot under Charts

2. Set the search criteria by clicking the down arrow to set the From date to 2007. Click **Continue** as shown in Figure 2-104.



**Search Criteria**

Use this page to define search criteria for the selected report 

From: Date:  Time:   
(yyyy-mm-dd) (hh:mm:ss)

To: Date:  Time:   
(yyyy-mm-dd) (hh:mm:ss)

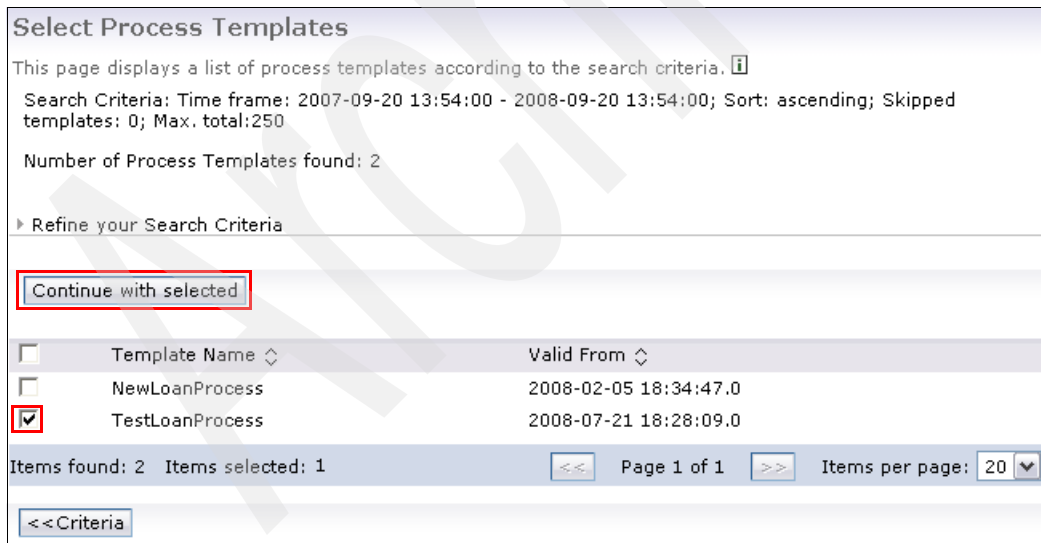
Number of results shown:

Number of skipped results:


Sort (by name):

Figure 2-104 Change the From date for a Search criteria date range

3. Select the check box next to one of the processes and click **Continue with Selected** as shown in Figure 2-105.



**Select Process Templates**

This page displays a list of process templates according to the search criteria. 

Search Criteria: Time frame: 2007-09-20 13:54:00 - 2008-09-20 13:54:00; Sort: ascending; Skipped templates: 0; Max. total:250

Number of Process Templates found: 2

► Refine your Search Criteria

<input type="checkbox"/>	Template Name	Valid From
<input type="checkbox"/>	NewLoanProcess	2008-02-05 18:34:47.0
<input checked="" type="checkbox"/>	TestLoanProcess	2008-07-21 18:28:09.0

Items found: 2 Items selected: 1  Page 1 of 1  Items per page:

Figure 2-105 Select a process and click Continue with selected

A chart is displayed showing the number of processes in each state. For our TestLoanProcess, all processes were Finished as shown in Figure 2-106.

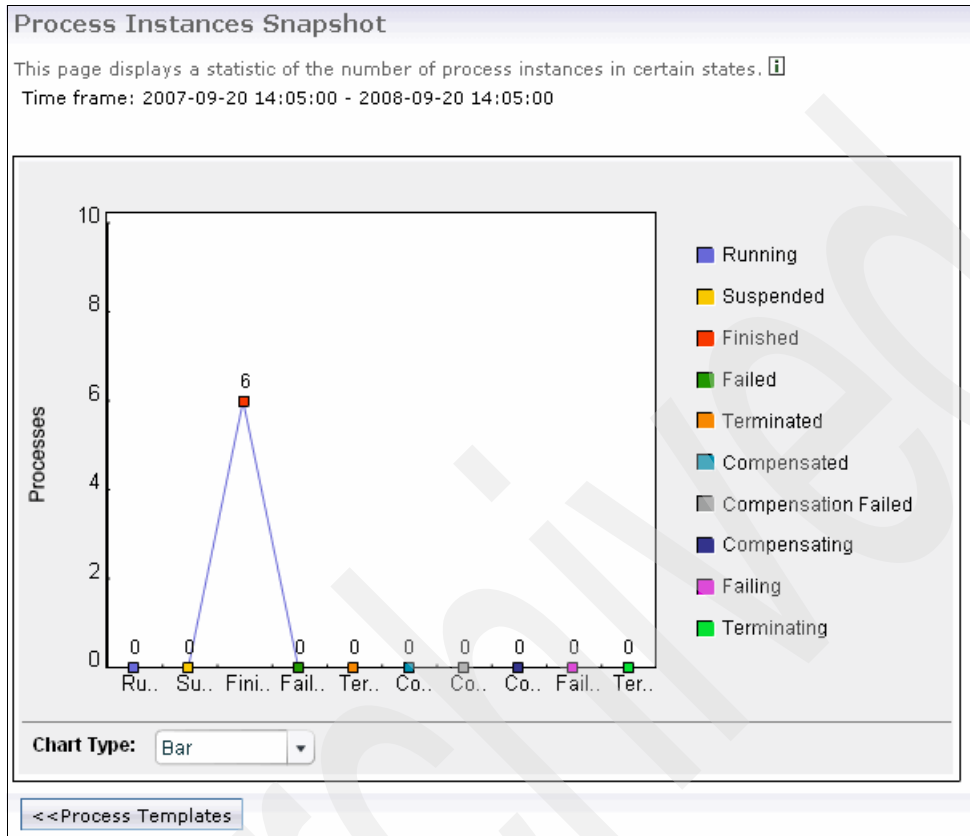


Figure 2-106 Chart showing Finished processes and verifying the Chart function

If the chart is displayed you have successfully verified the Chart function of BPC Observer.

**Note:** We found that the chart did not display immediately using Internet Explorer® V6.0.2900. This may have been due to a security setting preventing a pop-up. The chart displayed immediately using Firefox.



## Verify the Report function

Perform the following steps to verify the report function.

1. Click the icon to the right of Activity Reports as shown in Figure 2-107.

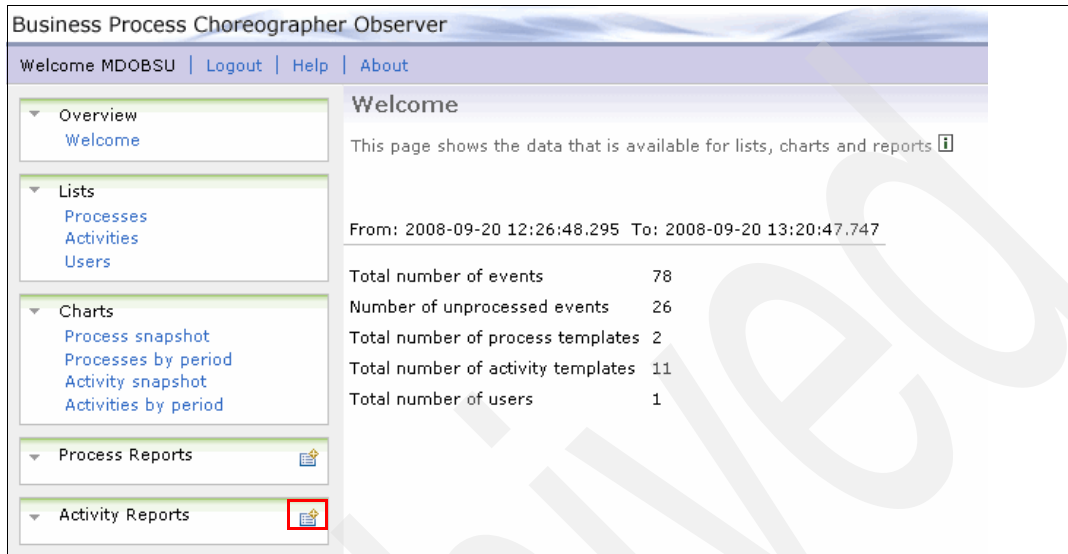


Figure 2-107 Creating a new Activity Report

2. Select the Snapshot Report radio button (Figure 2-108) and click **Next**.

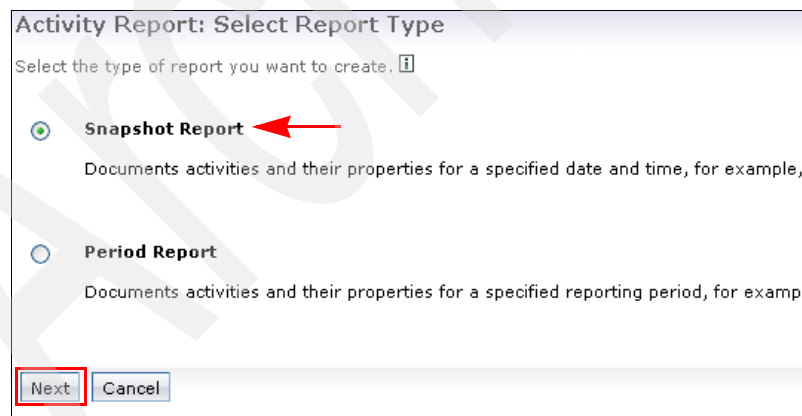


Figure 2-108 Choosing a Snapshot Report

You are presented with a panel that offers three different ways to take a snapshot report.

3. Select the Take a snapshot now radio button (Figure 2-109), and click **Next**.

Activity Snapshot Report: Select Snapshot Type

Specify when you want to take the snapshot.

☒ Take a snapshot now

☐ Take a snapshot at a specific date and time

☐ Take repeated snapshots according to a reporting cycle

Back Next Cancel

Figure 2-109 Choose to take a Snapshot now

4. Specify the content in your report.

A report is created by selecting attribute names that will become columns in the report, together with rules for selecting the data.

The report-creation tool provides list boxes from which you select attribute names and the layout your report. After you have chosen all the attribute names, specify content filters that limit the data presented in the report.

Notice the Threshold field in Figure 2-110. This is where you can set a limit on the number of results shown in the report.

Click **Add** to add the first Attribute Name.

Activity Snapshot Report: Specify Report Content

Select the information to be included in the report, for example process template name,

Attribute Name
<div>Add</div>

Number of results shown

Threshold:


☐ Use the threshold as a parameter





Back Next Cancel

Figure 2-110 Adding an attribute name

As the text at the top of the panel in Figure 2-111 says, this attribute will become one column in the report.

**Activity Snapshot Report: Specify Report Content**

Select the information to be included in the report, for example process template name, or event time. This information appears as columns in your report. 

	Attribute Name	Type	Sort	Action
1	Activity completed	Timestamp	No Sort	   

**Columns**

**Add**

**Number**

**Threshold**

☐ Use

**Back**

- Activity completed
- Activity event count
- Activity kind
- Activity name
- Activity started
- Activity instance ID
- Activity template ID
- Activity event
- Activity state
- Event time
- Process activity count
- Process activity event count
- Process completed
- Process event count
- Process deletion time
- Process started
- Process template name
- Process work time
- Process instance ID

Figure 2-111 Selecting Activity name as the first attribute (column) in the report

5. Choose an attribute, like Activity name, from the Attribute Name list box. Click **Add**.

6. Choose another attribute, like Activity state. Notice how this is the second attribute in the report, as shown in Figure 2-112. Click **Add**.

**Activity Snapshot Report: Specify Report Content**

Select the information to be included in the report, for example process template name, or event time. This information appears as columns in your report.

	Attribute Name	Type	Sort	Action
1	Activity name	String	No Sort	
2	Activity state	String	No Sort	

**Add**

**Number of results shown**

Threshold:

☐ Use the threshold as a parameter

**Back** **Next** **Cancel**

Figure 2-112 Selecting Activity state as the second attribute (column) in the report

7. Choose another attribute, like Activity instance ID. Click **Add**. The panel will now show three attributes (Figure 2-113). Notice that on the right of the window under the Action column, you have options to edit or delete the attribute, or to move the attribute up or down in the list.

There is also a list box under the Sort column for the Activity instance ID, which allows you specify the sort order for that attribute. The actions available in these fields depend on the nature of the attribute in the Attribute name column. Click **Next**.

**Activity Snapshot Report: Specify Report Content**

Select the information to be included in the report, for example process template name, or event time. This information appears as columns in your report.

	Attribute Name	Type	Sort	Action
1	Activity completed	Timestamp	No Sort	
2	Activity state	String	No Sort	
3	Activity instance ID	String	No Sort	

**Add**

**Number of results shown**

Threshold:

☐ Use the threshold as a parameter

**Back** **Next** **Cancel**

Figure 2-113 Add Activity instance ID as a third attribute and then click Next

8. On the Activity Snapshot Report:Specify Filter Content panel, specify one or more rules that will filter the content that is displayed in the report.

The procedure in step 9 will guide you through the steps to specify the content rule. Click **Add** to add the first filter condition, as shown in Figure 2-114.

**Activity Snapshot Report: Specify Filter Content**

Specify conditions to filter the content of the report. These conditions refine the content to help make your report more specific, for example you can restrict the event time to a specific date and time.

Attribute Name	Condition	Value	Parameter	Action
<input type="button" value="Add"/>				

Figure 2-114 Adding rules to filter content

9. Perform the following steps to specify the first filter content rule.
  - a. In the Attribute Name list box, select **Activity State**.
  - b. Allow the Condition to default to **Equal**.
  - c. Choose a state by clicking the lightbulb icon to the right of Value field.  
We chose **Finished** (Figure 2-115) because all our TestLoadProcesses were in Activity state Finished.
  - d. Click **Next**.

**Activity Snapshot Report: Specify Filter Content**

Specify conditions to filter the content of the report. These conditions refine the content to help make your report more specific, for example you can restrict the event time to a specific date and time.

Attribute Name	Condition	Value	Parameter	Action
Activity state	equal		<input type="checkbox"/>	

- Ready
- Running
- Waiting
- Claimed
- Finished
- Expired
- Failed
- Skipped
- Stopped
- Terminated

Figure 2-115 Specifying the filter content

10. Review the report structure (Figure 2-116) and click **Run**.

**Activity Snapshot Report: Report Summary**

This page summarizes the definition of the report. ⓘ

**Settings**

You want to report a snapshot on activity instances.  
The report contains the current information for the qualifying instances.

**The report contains the following attributes:**

	Attribute Name	Type	Sort
1	Activity completed	Timestamp	No Sort
2	Activity state	String	No Sort
3	Activity instance ID	String	No Sort

The result is limited to 20 entries.

**The following filter settings apply to the report:**

Attribute Name	Condition	Value	Parameter
Activity state	equal	Finished	<input type="checkbox"/>

Figure 2-116 Reviewing and then running the report

11. Figure 2-117 shows that the report contains the columns that were requested, and there are rows of data that match the filter Activity State=Finished.

Report Result (unsaved)

This page shows the result of a user defined report. ⓘ

Activity completed ▾	Activity state ▾	Activity instance ID ▾
2008-09-20 12:26:48.808	Finished	_AI:1004011c.7f4c235d.cbfbf3f6.2bb8019f
2008-09-20 12:26:53.266	Finished	_AI:1004011c.7f4c3641.cbfbf3f6.2bb801af
2008-09-20 12:26:53.378	Finished	_AI:1004011c.7f4c36f5.cbfbf3f6.2bb801b5
2008-09-20 12:26:54.178	Finished	_AI:1004011c.7f4c38df.cbfbf3f6.2bb801b9
2008-09-20 12:26:54.595	Finished	_AI:1004011c.7f4c3a86.cbfbf3f6.2bb801bd
2008-09-20 12:26:54.928	Finished	_AI:1004011c.7f4c3c22.cbfbf3f6.2bb801c1
2008-09-20 12:26:55.626	Finished	_AI:1004011c.7f4c3d72.cbfbf3f6.2bb801c5
2008-09-20 12:27:00.608	Finished	_AI:1004011c.7f4c4022.cbfbf3f6.2bb801c9
2008-09-20 12:27:00.694	Finished	_AI:1004011c.7f4c53a5.cbfbf3f6.2bb801cd
2008-09-20 12:27:01.576	Finished	_AI:1004011c.7f4c258f.cbfbf3f6.2bb801a5
2008-09-20 12:27:01.659	Finished	_AI:1004011c.7f4c576d.cbfbf3f6.2bb801d5
2008-09-20 13:15:27.269	Finished	_AI:1004011c.7f78ad19.cbfbf3f6.2bb80220
2008-09-20 13:15:30.091	Finished	_AI:1004011c.7f78b824.cbfbf3f6.2bb80230
2008-09-20 13:15:30.169	Finished	_AI:1004011c.7f78b8c4.cbfbf3f6.2bb80236
2008-09-20 13:15:30.771	Finished	_AI:1004011c.7f78ba4b.cbfbf3f6.2bb8023a
2008-09-20 13:15:31.134	Finished	_AI:1004011c.7f78bb67.cbfbf3f6.2bb8023e
2008-09-20 13:15:31.990	Finished	_AI:1004011c.7f78be73.cbfbf3f6.2bb80242
2008-09-20 13:15:32.906	Finished	_AI:1004011c.7f78c074.cbfbf3f6.2bb80246
2008-09-20 13:15:35.177	Finished	_AI:1004011c.7f78c3ce.cbfbf3f6.2bb8024a
2008-09-20 13:15:35.267	Finished	_AI:1004011c.7f78ccb1.cbfbf3f6.2bb8024e

Items found: 20      Page 1 of 1      Items per page: 20 ▾

Figure 2-117 The unsaved Report results

12. Save the report for use again later, or export it as file of comma-separated-values (csv) so it can be imported into a spreadsheet. Perform the following steps to export the report as a csv file.

- Click **Save**.
- A small window prompts you to enter the name of the report (Figure 2-118).

Report Result (unsaved)

This page shows the result of a user defined report. ⓘ

Enter the name of the report

Report name:

Activity completed      Activity state      Activity instance ID

2008-09-20 12:26:48.808      Finished      \_AI:1004011c.7f4c235d.cbfbf3f6.2bb8019f

Figure 2-118 Set the saved report name

- c. Enter a name for the report and click **Save**.
- d. The saved report will then appear under Activity Reports as shown in Figure 2-119.

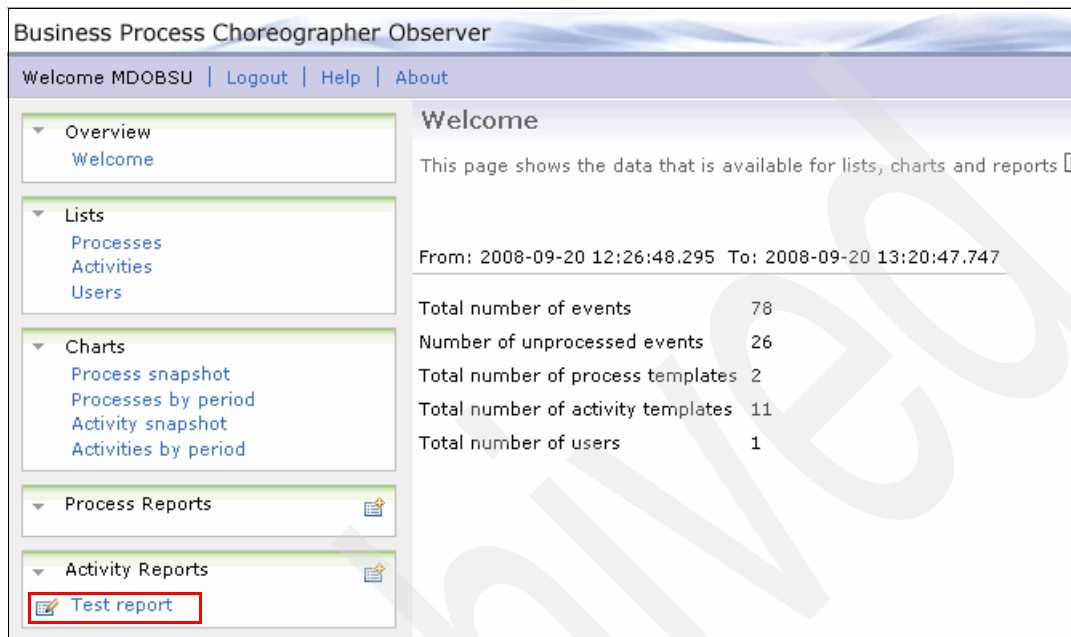


Figure 2-119 Saved Test report listed under Activity Reports



- e. Click **Export** and you are prompted to open the file or save it to disk on your work station (Figure 2-120).

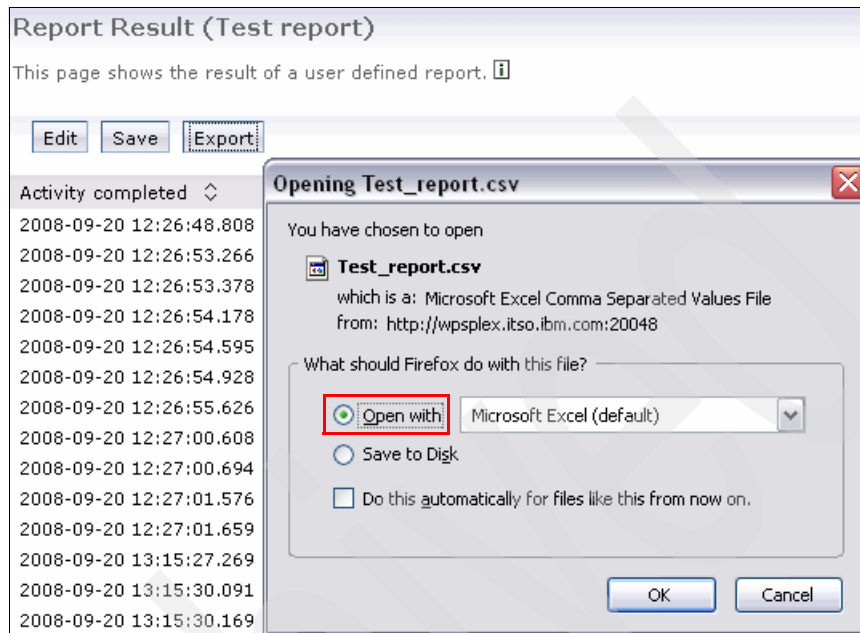


Figure 2-120 Choose whether to open file in Excel® or save it to disk

- f. Click **Open with** and then click **OK**. Excel opens with the values imported into the spreadsheet, as shown in Figure 2-121.

Test_report.csv						
	A	B	C	D	E	F
1	Activity completed	Activity st	Activity instance ID			
2	2008-09-20 10:26:48.808	Finished	_Al:1004011c.7f4c235d.cbfb3f6.2bb8019f			
3	2008-09-20 10:26:53.266	Finished	_Al:1004011c.7f4c3641.cbfb3f6.2bb801af			
4	2008-09-20 10:26:53.378	Finished	_Al:1004011c.7f4c36f5.cbfb3f6.2bb801b5			
5	2008-09-20 10:26:54.178	Finished	_Al:1004011c.7f4c38df.cbfb3f6.2bb801b9			
6	2008-09-20 10:26:54.595	Finished	_Al:1004011c.7f4c3a86.cbfb3f6.2bb801bd			
7	2008-09-20 10:26:54.928	Finished	_Al:1004011c.7f4c3c22.cbfb3f6.2bb801c1			
8	2008-09-20 10:26:55.626	Finished	_Al:1004011c.7f4c3d72.cbfb3f6.2bb801c5			
9	2008-09-20 10:27:00.608	Finished	_Al:1004011c.7f4c4022.cbfb3f6.2bb801c9			
10	2008-09-20 10:27:00.694	Finished	_Al:1004011c.7f4c53a5.cbfb3f6.2bb801cd			
11	2008-09-20 10:27:01.576	Finished	_Al:1004011c.7f4c258f.cbfb3f6.2bb801a5			

Figure 2-121 Exported report imported into Excel

If all this works as described you have successfully verified the Report function of BPC Observer.

## 2.19.7 Back up the WebSphere configuration HFS datasets

Back up the deployment manager's configuration ZFS data set and both of the node's configuration ZFS data set.

We used the JCL shown in Example 2-29 on page 104 but changed the //DASD DD card to //DASD DD DSN=WASCFG.MDCELL.DUMP.POSTIVT.

## 2.20 Configuring Business Space powered by WebSphere

Business Space powered by WebSphere is a new component of WebSphere Process Server for z/OS V6.1.2. It provides a common interface for application users to create, manage and integrate Web interfaces across the IBM WebSphere Business Process Management portfolio.

If you are familiar with the way WebSphere Portal Server operates, you will find Business Space easy to use. It uses much of the same underlying technology as WebSphere Portal Server to allow you to build customized web pages dynamically.

Business Space is part of the Support infrastructure, so in the two-clusters topology it is configured in the Support and Messaging cluster.

Perform the following steps to configure Business Space powered by WebSphere:

1. Create a data source for Business Space.  
See page 201.
2. Enable Widgets in Business Space.  
See page 202.
3. Run the Business Space Configuration wizard.  
See page 203.
4. Prepare the Business Space DDL.  
See page 205.
5. Execute the Business Space DDL.  
See page 205.
6. Define EJBROLE profile for role Administrator.  
See page 206.

7. Start the applications.  
See page 207.
8. Enable Business Rules in the Business Space.  
See page 207.
9. Configure local Business Space help.  
See page 209.
10. Verify Business Space.  
See page 209.

### 2.20.1 Create a data source for Business Space

The reason to create a data source for Business Space before running the Business Space configuration wizard is that the wizard will look for an existing data source with a JNDI name of `jdbc/bpm/BusinessSpace` and will use this if it is available. If it does not find an existing data source it will create one by modelling it on the existing data source called `WBI_Datasource`.

Although this process creates the Business Space data source successfully, we found that the DDL for Business Space could not be created because of a missing `'` in the path to the directory where the DDL was being written. When an existing data source was used this error did not occur.

Perform the following steps to create a data source for Business Space.

1. Create the data source at the cluster level by performing the following steps.
  - a. In Step 1: Enter basic data source information, enter the following attributes:
    - Name: Business Space Data Source
    - JNDI Name: `jdbc/bpm/BusinessSpace`
    - Component Managed authentication alias: Choose the alias used by the BPC data source. For our MD cell that was `BPCDB_mdcl01_WPS_Auth_Alias`.
  - b. In Step 2: Enter database specific properties for the data source, enter the following attributes:
    - Database name: `DB9F` (the DB2 for z/OS Location name)
    - Driver type: `2`
    - Server name: `wtsc42.itso.ibm.com` (hostname of DB2 DDF listener)
    - Port number: `37893` (port number of DB2 DDF listener)
  - c. In Step 3: Summary, click **Finish**.

2. Save the configuration changes.
3. Click **Business Space Data Source**. Click **Custom properties**. Set property `fullyMaterializeLobData=false`.
4. Save the configuration changes.

## 2.20.2 Enable Widgets in Business Space

All widgets are installed with Business Space powered by WebSphere but they must be enabled before they can be used in Business Space. The WebSphere Process Server for z/OS Information Center says to perform this step after configuring Business Space, but in our experience it is necessary to have the customized endpoint xml files in place before you run the Business Space configuration wizard.

When WebSphere Process Server for z/OS is configured into a stand-alone server, the widgets are enabled when the server is configured by `zWPSConfig.sh`. However, when WebSphere Process Server for z/OS is a network deployment configuration (managed node or a cluster) you must customize endpoint registry files that point to the virtual URLs for your cluster.

The original endpoint files provided in the product `registryData` directory contain tags with relative paths that work well when all files are on a local host. They must be updated with the full url when not.

Perform the following steps to enable widgets in Business Space.

1. Create both of the directories `/BusinessSpace/registryData` under each node's `<profile_root>` and then copy the endpoint registration files `wpsEndpoints.xml` and `bpcEndpoints.xml` from the product's `<install_root>/BusinessSpace/wps/registryData` directory to `<was_home>/profiles/default/BusinessSpace/registryData` on each node.

We created a shell called `copyendpoints.sh` that makes the directories `/BusinessSpace/registryData` under `<was_home>/profiles/default` and then copies the files. The shell also sets ownership of everything below the `BusinessSpace` directory to the cell's file system owner and configuration group by issuing the following `chown` command:

```
chown -Rh MOWNER:MDCFG BusinessSpace
```

2. In `bpcEndpoints.xml`, edit the tags `<tns:url>` in each file to set a fully-qualified url and port for the cluster where Business Process Container is configured.

In the two-clusters topology this is the `AppTarget` cluster.

The port you must specify is the `wc_defaulthost` port for the cluster.

For example, when configuring Business Space in our two-clusters topology MP cell we added the hostname:port of the cluster to the <tns:url> tags as high-lighted in bold in Example 2-42.

*Example 2-42 Customizing the bpcEndpoints.xml file*

---

```
<tns:Endpoint>
  <tns:id>{com.ibm.bspace.htm}bpaceTeamTaskListServiceRootId</tns:id>
  <tns:version>1.0.0.0</tns:version>
  <tns:url>http://wpsplex.itso.ibm.com:22048/rest/bpm/htm</tns:url>
  <tns:description>Location of services for HTM widgets</tns:description>
</tns:Endpoint>
```

---

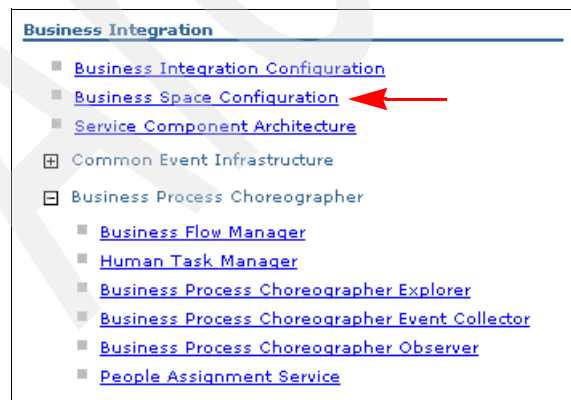
3. In wpsEndpoints.xml, edit the tags<tns:url> in each file to set a fully-qualified url and port of the cluster running the core WebSphere Process Server for z/OS functions.

If you have configured an HTTP Server to balance work across the cluster, specify the distributed VIPA address and port of the HTTP Server in both of the endpoint files.

### 2.20.3 Run the Business Space Configuration wizard

Perform the following steps to run the Business Space Configuration wizard.

1. In the navigation pane click **Servers** → **Clusters** and click the cluster name. Business Space is part of the support infrastructure and so is configured in the Support and Messaging cluster when you are configuring it in the two-cluster topology.
2. On the Configuration page, under Business Integration, click **Business Space Configuration**, as shown in Figure 2-122.



*Figure 2-122 Launch the Business Space Configuration wizard*

3. On the Business Space Configuration panel (Figure 2-123), if Business Space has already been configured, you can view the panel but cannot edit the fields. When configuring Business Space for the first time, complete the panel as follows.

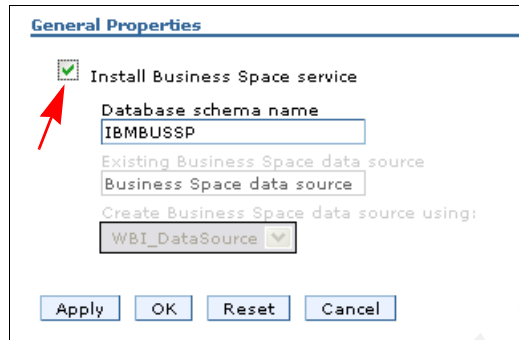


Figure 2-123 Configure Business Space schema and data source

- a. Select the Install Business Space service check box.
- b. In the Database schema name text box, enter the schema name you want to use for Business Space in DB2 for z/OS. We entered the default schema name of IBMBUSSP because in WebSphere Process Server for z/OS V6.1.2.0, Business Space did not use the schema name we entered. This problem is resolved in WebSphere Process Server for z/OS V6.1.2.1.
- c. If you already created the data source Business Space data source with a JNDI name of jdbc/bpm/BusinessSpace, that data source will already be set in the Existing Business Space data source field.

If you did not pre-define the Business Space data source, select a datasource from the list box in the Create Business Space data source using: field. For our cell the only datasource listed was WBI\_DataSource, which is the one used by the WebSphere Process Server common components.

**Note:** Selecting the WBI\_DataSource does not mean that Business Space will use that datasource. Instead a new data source is created called Business Space DataSource with a JNDI name of jdbc/bpm/BusinessSpace and with the attributes of WBI\_DataSource.

4. Save the configuration changes.

## 2.20.4 Prepare the Business Space DDL

The DDL for the Business Space tables is located in the deployment manager's configuration at

`<dmgr_root>/profiles/default/dbscripts/BusinessSpace/<dbtype>/<db_name>.`

For example, for our MD cell the Business Space DDL was in  
`/wasv61config/mdcell/mddmnode/DeploymentManager/profiles/default/dbscripts/BusinessSpace/DB2zOSV9/DB9F.`

The DDL in the `createTable_BusinessSpace-sql` file is in ASCII. It is only partly customized. The DDL must be converted to EBCDIC and edited before you can run it using DSNTEP2. We created sample job MDCPYBSP to convert the Business Space DDL to EBCDIC and copy it to a z/OS sequential file for editing.

The schema specified on the Business Space Configuration wizard is used to fully-qualify all the DDL. However, the following changes are required to the DDL:

- ▶ Split lines that continue beyond column 80.
- ▶ Change `@STOGRP@` to our DB2 Storage group name, `MDWPSSG`.
- ▶ Change `@DBNAME@` to our DB2 Database name, `MDBPCDB`.
- ▶ Change Bufferpools to those we wanted to use. Do not use `BP0`, `BP8K0`, or `BP16K0`, because these are used by the DB2 catalog and directory.

After editing the DDL, we copied it to a partitioned dataset with LRECL 80 called `WASCFG.MDCELL.WPS.SQL`.

## 2.20.5 Execute the Business Space DDL

We created a database for Business Space called `MDBSPDB` in Section 2.14.7, "Grant table privileges on CEI tables" on page 143 when we created all the databases with the job `MDCREDB`.

We created job `MDDDLBSP` to execute the Business Space DDL in `WASCFG.MDCELL.WPS.SQL(MDBSPTB)`, which contained our chosen schema of `MDWPS`. When we discovered that Business Space used a hard-coded schema of `IBMBUSSP`, we created new DDL with a schema of `IBMBUSSP` in member `IBMBUSSP` of dataset `WPS.SQL`. We created job `IBMBUSSP` in the `WPS.CNTL` dataset to execute that DDL.

## 2.20.6 Define EJBROLE profile for role Administrator

When using RACF to perform EJBROLE authorization checking, define the role Administrator to RACF. Note that the role name starts with an upper case “A.” This means this is a different role to “administrator” (lowercase A), which is the Integrated Solutions Console administrator role.

For our MD cell we created the role MD.Administrator as part of the job MDEJBROL. An extract of that job showing the commands applicable to Business Space is shown in Example 2-43.

*Example 2-43 Commands to define EJBROLE Administrator for Business Space*

RDEFINE	EJBROLE	MD.Administrator	+
		UACC(NONE)	+
		OWNER(WASCFG)	+
		DATA('MD BSpace Admin role')	
PERMIT		MD.Administrator	+
		CLASS(EJBROLE)	+
		RESET	
PERMIT		MD.Administrator	+
		CLASS(EJBROLE)	+
		ID(MDBSPADG)	ACCESS(READ)

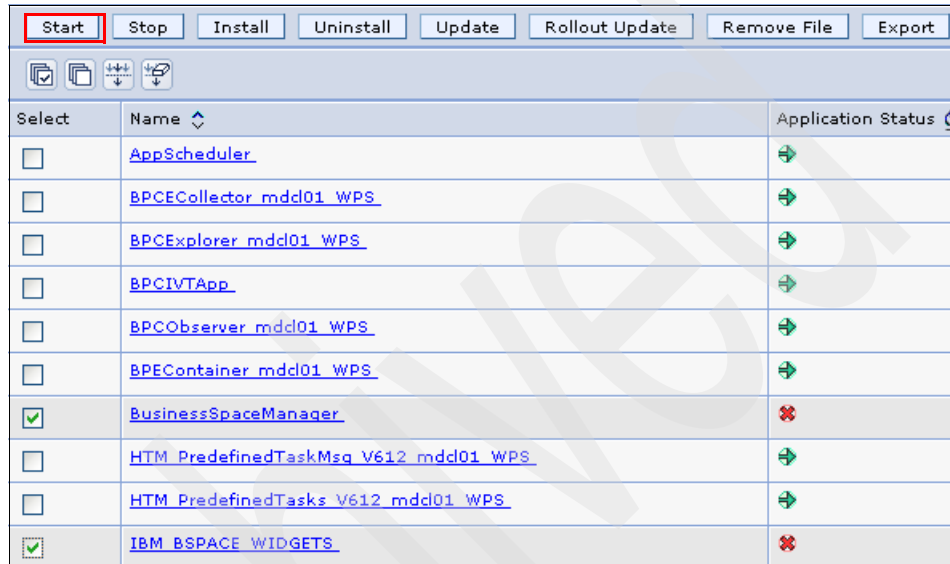
As shown in Example 2-43, we defined a specific group called MDBSPADG (BSPADG=Business Space Administrator Group) to which other user IDs can be connected in order to be given the role Administrator.

If you prefer to use GEJBROLE profiles, use job GEJBROLE, which assigned the Business Space role Administrator to the profile protecting administration roles. See Section 2.2.7, “Managing EJB roles using RACF GEJBROLE profiles” on page 41.



## 2.20.7 Start the applications

To start the applications navigate to **Applications** → **Enterprise applications**, check the BusinessSpaceManager and IBM BSPACE WIDGETS check boxes, and click **Start** (Figure 2-124). Remember to verify that the applications start and that there are no errors in the logs.



<div>Start Stop Install Uninstall Update Rollout Update Remove File Export</div>		
<div>☑ ☐ ⬆ ⬇ ⬆ ⬇</div>		
Select	Name	Application Status
<input type="checkbox"/>	<a href="#">AppScheduler</a>	➡
<input type="checkbox"/>	<a href="#">BPCECollector_mdd01_WPS</a>	➡
<input type="checkbox"/>	<a href="#">BPCEExplorer_mdd01_WPS</a>	➡
<input type="checkbox"/>	<a href="#">BPCIvTApp</a>	➡
<input type="checkbox"/>	<a href="#">BPCObserver_mdd01_WPS</a>	➡
<input type="checkbox"/>	<a href="#">BPCEContainer_mdd01_WPS</a>	➡
<input checked="" type="checkbox"/>	<a href="#">BusinessSpaceManager</a>	✖
<input type="checkbox"/>	<a href="#">HTM_PredefinedTaskMsg_V612_mdd01_WPS</a>	➡
<input type="checkbox"/>	<a href="#">HTM_PredefinedTasks_V612_mdd01_WPS</a>	➡
<input checked="" type="checkbox"/>	<a href="#">IBM BSPACE WIDGETS</a>	✖

Figure 2-124 Start the Business Space applications

## 2.20.8 Enable Business Rules in the Business Space

After installing Business Space powered by WebSphere in a clustered environment, you must run a script so that business rule widgets are available in Business Space at run time. This is configured automatically on a stand-alone server profile, but when you configure a cluster you need to run a jacl script.

The script that you use to enable business rules in Business Space is called `installBRRestAPI.jacl` and is located in the WebSphere Process Server for z/OS `<install_root>/bin` directory. We created sample job `MDBUSRUL` to run `installBRRestAPI.jacl`, as shown in Example 2-44 on page 208.

*Example 2-44 Sample JCL MDBUSRUL to run installBRRestAPI.jacl*

---

```
//MDBUSRUL JOB (0),CLASS=A,MSGCLASS=A,REGION=0M,
// USER=MDADMIN,PASSWORD=MDADMIN
//*****
/* Configure Business Rules in Business Space */
//*****
//INSTO EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
cd /wasv61config/mdcell/mddmnode/DeploymentManager/bin; +
./wsadmin.sh -user MDADMIN -password MDADMIN +
-f installBRRestAPI.jacl +
-clusterName mdc101_WPS +
1> /tmp/mdbussrul_1308.out +
2> /tmp/mdbussrul_1308.err;
//*****
/* STEPS TO COPY THE OUTPUT THE JOB LOG */
//*****
//CPOUT EXEC PGM=IKJEFT01,REGION=0M
//SYSEXEC DD DISP=SHR,DSN=BBO6142.MDDMNODE.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BBOHFSWR '/tmp/mdbussrul_1308.out'
BBOHFSWR '/tmp/mdbussrul_1308.err'
/*
```

---

An extract of the messages issued is shown in Example 2-45.

*Example 2-45 Messages issued when running installBRRestAPI.jacl*

---

```
WASX7303I: The following options are passed to the scripting environment and
are available as arguments that are stored in the argv variable:
"[-clusterName, mdc101_WPS]"
WASX7327I: Contents of was.policy file:
grant codeBase "file:${application}" {
    permission java.security.AllPermission;
};
ADMA5016I: Installation of BusinessRuleRestAPI_mdc101_WPS started.
ADMA5058I: Application and module versions are validated with versions of
deployment targets.
. .
ADMA5013I: Application BusinessRuleRestAPI_mdc101_WPS installed successfully.
>>>> The application is successfully installed.
>>>> Please wait for starting the application...
>>>> The application is successfully started.
```

---

## 2.20.9 Configure local Business Space help

By default, Business Space uses the Internet to access the help on the IBM Web site. If you want to change the configuration of the help so that business users access the help on your local intranet, you can change the help URL on the Business Space landing page. Perform the following steps to change the help URL on the Business Space landing page.

1. Obtain the URL and port of your local help server.

The default help port is 8888. Look in either the IC\_start.bat file or the IC\_start.sh file to see the port the help server is using.

2. Go to the BSpaceManager.ear file, where it is deployed.
3. Open the landingpage.html with an ASCII text editor. The file is, located at BSpaceManager.ear\_location/WebContent/bspace\_help/framework\_help/.
4. Change the help URL to an internal URL, like the following URL:

```
http://wpsplex.itso.ibm.com:8888/help/topic/com.ibm.btools.help.\
modeler.collab.publish.doc/doc/tasks/viewing/viewingdataitems.html
```

5. Save and close the landingpage.html file.

## 2.20.10 Verify Business Space

Perform a brief test of Business Space to verify the installation.

1. Launch the Business Space application from a browser using context root /BusinessSpace. For example, for our MD cell, the url was as follows:

```
http://wtsc42.itso.ibm.com:22048/BusinessSpace
```

2. Log in with a user ID that has been permitted to the Business Space role Administrator (with an upper case A).

Earlier we had permitted RACF group MDBSPADG to the EJBROLE profile called MD.Administrator and the group MDBSPADG contained the user ID MDBSPADM. Therefore we logged in with user ID MDBSPADM.

If RACF APPL class is active, ensure that the user ID you login with has access to the APPL profiles protecting the cell.

3. After logging in, you see the Business Space manager workspace shown in Figure 2-125.

It is largely empty at this point because the user ID has no Business Spaces. The Business Space Manager workspace is shown so you can create a new Business Space and add new pages to the space. Then you can add widgets to the pages, a bit like added portlets to pages in WebSphere Portal Server.

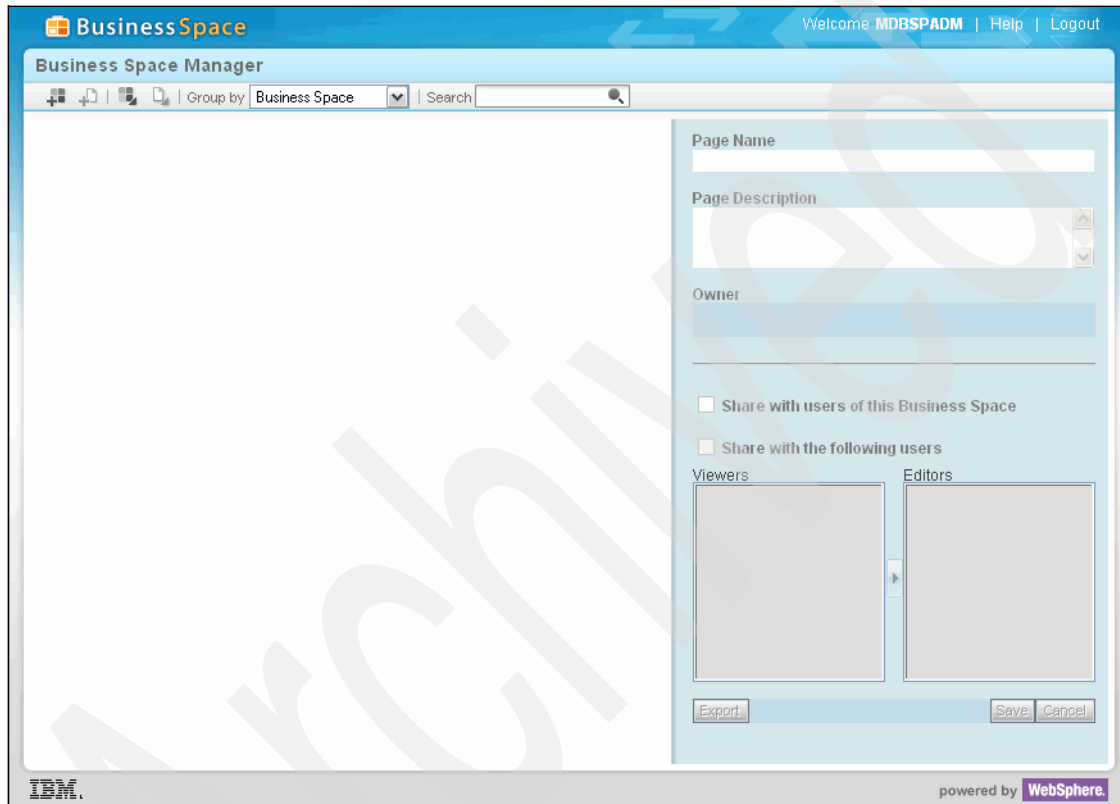


Figure 2-125 Business Space manager initial window

4. A user can have many business spaces with each one having a different purpose but Business Space can display the contents of one space at a time. That space is known as the open Business Space.

Create a new Business Space by clicking the icon on the top left in the tool bar as shown in Figure 2-126.



Figure 2-126 Click the icon to create a new Business Space

5. Enter a name for the Business Space (such as TestSpace1), choose the Business Monitoring template and click **OK**, as shown in Figure 2-127.

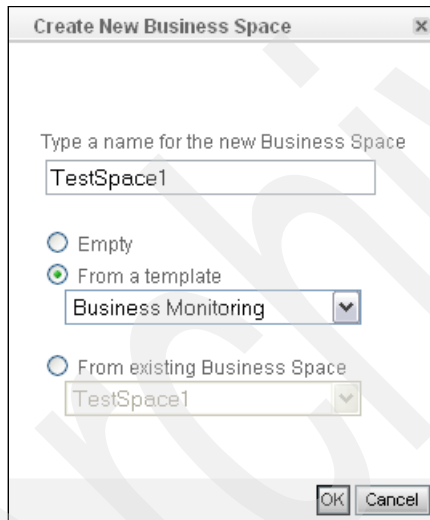


Figure 2-127 Create a new workspace

6. Click **TestSpace1** to select it. It turns pale orange when selected. Click the icon to create a new page as shown in Figure 2-128.

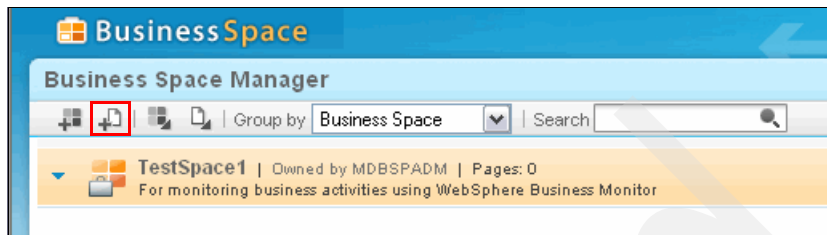


Figure 2-128 Select the TestSpace1 and click the icon to create a new page

7. Enter a name for the page (such as TestPage1) as shown in Figure 2-129 and click **OK**.

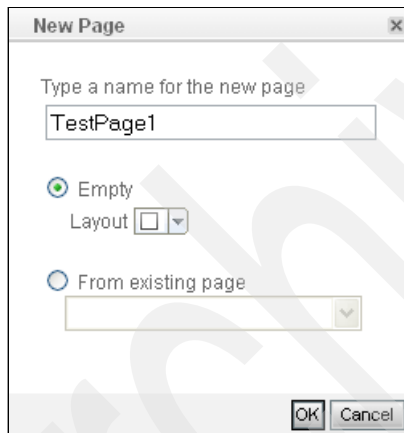


Figure 2-129 Enter a name for the new page and click OK

8. Click the link to TestPage1 in the orange page area to open it, as shown in Figure 2-130.

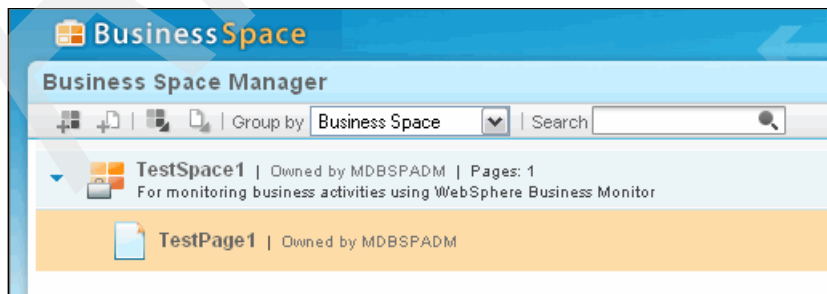


Figure 2-130 Click TestPage1 in the orange new page area to open

9. Click the **Add widget** icon to add a widget, as shown in Figure 2-131.

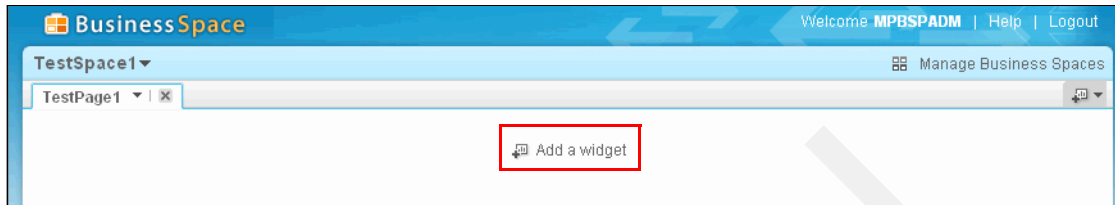


Figure 2-131 Add a widget

10. Select a spreadsheet widget from the list box shown in Figure 2-132, and click **OK**.

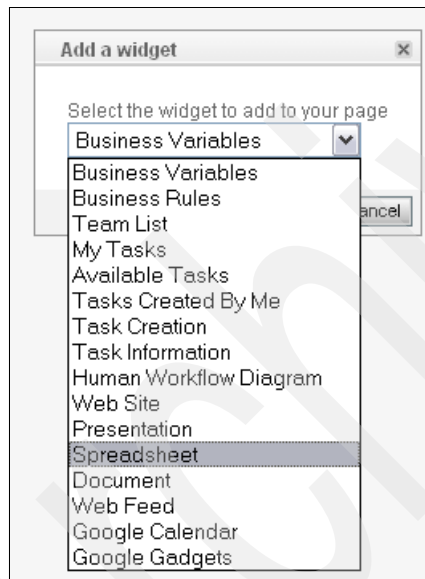


Figure 2-132 Select a spreadsheet widget

11. Click **Save** to download the sample spreadsheet or **Open** to open it in the browser.

We first tested that we could download and save the spreadsheet. If you decide to save the spreadsheet on your workstation, your browser will probably issue a warning about downloading a file, as shown in Figure 2-133.

Click **Continue** if you see this message.



Figure 2-133 Click continue to allow download of the spreadsheet

12. Save the file on your workstation.
13. Click **Open** when the download is complete. The spreadsheet will open in Excel. This has verified the file-download function.



14. Click the spreadsheet widget on the page and you are prompted again to **Open** or **Save** (Figure 2-134). Click **Open**.

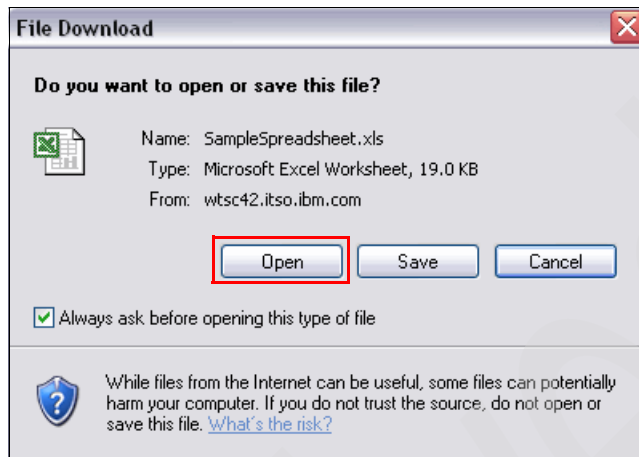


Figure 2-134 Click Open to open the file on the page

15. The spreadsheet will open in the area on the page reserved for the first widget, as shown in Figure 2-135.

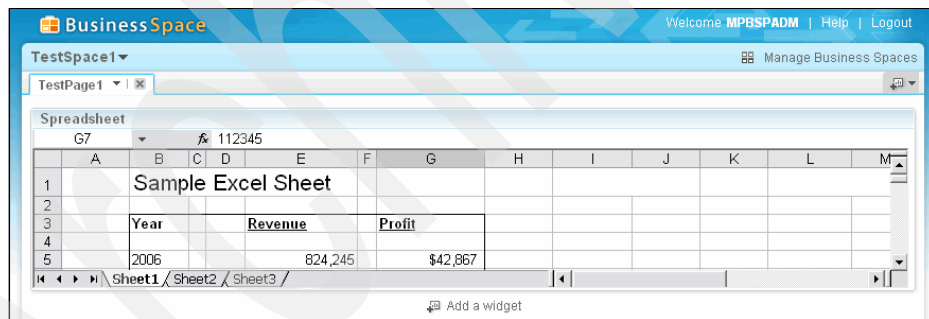


Figure 2-135 The spreadsheet opened in the page

16. Click the **Add a widget** icon that is below the spreadsheet in Figure 2-135.

17. Add the **My Tasks** widget. At this time you will have no tasks to display so the My Tasks pane is empty, as shown in Figure 2-136.

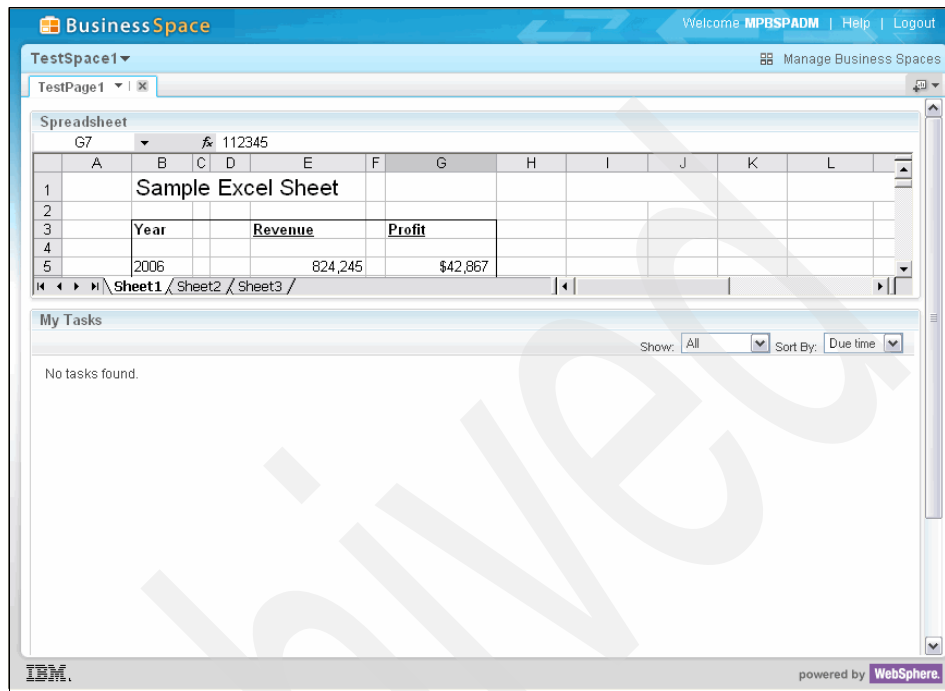


Figure 2-136 Business Space with My Tasks widget

You have now verified the basic functionality of Business Space.

## 2.21 People Directory Provider

You will need to configure a People Directory Provider if your applications use Human Task roles that will require People resolution at run time. Such applications contain People settings (formerly called Staff settings) that specify the user IDs or groups associated with roles related to Human Tasks, such as who can initiate a task and who can work on it.

The resolution of user IDs and groups is not performed by WebSphere Application Server for z/OS through the cell's user registry. Instead, the application references the JNDI name of a People Directory Provider configuration that contains the details of the user registry, which is responsible for this function. This allows different applications the ability to use different forms of People resolution.

When your application uses Human Tasks, it is necessary to ask the application developer for the details of how People resolution is performed.

Several different kinds of user registry are supported and some sample People Directory provider configurations are already provided with WebSphere Process Server for z/OS. One is a System People Directory configuration, which is meant to be used with a LocalOS user registry. If you have decided to use a LocalOS user registry like RACF with your WebSphere Process Server cell, you can choose to use that same RACF database for People resolution. This configuration has a JNDI name of `bpe/staff/systemconfiguration` so you can change the People settings in the application to refer to that JNDI name.

It is common to find that People resolution is performed against an LDAP user registry. Although a sample LDAP People Directory Provider configuration is provided, it is unlikely to work with your LDAP Server. You will need to create a new LDAP People Directory Provider configuration.

Whichever user registry is used for People resolution, it is necessary to ask the application developer for a list of all the roles that the application uses and the user IDs and groups that are mapped to those roles.

### 2.21.1 Customize an LDAP transformation xsl file

Transformation of Human Task people resolution variables (like people verbs or group names) into LDAP queries is done using an XSL Transform file.

Depending on the type of LDAP Server you are using you may have to update some properties in the XSL Transform file so they match the names of the object classes used for the user and group object classes in the LDAP server.

The file `<WPS_HOME>/ProcessChoreographer/Staff/LDAPTransformation.xsl` is a symbolic link to a sample `LDAPTransformation.xsl` file that is within the read-only product HFS. Even if you could update this file, do not do so. The file can be overwritten when fixes are installed. Instead, edit a copy of the file.

**Note:** The customized LDAP XSL Transform file must be available to the `<dmgr_home>/ProcessChoreographer/Staff` directory in the Deployment Manager's node. When BPEL applications that reference the LDAP Staff plug-in are deployed, the Deployment Manager will look for the customized LDAP Transformation file in its `<dmgr_home>/ProcessChoreographer/Staff` directory. You must complete the customization of the LDAP XSL Transform file and configure the LDAP People Directory Provider before you attempt to deploy any applications that depend on the LDAP People Directory Provider.

Perform the following steps to customize an LDAP transformation xsl file.

1. Copy the sample LDAP transformation xsl file to a file with a different name in the deployment manager's /ProcessChoreographer/Staff directory.
  - a. Open an OMVS or telnet session and issue the following commands:

```
> cd /wasv61config/mdcell/mddmnode/DeploymentManager
> cd ProcessChoreographer/Staff
```

**Note:** There are two cd commands above instead of one so you can easily copy and paste onto the OMVS command line. The whole path is too long to paste into the OMVS command line.

```
> cp LDAPTransformation.xsl MyLDAPTransformation.xsl
> chmod 775 MyLDAPTransformation.xsl
> chown MDADMIN:MDCFG MyLDAPTransformation.xsl
```

2. The MyLDAPTransformation.xsl file is in ASCII. Download it to your workstation for editing.
3. Notice the following lines near the beginning of the file:

```
<xsl:variable name="DefaultGroupClass">groupOfNames</xsl:variable>
<xsl:variable name="DefaultGroupClassMemberAttribute">member</xsl:variable>
<xsl:variable name="DefaultRecursivity">yes</xsl:variable>
```

Ensure the values for all the <xsl:variable> variables correspond to the object classes your LDAP server uses for users and groups. For Tivoli Directory Server it is necessary to modify only two values:

- a. Change the DefaultGroupClass variable from groupOfNames to groupOfUniqueNames.
- b. Change the DefaultGroupClassMemberAttribute variable from member to uniquemember.

**Global change?:** A global change on the string member is a bad idea because that string appears in many attributes in the file. It is best to do these changes individually.

When you have finished the relevant lines look like Example 2-46.

*Example 2-46 Modified values in MyLDAPTransformation.xsl file*

```
<xsl:variable name="DefaultGroupClass">groupOfUniqueNames</xsl:variable>
<xsl:variable name="DefaultGroupClassMemberAttribute">uniquemember</xsl:variable>
```

4. Upload the file to <dmgr\_home>/ProcessChoreographer/Staff.
5. Upload the file to <was\_home>/ProcessChoreographer/Staff for each node in your cluster.

## 2.21.2 Create an new LDAP People Directory Provider configuration

Our application did not need a People Directory Provider, but this section outlines the steps to create a new LDAP People Directory Provider configuration.

**Note:** When we tested WebSphere Business Services Fabric for z/OS in Section 3.6, “Install and test the sample application” on page 282, we had to configure an LDAP user registry because Fabric uses one to manage its application users, but Fabric does not require a People Directory Provider configuration.

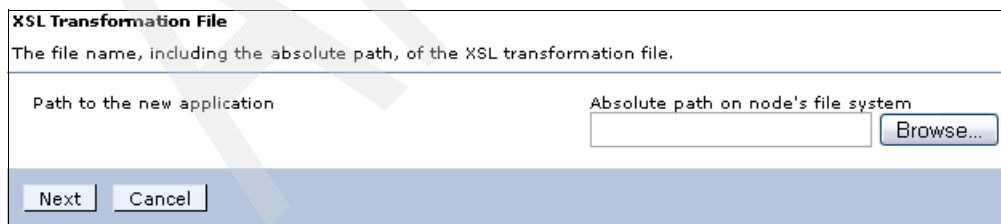
When you create a new LDAP People Directory Provider configuration the first step is to specify the location of the XSL Transform file. You can store your customized XSL Transform file anywhere, but when you create the new LDAP People Directory Provider configuration the wizard will copy your customized file to <dmgr\_home>/ProcessChoreographer/Staff and to <wps\_home>/ProcessChoreographer/Staff.

Perform the following steps to Create an new LDAP People Directory Provider configuration.

1. Log in to the Integrated Solutions Console and navigate to **Resources** → **People directory provider**.

Make sure you are at Node scope with the node set to the Process Server node, mdnodea.

2. Click **LDAP People Directory Provider** in the table.
3. Under Additional Properties, click **People directory configuration**.  
You will see the LDAP People Directory Configuration sample listed.
4. Click **New**.
5. Click **Browse** to navigate through the file system (Figure 2-137).



**XSL Transformation File**  
The file name, including the absolute path, of the XSL transformation file.

Path to the new application

Absolute path on node's file system

Figure 2-137 Navigating through the file system

6. Locate the modified MyLDAPTransformation.xml file and select the corresponding radio button, as shown in Figure 2-138.

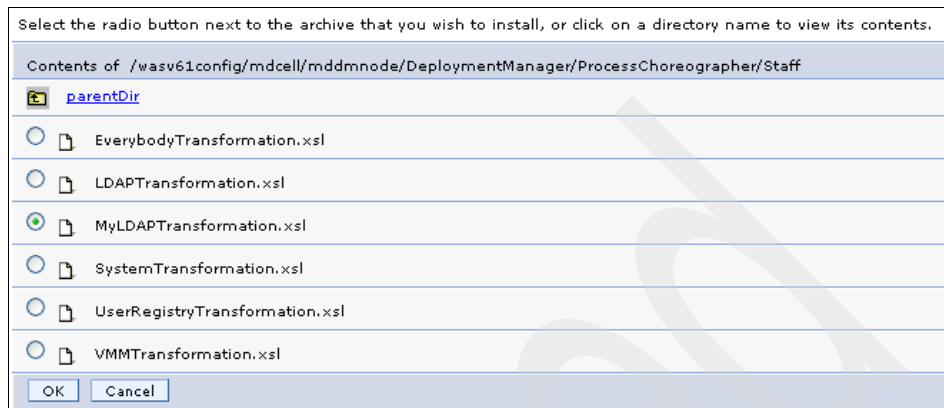


Figure 2-138 Select the customized LDAP xml transformation file

7. Click **OK**. You are returned to the XSL Transformation file window with the path to the customized transformation file completed. Click **Next**.
8. Enter the following variables in the appropriate fields, as shown in Figure 2-139.
  - Name: MyLDAPPeopleDirectory
  - Description: My LDAP People Directory Configuration
  - JNDI name: bpe/staff/MyLDAPPeopleDirectory

Figure 2-139 New LDAP People Provider configuration form

9. Click **OK**.

**Note:** The JNDI name for the LDAP People Provider must match what is specified in the application's Properties Details of the People Settings in WebSphere Integration Developer.

The new People Directory configuration is listed as shown in Figure 2-140. Do not save the changes yet.

<input type="checkbox"/>	<a href="#">LDAP People Directory Configuration sample</a>	This sample people directory configuration can be used for LDAP based people queries.	bpe/staff/sampleldapconfiguration
<input type="checkbox"/>	<a href="#">MyLDAPPeopleDirectory</a>	My LDAP People Directory configuration	bpe/staff/MyLDAPPeopleDirectory

Figure 2-140 New LDAP People Provider configuration listed

**Note:** The configuration shows the full path to the customized LDAP transformation file using the `${WBI_INSTALL_ROOT}` variable. The customized `MyLDAPTransformation.xsl` file that you placed in `/var/WebSphere/home/MDCFG/wpswork` is copied to the Process Server node. You can find a copy at `<wps_home>/ProcessChoreographer/Staff`.

10. Click the **MyLDAPPeopleDirectory** configuration in order to customize it.
11. Under Additional Properties, click the **Custom properties** link.
12. The BaseDN and ProviderURL properties are set to default values. Perform the following steps to update these values for your LDAP server.
  - a. Click **BaseDN**. Change the Value field to `o=ibm`.
  - b. Click **OK** to return to the list of custom properties.
  - c. Click **ProviderURL** and change the Value field to `ldap://wpsplex.itso.ibm.com:2389`.
  - d. Click **OK** to return to the list of properties.

**Note:** If you use copy/paste, be careful not to leave any blanks before or after the text you copy.

13. Click **Save** at the top of the window to save the changes and synchronize to the nodes.
14. Navigate to **Resources** → **People directory provider** and change the scope to the other node in the cluster.
15. Click **New**.
16. Click **Next**.
17. Repeat the steps above to define the new LDAP People Directory Provider configuration in the other node.
18. Stop and restart the cluster.

After restarting the cluster, if you look in the servant log you will see the new Staff Plugin configuration being bound to the NDI name.

```
WSVR0049I: Binding LDAP Staff Plugin Lab Configuration as  
bpe/staff/MyLDAPPeopleDirectory
```

### 2.21.3 Add users and groups to the LDAP server

Normally the application will have been tested in a non-z/OS environment using a test LDAP server, so part of the process of deploying the application into production for the first time is to add equivalent user IDs and groups to the production LDAP server.

The mapping of user IDs and groups to Human Task roles is specified in the application using fully-qualified DN's. Therefore, the application must be specifically configured for deployment into a new WebSphere Process Server for z/OS cluster that uses a different LDAP server from the one in the test environment.

In our case, the simple application we used to test WebSphere Business Services Fabric for z/OS did not use Human Task roles so we did not need to configure the People provider.

### 2.21.4 Deploy the application

When you deploy the application, the deployment manager will bind the JNDI name of the People Directory Provider in the application to the People Directory Provider configuration you created with that JNDI name. If you have not successfully created the LDAP People Provider configuration and restarted the cluster before installing the application, the application deployment may appear to complete normally, but when you try to run the application it will not be able to perform People resolution.



## 2.21.5 Test the application

When the application is deployed you can start it and test that it works using the BPC Explorer.

## 2.22 Failed events

This section provides guidance for failed events in WebSphere Process Server for z/OS.

### 2.22.1 What is an event?

An event is a Service Data Object (SDO) that is received by a WebSphere Process Server for z/OS application. An SDO is made up of data and a reference to the business operation executed by the application. When WebSphere Process Server for z/OS receives the event, the SDO is processed by the appropriate business application based on the referenced business operation.

Every system based on business processes will contain events. There will always be processes and events that fail. The expectation is that a well-developed application is developed by business knowledgeable people and the business knows best how to handle failed events and process. The application's exception and fault handling code is responsible for resolving business failures.

Most system level failures appear as a communication issue. There are two types of communication:

- ▶ Synchronous (is blocking)

The call is initiated and the thread waits for a response prior to processing further. In case of any failure, the invoking application is responsible for failure capture and retry logic. There is no administrative action available for a WebSphere Process Server for z/OS administrator.

- ▶ Asynchronous (is not blocking)

The call is initiated and the event is placed on a queue. The receiving process is listening on the queue to process the event and reply to the calling process. If there is business exception or fault in the receiving process, the application is responsible for failure capture and retry logic. There is no administrative action available for a WebSphere Process Server for z/OS administrator.

If two Service Component Architecture (SCA) components are communicating asynchronously and there is a failure (such as the system is not available), WebSphere Process Server for z/OS has built-in retry logic. Five retries is the

default. If the retry logic fails, then the event is considered failed and the WebSphere Process Server for z/OS Recovery Service (WPSRS) will move the event to the failed event queue. The WPSRS will persist the event into a database. The WebSphere Process Server for z/OS administrator can take administrative action using the Failed Event Manager.

**Important:** Because adapters are an asynchronous technology, configurations that make use of adapters see a greater occurrence of failed events.

### 2.22.2 How to use the Failed Event Manager

WebSphere Process Server for z/OS Integrated Solutions Console has a tool called Failed Event Manager. This is a Web-based tool that will enable the administrator to submit events to a component that failed to complete.

1. In order to use the Failed Event Manager your user ID must be permitted to the role WBIOperator. Because the Failed Event Manager is part of the Integrated Solutions Console, your user ID must also be allowed to log in to the Integrated Solutions Console.

In our system we decided to make the role WBIOperator UACC(NONE) and we permitted the WebSphere configuration group, the BPE administrator group and the Event administrator groups to the role. This was done in the sample job MDEJBROLE executed in Section 2.3.3, “Create RACF user IDs, groups, and EJBROLE profiles” on page 62. The commands related to Failed Event Manager in the MDEJBROL job are shown in Example 2-47.

*Example 2-47 Extract of job MDEJBROL showing the commands that set up the EJBROLE profile for WBIOperator role*

---

```
/* Define roles required by Failed Event Mgr */
RDELETE EJBROLE MD.WBIOperator
RDEFINE EJBROLE MD.WBIOperator +
UACC(NONE) +
OWNER(WASCFG) +
DATA('MDCELL WBIOperator role')
PERMIT MD.WBIOperator +
CLASS(EJBROLE) +
RESET
PERMIT MD.WBIOperator +
CLASS(EJBROLE) +
ID(MDCFG, MDBPEADG, MDEVTADG) ACCESS(READ)
```

---

2. Launch the Failed Event Manager by performing the following steps:
  - a. Log in to the Integrated Solutions Console.
  - b. Click **Integration Applications** → **Failed Event Manager**.

The Failed Event Manager shown in Figure 2-141 allows you to search for failed events. There are seven default searches and one custom. One of these searches may suit your needs more than any other. If there is a system failure, you may narrow your search by either destination or date to help you achieve your desired result set.

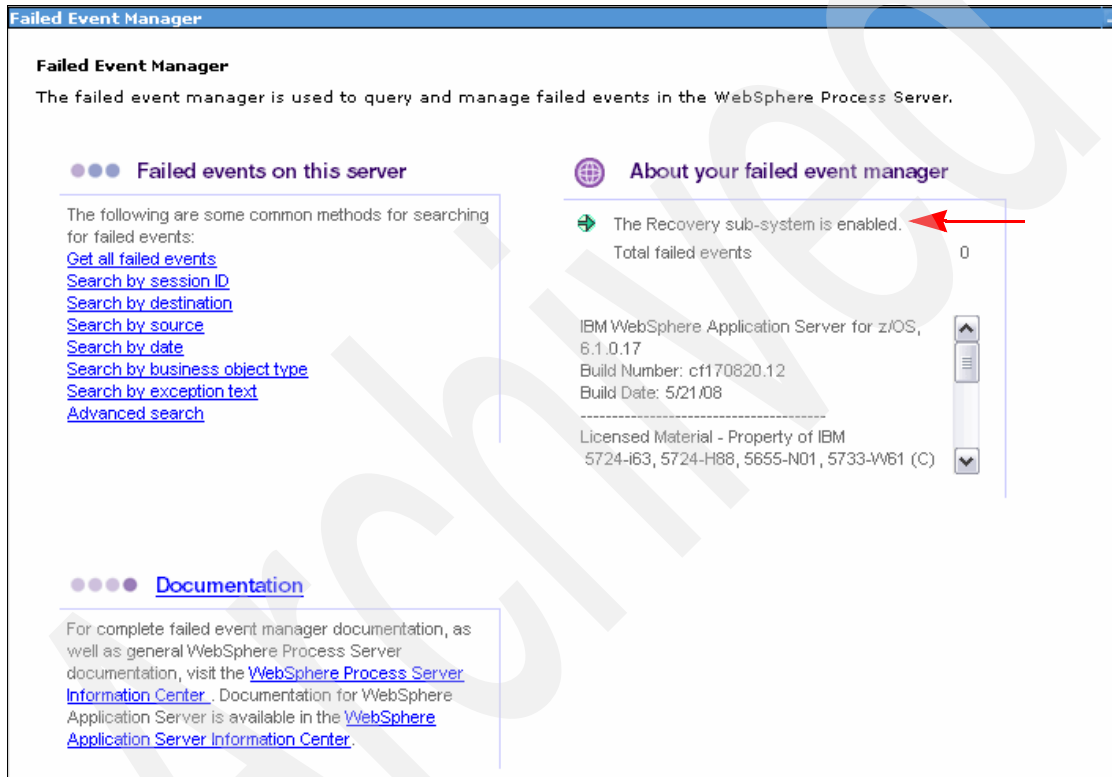


Figure 2-141 Failed Event manager main panel

**Important:** If the About your failed event manager section says that the Recovery sub-system is disabled, verify that the SCA container is started. In this configuration, it is the mdcl01\_WPS. If this does not enable the recovery sub-system, then review the *Failed event manager recovery subsystem is disabled* article available at the IBM support Web page:

<http://www.ibm.com/support/docview.wss?rs=2307&uid=swg21293460>

When your search is complete, you may need to take some action. Administrators and Operators are allowed to take action on the Failed Event Manager. If you are in another role, you will not see the buttons shown in Figure 2-142.

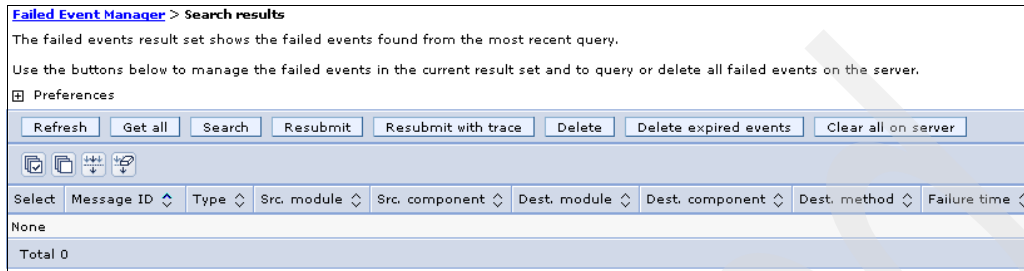


Figure 2-142 Failed Event Manager actions

The Failed Event Manager shows you information about the failed event, so that you may take some action. If the destination module was stopped and this was the reason that the event failed, resubmit the event through the following steps:

1. Check the box in the select column next to the event you wish to resubmit.
2. Click **Resubmit**.
3. Click **Refresh**.

This clears the event. If it still shows up with a new failure time, resubmit with trace to learn why the event failed. Perform the following steps:

1. Check the box in the select column next to the event you wish to resubmit.
2. Click **Resubmit with trace**.
3. From the Resubmit with trace panel, specify the Trace Control field with trace specification.
4. Click **Resubmit**.

**Important:** You cannot resubmit an event that has expired. If the event has not expired, you can edit the expiration date prior to resubmitting.

When a failed event has expired or you do not wish to resubmit, then you want to delete this event. There are three buttons you can use to do this:

- ▶ **Delete**  
Use this button if you only want to delete a specific event
- ▶ **Delete expired events**  
Use this button if you want to delete any events with an expired date
- ▶ **Clear all on server**  
Use this button if you want to delete all events in the Failed Event Manager.

# Incorporating WebSphere Business Services Fabric into a production topology on z/OS

This chapter provides detailed instructions on how to incorporate WebSphere Business Services Fabric into a production WebSphere Process Server for z/OS topology.

The structure of this chapter is similar to that for configuring WebSphere Business Services Fabric on non-z/OS platforms, described in Chapter 3, “Incorporating WebSphere Business Services Fabric into a production topology on z/OS” on page 227.

This chapter contains the following sections:

- ▶ “Introduction” on page 228
- ▶ “Planning and preparing the Fabric installation” on page 230
- ▶ “Installing Fabric in a clustered environment” on page 231
- ▶ “Add Fabric resources to WebSphere Process Server for z/OS” on page 245
- ▶ “Verify the Fabric installation and configuration” on page 278
- ▶ “Install and test the sample application” on page 282

## 3.1 Introduction

WebSphere Business Services Fabric for z/OS is a comprehensive Service-Oriented Architecture (SOA)-based offering to deliver dynamic SOA that uses existing IT assets and delivers business value incrementally. For detailed information about WebSphere Business Services Fabric concepts, architecture, and the development of dynamic SOA applications, refer to Redbooks publication *Getting Started with IBM WebSphere Business Services Fabric V6.1*, SG24-7614.

This chapter focuses on incorporating WebSphere Business Services Fabric into an existing WebSphere Process Server production topology.

We tested two topologies:

- ▶ A single-cluster topology with WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS in the same cluster.
- ▶ A two-clusters topology, which is the z/OS equivalent of the WebSphere Process Server Remote Messaging and Remote Support topology pattern (also known as the gold pattern).

In this topology, one cluster runs the WebSphere Process Server for z/OS Support and Messaging infrastructure while a second cluster supports Business Process Choreographer (BPC), WebSphere Business Services Fabric for z/OS, and the applications.

Configuring WebSphere Business Services Fabric for z/OS is similar regardless of the topology. In this chapter most of the illustrations show the single-cluster topology. Where there are differences for the two-clusters topology, these are highlighted.

Figure 3-1 on page 229 shows the two-clusters topology pattern. It also shows where WebSphere Business Services Fabric for z/OS components are added to it.

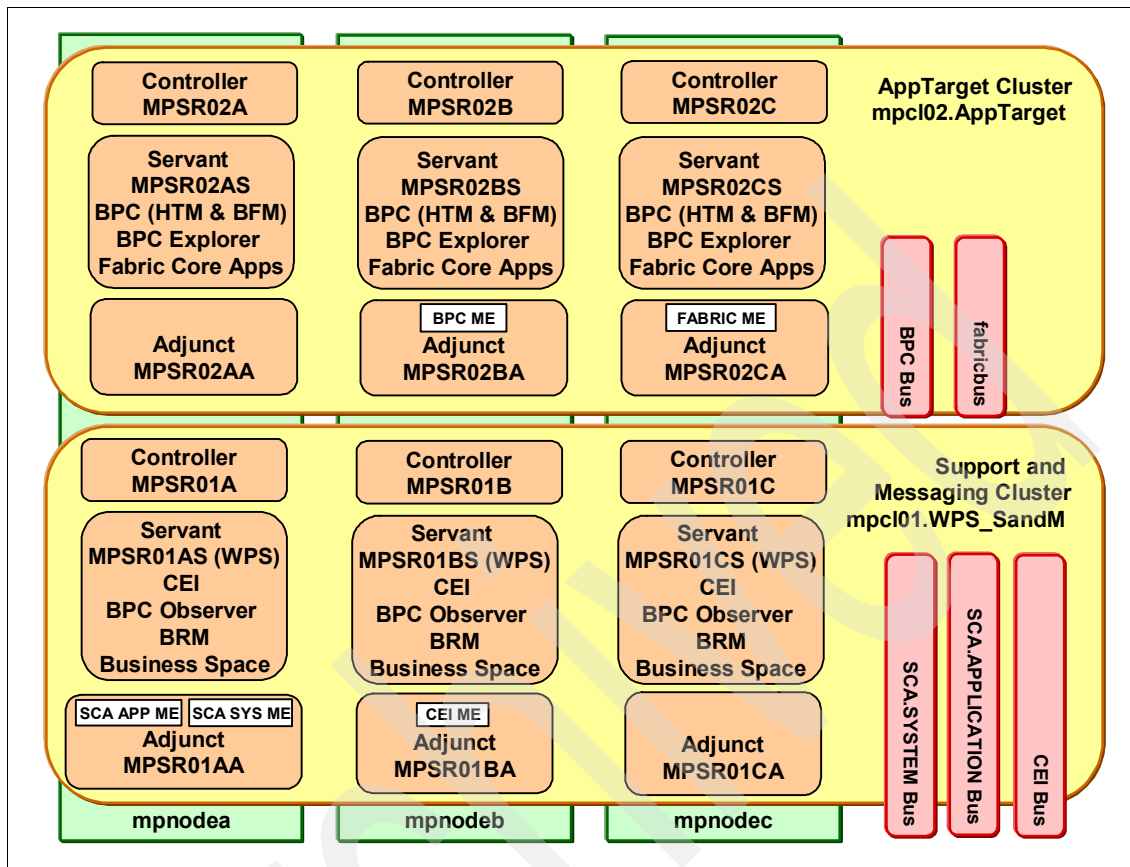


Figure 3-1 WebSphere Business Services Fabric for z/OS in the two-clusters topology

When you add WebSphere Business Services Fabric for z/OS to the topology, the following components are added:

- ▶ A DB2 database is added to the DB2 server. We called ours MDFABDB.
- ▶ A service integration bus, named fabricbus, is added.

In the two-clusters topology we added this locally to the cluster running BPC and Fabric (mpcl02.AppTarget cluster in Figure 3-1).

- ▶ The WebSphere Business Services Fabric for z/OS core application EAR files are added to the cluster.

In the two-clusters topology the Fabric EAR files are deployed to the AppTarget cluster, (mdcl02.AppTarget in Figure 3-1).

- ▶ WebSphere Business Services Fabric for z/OS events are emitted to the JMS destinations present in the support cluster.

## 3.2 Planning and preparing the Fabric installation

This section describes the software pre-requisites and the installation of the product using SMP/E. It introduces the additional material we provide with this Redbooks publication. Finally there are some comments about creating the deployment directory.

### 3.2.1 Software pre-requisites

The pre-requisites for WebSphere Business Services Fabric for z/OS are found in the IBM support article *Failed event manager recovery subsystem is disabled*, available at the following Web page:

<http://www.ibm.com/support/docview.wss?rs=36&uid=swg27012676>

The key requirement is WebSphere Process Server for z/OS V6R1M2. The requirement for LDAP depends on whether you decide to use LDAP or a local SAF user registry.

### 3.2.2 Installing WebSphere Business Services Fabric in SMP/E

The instructions for installing WebSphere Business Services Fabric for z/OS into SMP/E are found in *WebSphere Business Service Fabric 6.1.2 Program Directory*, GI11-2874-01, which is shipped with the product. The Program Directory can also be downloaded from IBM Publications at the following Web page:

<http://www.ibm.com/shop/publications/order>

The product ZFS occupies around 2500 CYL of disk space.

When SMP/E work is finished, the WebSphere Business Services Fabric product ZFS is typically mounted read-only at /usr/lpp/webify/V6R1M2. That copy is shared across the sysplex. Configuration of WebSphere Business Services Fabric for z/OS starts once the product has been made available in the UNIX System Services file system.

### 3.2.3 Sample JCL and shell scripts used in this chapter

The sample JCL and shell scripts described in this chapter can all be found in the /zos\_wbsf directory in Appendix A, “Additional material” on page 351. The JCL is in the WBSF.CNTL library, the DDL is in the WBSF.SQL library, and the shell scripts in the /wpswork directory under /zos\_wbsf.



### 3.2.4 Creating the deployment directory

Under the heading “Creating the Deployment Directory”, the WebSphere Business Services Fabric for z/OS Information Center recommends that you take copies of the product ZFS so that you can customize files such as the Data Definition Language (DDL) statements and the JACL. However after completing the configuration we found that it was not necessary to take a copy of the entire ZFS. It is sufficient to copy any files you need to change into a work directory where you can edit them and then transfer them into the appropriate directories within the node’s file system.

## 3.3 Installing Fabric in a clustered environment

This section describes the steps in installing WebSphere Business Services Fabric into a clustered environment.

The initial steps are:

1. Copy the Fabric artifacts.  
See page 231.
2. Create required RACF user ID, groups, and profiles.  
See page 234.
3. Create the Fabric databases.  
See page 239.

In Section 3.4, “Add Fabric resources to WebSphere Process Server for z/OS” on page 245, you perform the majority of the tasks to add Fabric into the cell.

### 3.3.1 Copy the Fabric artifacts

To install WebSphere Business Services Fabric for z/OS, several jars and product version files have to be copied to the WebSphere Process Server for z/OS nodes from the WebSphere Business Services Fabric for z/OS product ZFS.

The WebSphere Business Services Fabric for z/OS Information Center lists the files to be copied. You can see how to copy these files individually in the non-z/OS equivalent of this section, Section 3.3.1, “Copy the Fabric artifacts” on page 231 of *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665. It is also necessary to copy the registration file WBSF.registrar into each node’s properties/version directory.

We created a shell script called `copyartifcat.sh` (Example 3-1) to do all the necessary copies. (The copy commands (`cp`) in Example 3-1 go on one line but some have been split across 2 lines to fit on the page. Line breaks are indicated by a backslash.)

**Note:** The artifacts must be copied to the `../profiles/default/BusinessSpace/registryData` on each node. The directories `../BusinessSpace/registryData` are created under `/profiles/default` for each node when you configured Business Space.

Confirm that the directories exist before running the `copyartifcat.sh`.

#### Example 3-1 Sample shell script `copyartifcat.sh`

```
#!/bin/sh
# copyartifacts.sh
# Copy fabric artifacts
#
#-----
# Setup
#
# Set the following 4 variables to match your environment
#
FABRIC=/usr/lpp/webifyMK/V6R1M2
DMNODE=/wasv6lconfig/mdcell/mdmnode/DeploymentManager
NODEA=/wasv6lconfig/mdcell/mdnodea/AppServer
NODEB=/wasv6lconfig/mdcell/mdnodeb/AppServer
#
echo 'Copying to '$DMNODE'/lib/ext'
cp $FABRIC/runtime/fabric-da-api.jar $DMNODE/lib/ext/.
cp $FABRIC/runtime/fabric-da-model.jar $DMNODE/lib/ext/.
cp $FABRIC/runtime/fabric-da-sca.jar $DMNODE/lib/ext/.
cp $FABRIC/runtime/fabric-da-scdl.jar $DMNODE/lib/ext/.
cp $FABRIC/runtime/fabric-types.jar $DMNODE/lib/ext/.
cp $FABRIC/runtime/fabric-da-api.jar $NODEA/lib/ext/.
echo 'Copying to '$NODEA'/lib/ext'
cp $FABRIC/runtime/fabric-da-model.jar $NODEA/lib/ext/.
cp $FABRIC/runtime/fabric-da-sca.jar $NODEA/lib/ext/.
cp $FABRIC/runtime/fabric-da-scdl.jar $NODEA/lib/ext/.
cp $FABRIC/runtime/fabric-types.jar $NODEA/lib/ext/.
echo 'Copying to '$NODEB'/lib/ext'
cp $FABRIC/runtime/fabric-da-api.jar $NODEB/lib/ext/.
cp $FABRIC/runtime/fabric-da-model.jar $NODEB/lib/ext/.
cp $FABRIC/runtime/fabric-da-sca.jar $NODEB/lib/ext/.
cp $FABRIC/runtime/fabric-da-scdl.jar $NODEB/lib/ext/.
cp $FABRIC/runtime/fabric-types.jar $NODEB/lib/ext/.
echo 'Copying to '$DMNODE'/properties/version'
cp $FABRIC/configuration/Runtime/WBSF.product $DMNODE/properties/version/.
cp $FABRIC/configuration/Runtime/WBSFengine.component $DMNODE/properties/version/.
echo 'Copying to '$NODEA'/properties/version'
cp $FABRIC/configuration/Runtime/WBSF.product $NODEA/properties/version/.
```

```

cp $FABRIC/configuration/Runtime/WBSFEngine.component $NODEA/properties/version/.
echo 'Copying to '$NODEB'/properties/version'
cp $FABRIC/configuration/Runtime/WBSF.product $NODEB/properties/version/.
cp $FABRIC/configuration/Runtime/WBSFEngine.component $NODEB/properties/version/.
echo 'Copying to '$DMNODE'/profiles/default/BusinessSpace/registryData'
cp $FABRIC/configuration/Runtime/wbsfEndpoints.xml
$DMNODE/profiles/default/BusinessSpace/registryData/.
echo 'Copying to '$NODEA'/profiles/default/BusinessSpace/registryData'
cp $FABRIC/configuration/Runtime/wbsfEndpoints.xml
$NODEA/profiles/default/BusinessSpace/registryData/.
echo 'Copying to '$NODEB'/profiles/default/BusinessSpace/registryData'
cp $FABRIC/configuration/Runtime/wbsfEndpoints.xml
$NODEB/profiles/default/BusinessSpace/registryData/.
echo 'Creating symbolic links'
ln -s $FABRIC/configuration/Runtime/zos/WBSF.registrar \
$DMNODE/properties/version/WBSF.registrar
ln -s $FABRIC/configuration/Runtime/zos/WBSF.registrar \
$NODEA/properties/version/WBSF.registrar
ln -s $FABRIC/configuration/Runtime/zos/WBSF.registrar \
$NODEB/properties/version/WBSF.registrar
echo 'Setting permissions'
chmod -Rh 775 $DMNODE/lib/ext/*
chown -Rh MOWNER:MDCFG $DMNODE/lib/ext/*
chmod -Rh 775 $NODEA/lib/ext/*
chown -Rh MOWNER:MDCFG $NODEA/lib/ext/*
chmod -Rh 775 $NODEB/lib/ext/*
chown -Rh MOWNER:MDCFG $NODEB/lib/ext/*
chmod -Rh 775 $DMNODE/properties/version/*
chown -Rh MOWNER:MDCFG $DMNODE/properties/version/*
chmod -Rh 775 $NODEA/properties/version/*
chown -Rh MOWNER:MDCFG $NODEA/properties/version/*
chmod -Rh 775 $NODEB/properties/version/*
chown -Rh MOWNER:MDCFG $NODEB/properties/version/*
chmod -Rh 775 $DMNODE/profiles/default/BusinessSpace/registryData/*
chown -Rh MOWNER:MDCFG $DMNODE/profiles/default/BusinessSpace/registryData/*
chmod -Rh 775 $NODEA/profiles/default/BusinessSpace/registryData/*
chown -Rh MOWNER:MDCFG $NODEA/profiles/default/BusinessSpace/registryData/*
chmod -Rh 775 $NODEB/profiles/default/BusinessSpace/registryData/*
chown -Rh MOWNER:MDCFG $NODEB/profiles/default/BusinessSpace/registryData/*
chmod -Rh 775 $DMNODE/properties/version/WBSF.registrar
chown -Rh MOWNER:MDCFG $DMNODE/properties/version/WBSF.registrar
chmod -Rh 775 $NODEA/properties/version/WBSF.registrar
chown -Rh MOWNER:MDCFG $NODEA/properties/version/WBSF.registrar
chmod -Rh 775 $NODEB/properties/version/WBSF.registrar
chown -Rh MOWNER:MDCFG $NODEB/properties/version/WBSF.registrar

```

You can invoke copyartifact.sh from the OMVS command line or run sample job MDCPYART, which invokes copyartifact.sh. Notice that the wbsfEndpoint.xml is copied to ../profiles/default/BusinessSpace/registryData in all the nodes in the cluster.

### 3.3.2 Create required RACF user ID, groups, and profiles

The WebSphere Business Services Fabric for z/OS Information Center describes how to use three different types of user registry. When using RACF as the user registry for the cell, consult the Information Center article *Configuring security*, available at the following Web page:

[http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.ws.fabric.install612.doc/fpiZOS/task/t\\_%20configuring\\_security.html](http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.ws.fabric.install612.doc/fpiZOS/task/t_%20configuring_security.html)

Some RACF groups and user IDs are required for J2C authentication aliases. Plan some groups and user IDs to permit to various roles. We planned the groups and user IDs we would use with a spreadsheet. The spreadsheet is included in Appendix A, “Additional material” on page 351.

We created sample job MDRACF (Example 3-2) to define all the groups and user IDs we required for use with Fabric.

*Example 3-2 Sample job MDRACF to define user IDs and groups for Fabric*

---

```
//MDRACF JOB (0), 'SUWS100', CLASS=A, MSGCLASS=H, REGION=OM,
// NOTIFY=SENIOR
/*JOBPARM SYSAFF=SC42
/*****
/* Make sure you edit this file with CAPS OFF. */
/* Run this job using a userid with RACF SPECIAL. */
/* Change MD to your two character cell prefix. */
/* For example, MD. (Use upper case.) */
/* Set the OWNER as you want. */
/* Set the UIDs and GIDs so they are unique for this cell.*/
/*****
/*
/* Add groups and userids for J2C alias and JMS alias
/*
//RACF EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
/* Group for FAB administrators */
AG MDFADMG SUPGROUP(WASCFG) OMVS(AUTOGID) +
DATA('MD Fabric Admins')
/* Fabric admin userid */
AU MDFABADM DFLTGRP(MDFADMG) OMVS(AUTOUID +
HOME(/var/WebSphere/home/MDFADMG) +
PROGRAM(/bin/sh)) +
```

```

NAME('MD Fabric Admin')          +
OWNER(WASCFG)
PW  USER(MDFABADM) NOINTERVAL
ALU MDFABADM      PASSWORD(MDFABADM) NOEXPIRED
/*  Group for Fabric Governance administrators */
AG  MDFGADMG      SUPGROUP(WASCFG) OMVS(AUTOUID) +
DATA('MD Fabric Governance Admins')
/*  Fabric governance admin userid */
AU  MDFGADM      DFLTGRP(MDFGADMG) OMVS(AUTOUID +
HOME(/var/WebSphere/home/MDFGADMG) +
PROGRAM(/bin/sh)) +
NAME('MD Fab Gov Adm') +
OWNER(WASCFG)
PW  USER(MDFGADM) NOINTERVAL
ALU MDFGADM      PASSWORD(MDFGADM) NOEXPIRED
/*  Group for Fabric Performance Usersa */
AG  MDFPUG      SUPGROUP(WASCFG) OMVS(AUTOUID) +
DATA('MD FAB PERF USER')
/*  Fabric Performance User userid */
AU  MDFPU      DFLTGRP(MDFPUG) OMVS(AUTOUID +
HOME(/var/WebSphere/home/MDFPUG) +
PROGRAM(/bin/sh)) +
NAME('MD Fabric Performance Users') +
OWNER(WASCFG)
PW  USER(MDFPU) NOINTERVAL
ALU MDFPU      PASSWORD(MDFPU) NOEXPIRED
/*  Group for Fabric Subscriber Managers */
AG  MDFSMG      SUPGROUP(WASCFG) OMVS(AUTOUID) +
DATA('MD Fabric Subscriber Managers')
/*  Fabric Subscriber Manager userid */
AU  MDFSMU      DFLTGRP(MDFSMG) OMVS(AUTOUID +
HOME(/var/WebSphere/home/MDFSMG) +
PROGRAM(/bin/sh)) +
NAME('MD FAB SUB MGR') +
OWNER(WASCFG)
PW  USER(MDFSMU) NOINTERVAL
ALU MDFSMU      PASSWORD(MDFSMU) NOEXPIRED
/*  Group for Fabric Basic Users */
AG  MDFBUG      SUPGROUP(WASCFG) OMVS(AUTOUID) +
DATA('MD Fabric Basic Users')
/*  Fabric Basic User userid */
AU  MDFBU      DFLTGRP(MDFBUG) OMVS(AUTOUID +
HOME(/var/WebSphere/home/MDFBUG) +
PROGRAM(/bin/sh)) +
NAME('MD FAB Base User') +

```

```

                                OWNER(WASCFG)
PW  USER(MDFBU)      NOINTERVAL
ALU MDFBU PASSWORD(MDFBU) NOEXPIRED
/*  Group for Fabric Studio Users      */
AG  MDFSUG           SUPGROUP(WASCFG) OMVS(AUTOGID) +
                                DATA('MD Fabric Studio Users')
/*  Fabric Studio User userid          */
AU  MDFSU            DFLTGRP(MDFSUG)  OMVS(AUTOUID +
                                HOME(/var/WebSphere/home/MDFSUG) +
                                PROGRAM(/bin/sh)) +
                                NAME('MD FAB Studio User') +
                                OWNER(WASCFG)
PW  USER(MDFSU)      NOINTERVAL
ALU MDFSU            PASSWORD(MDFSU) NOEXPIRED
/* Permit the role groups to the APPL profile */
PE  MD CLASS(APPL) ID(MDFADMG, MDFGADM, MDFPUG, MDFSMG)
PE  MD CLASS(APPL) ID(MDFBUG, MDFSUG)
SETR RACLIST(APPL) REFRESH
/*  Fabric Bus Schema group          */
AG  MD01F           SUPGROUP(WASCFG) OMVS(AUTOGID) +
                                DATA('MD SIB01 FABRIC SCHEMA')
/*  Allow alias to switch to bus schema */
/*  This connect is needed in DB2 V8.1 */
/*  but not in DB2 9.1                */
/*  CONNECT MDDBU      GROUP(MD01F)    */
/*
//*****
//**  CREATE HOME DIRECTORIES FOR USERIDS      */
//*****
//CREHOME EXEC PGM=IKJEFT01,REGION=0M,TIME=1440
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
BPXBATCH SH +
mkdir /var/WebSphere/home/MDFADMG; +
chown -Rh MDFABADM:MDFADMG /var/WebSphere/home/MDFADMG; +
chmod -Rh 775 /var/WebSphere/home/MDFADMG; +
mkdir /var/WebSphere/home/MDFGADMG; +
chown -Rh MDFGADM:MDFGADMG /var/WebSphere/home/MDFGADMG; +
chmod -Rh 775 /var/WebSphere/home/MDFGADMG; +
mkdir /var/WebSphere/home/MDFPUG; +
chown -Rh MDFPU:MDFPUG /var/WebSphere/home/MDFPUG; +
chmod -Rh 775 /var/WebSphere/home/MDFPUG; +
mkdir /var/WebSphere/home/MDFSMG; +
chown -Rh MDFSMU:MDFSMG /var/WebSphere/home/MDFSMG; +

```

```

chmod -Rh 775 /var/WebSphere/home/MDFSMG; +
mkdir /var/WebSphere/home/MDFBUG; +
chown -Rh MDFBU:MDFBUG /var/WebSphere/home/MDFBUG; +
chmod -Rh 775 /var/WebSphere/home/MDFBUG; +
mkdir /var/WebSphere/home/MDFSUG; +
chown -Rh MDFSU:MDFSUG /var/WebSphere/home/MDFSUG; +
chmod -Rh 775 /var/WebSphere/home/MDFSUG; +
1> /tmp/dbmkdir.out +
2> /tmp/dbmkdir.err
/*****/
/* STEPS TO COPY THE OUTPUT THE JOB LOG */
/*****/
//CPOUT EXEC PGM=IKJEFT01,REGION=OM
//SYSEXEC DD DISP=SHR,DSN=BBO6142.MDDMNODE.SBBOEXEC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
    BBOHFSWR '/tmp/dbmkdir.out'
    BBOHFSWR '/tmp/dbmkdir.err'
/*

```

---

Next, we created job MDEJBROL to define RACF EJBROLE profiles for the roles related to WebSphere Business Services Fabric. This job also permitted the necessary groups to use the roles.

*Example 3-3 Sample job MDEJBROL to define EJBROLE profiles for Fabric*

```

//MDEJBROL JOB (0),'RACFADM',CLASS=A,MSGCLASS=A,REGION=OM,
// NOTIFY=&SYSUID
/*****/
/* Make sure you edit this file with CAPS OFF. */
/* Run this job using a userid with RACF SPECIAL. */
/* The RACF userids and groups used here must exist. */
/* Run job XXRACF first. */
/* Change MD to your two character cell prefix. */
/* For example, MD. (Use upper case.) */
/* Set the OWNER as you want. */
/*****/
/*
/* Create EJB role profiles for WPS
/*
//RACFDBA EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *

```

```

/* Define roles required by BPE */
RDELETE EJBROLE MD.FabricAdministrator
RDEFINE EJBROLE MD.FabricAdministrator +
                UACC(NONE) +
                OWNER(WASCFG) +
                DATA('MD FabricAdministrator role')
PERMIT MD.FabricAdministrator +
        CLASS(EJBROLE) +
        RESET
PERMIT MD.FabricAdministrator +
        CLASS(EJBROLE) +
        ID(MDFABADG) ACCESS(READ)
RDELETE EJBROLE MD.FabricGovernanceAdministrator
RDEFINE EJBROLE MD.FabricGovernanceAdministrator +
                UACC(NONE) +
                OWNER(WASCFG) +
                DATA('MD FabricGovernanceAdministrator role')
PERMIT MD.FabricGovernanceAdministrator +
        CLASS(EJBROLE) +
        RESET
PERMIT MD.FabricGovernanceAdministrator +
        CLASS(EJBROLE) +
        ID(MDFGADMG) +
        ACCESS(READ)
/* Define roles required by Human Task mgr */
RDELETE EJBROLE MD.FabricPerformanceUser
RDEFINE EJBROLE MD.FabricPerformanceUser +
                UACC(NONE) +
                OWNER(WASCFG) +
                DATA('MD FabricPerformanceUser role')
PERMIT MD.FabricPerformanceUser +
        CLASS(EJBROLE) +
        RESET
PERMIT MD.FabricPerformanceUser +
        CLASS(EJBROLE) +
        ID(MDFPUG,MDFABADG) +
        ACCESS(READ)
RDELETE EJBROLE MD.FabricSubscriberManager
RDEFINE EJBROLE MD.FabricSubscriberManager +
                UACC(NONE) +
                OWNER(WASCFG) +
                DATA('MD FabricSubscriberManager role')
PERMIT MD.FabricSubscriberManager +
        CLASS(EJBROLE) +
        RESET

```



```

PERMIT MD.FabricSubscriberManager +
CLASS(EJBROLE) +
ID(MDFSMG,MDFABADG) +
ACCESS(READ)
RDELETE EJBROLE MD.FabricBasicUser
RDEFINE EJBROLE MD.FabricBasicUser +
UACC(NONE) +
OWNER(WASCFG) +
DATA('MD FabricBasicUser role')
PERMIT MD.FabricBasicUser +
CLASS(EJBROLE) +
RESET
PERMIT MD.FabricBasicUser +
CLASS(EJBROLE) +
ID(MDFBUG,MDFABADMG) +
ACCESS(READ)
RDELETE EJBROLE MD.FabricStudioUser
RDEFINE EJBROLE MD.FabricStudioUser +
UACC(NONE) +
OWNER(WASCFG) +
DATA('MD FabricStudioUser role')
PERMIT MD.FabricStudioUser +
CLASS(EJBROLE) +
RESET
PERMIT MD.FabricStudioUser +
CLASS(EJBROLE) +
ID(MDFSUG,MDFABADG) +
ACCESS(READ)
SETR RACLIST(EJBROLE) REFRESH
SEARCH MASK(MD) CLASS(EJBROLE)
/*

```

---

Our security policy for these roles was to make all the roles UACC(NONE), to permit an “administrator” group to all the roles, and then have a group dedicated to providing access to a particular role.

### 3.3.3 Create the Fabric databases

WebSphere Business Services Fabric for z/OS requires a database for its tables. You will probably want to create a second database to hold the tables related to the Fabric Service Integration Bus. The names we used for these two databases accessed by our MD cell were MDFABDB and MD01F respectively.

The DDL is provided at /usr/lpp/webify/V6R1M2/configuration/DB2Scripts.

We created sample job MDCPYDDL to convert the DDL from ASCII to EBCDIC and then copied it from to z/OS sequential files so we could edit it.

The source file names and the z/OS sequential dataset names are shown in Table 3-1.

*Table 3-1 Source DDL files and their equivalent EBCDIC z/OS files*

Description	Source file name	z/OS sequential dataset name
Database, stogroup and tablespaces	define_db.sql	WASCFG.MDCELL.SQL.MDFABDB
Tables	define_tables.sql	WASCFG.MDCELL.SQL.MDFABTB
Foreign-keys	define_foreignkeys.sql	WASCFG.MDCELL.SQL.MDFABFK
Indexes	define_index_objects.sql	WASCFG.MDCELL.SQL.MDFABIO
Sequence Objects	define_sequence_objects.sql	WASCFG.MDCELL.SQL.MDFABSO
Unique indexes	define_unique_index.sql	WASCFG.MDCELL.SQL.MDFABUI

Perform the following steps to create the Fabric databases.

1. Edit the DDL that creates the database, storage group and tablespaces to match your requirements (Source file define\_db.sql and z/OS file WASCFG.MDCELL.SQL.MDFABDB).

The changes we made to the MDFABDB file are listed below:

- Commented out the SET CURRENT SQLID at the start of all the DDL files because it is better to specify this in the JCL that are used to execute the DDL.
- Made the following changes:
  - Changed the STOGROUP name from WSFST02
  - VOLUMES from (WBIUS5,WBIUS4,WBIUS3) VCAT WSDB2 to our values.

We then commented out the CREATE STOGROUP statement because we were using a STOGROUP that already exists.

- Commented out the GRANT USE OF STOGROUP because we are using a STOGROUP that already exists, and the same schema as the one you are using for the other WebSphere Process Server databases. The grant will have already been issued in this case.

- Commented out the CREATE DATABASE statement because we created the database separately using job MDCREDB.

**Note:** Leaving correct but commented-out CREATE DATABASE and CREATE STOGROUP statements in the DDL serves to act as documentation of which storage group is being used. This is better than deleting the statements entirely.

- Changed the database name from FABRICDB to MDFABDB.
  - Changed the BUFFERPOOL assignment to the bufferpool planned (BP11) and added an INDEXBP BP12 statement to the CREATE DATABASE.
  - Changed all PRIQTY and SECQTY statements to -1.
2. We created sample job MDCREDB to create the MDFABDB and MD01FDB databases, and to issue required GRANTs to the schema groups.

*Example 3-4 Sample job to create the Fabric databases*

```
//MDCREDB JOB (0), 'MDADMIN', CLASS=A, REGION=OM, NOTIFY=&SYSUID
/*JOBPARM S=SC42
/*****/
/* Run this job using a userid with DB2 SYSADM authority */
/* or grant DBADM to the <sqlid> on SET CURRENT SQLID. */
/* Set the SYSAFF to the LPAR for this node. */
/* Change <xx> to your two character cell prefix. */
/* For example, MD. (Use upper case.) */
/* The DB2 system administrator should review all the */
/* names and bufferpools used. */
/* Change all the dataset names and plan name to match */
/* those in use with your DB2 system. */
/* Set the SYSIN to the dataset holding the DDL. */
/*****/
//JOB LIB DD DISP=SHR, DSN=DB9F9.SDSNEXIT
// DD DISP=SHR, DSN=DB9F9.SDSNLOAD
/*
/* Before running this:
/* The userid on the SET CURRENT SQLID statement must
/* have DB2 SYSADM authority and EXECUTE on PLAN DSNTDP2
/*
/* Create Fabric database and tablespaces.
/*
//CREATE EXEC PGM=IKJEFT01, DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
```

```

//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
DSN SYSTEM(D9FG)
RUN PROGRAM(DSNTEP2) PLAN(DSNTEP91) LIB('DB9FU.RUNLIB.LOAD')
//SYSIN DD *
CREATE DATABASE MDFABDB
STOGROUP MDWPSSG
BUFFERPOOL BP11
INDEXBP BP12
CCSID UNICODE;

CREATE DATABASE MDSOFADB
STOGROUP MDWPSSG
BUFFERPOOL BP3
INDEXBP BP4;

GRANT DBADM ON DATABASE MDFABDB TO MDWPS;
GRANT DBADM ON DATABASE MDS01FDB TO MDS01F;
GRANT CREATEIN, DROPIN ON SCHEMA MDS01F TO MDS01F;
GRANT USE OF STOGROUP MDWPSSG TO MDS01F;
GRANT USE OF BUFFERPOOL BP3 TO MDS01F;
GRANT USE OF BUFFERPOOL BP4 TO MDS01F;
/*

```

---

3. Edit the DDL that creates the tables to match your requirements. (Source file `define_tables.sql` and z/OS file `WASCFG.MDCELL.SQL.MDFABTB`.)  
We changed the database name from `FABRICDB` to `MDFABDB`.
4. Review the other DDL files. We found that no further editing was required.
5. We copied the customized DDL from the variable-length sequential files to a fixed-block, LRECL 80 partitioned dataset so it could be executed by `DSNTEP2`. The partitioned dataset was called `WASCFG.MDCELL.WBSF.SQL`.
6. We created sample job `MDDDLFAB` (Example 3-5 on page 243) in `WASCFG.MDCELL.WBSF.CNTL` to execute the DDL. Make sure that you execute the DDL that creates the tablespaces first (`MDFABDB`) followed by the DDL that creates the tables (`MDFABTB`), and then the other DDL that creates the indexes. Note the order of the concatenation in the `SYSIN DD` of job `MDDDLFAB`.

*Example 3-5 Sample job MDDDLFAB to execute the Fabric DDL*

---

```
//MDDDLFAB JOB (0),'MDADMIN',CLASS=A,REGION=OM,NOTIFY=&SYSUID
/*JOBPARM S=SC42
/*****/
/* Run this job using a userid with DB2 SYSADM authority */
/* or grant DBADM to the <sqlid> on SET CURRENT SQLID. */
/* Set the SYSAFF to the LPAR for this node. */
/* Change <xx> to your two character cell prefix. */
/* For example, MD. (Use upper case.) */
/* The DB2 system administrator should review all the */
/* names and bufferpools used. */
/* Change all the dataset names and plan name to match */
/* those in use with your DB2 system. */
/* Set the SYSIN to the dataset holding the DDL. */
/*****/
//JOB LIB DD DISP=SHR,DSN=DB9F9.SDSNEXIT
// DD DISP=SHR,DSN=DB9F9.SDSNLOAD
/*
/* Before running this:
/* The userid on the SET CURRENT SQLID statement must
/* have DB2 SYSADM authority and EXECUTE on PLAN DSNTPA91
/*
/* Create Fabric tables and indexes etc.
/*
//CREATE EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DSN SYSTEM(D9FG)
RUN PROGRAM(DSNTP2) PLAN(DSNTP91) LIB('DB9FU.RUNLIB.LOAD')
//SYSIN DD *
SET CURRENT SQLID = 'MDWPS' ;
// DD DISP=SHR,DSN=WASCFG.MDCELL.WBSF.SQL(MDFABDB)
// DD DISP=SHR,DSN=WASCFG.MDCELL.WBSF.SQL(MDFABTB)
// DD DISP=SHR,DSN=WASCFG.MDCELL.WBSF.SQL(MDFABUI)
// DD DISP=SHR,DSN=WASCFG.MDCELL.WBSF.SQL(MDFABIO)
// DD DISP=SHR,DSN=WASCFG.MDCELL.WBSF.SQL(MDFABFK)
// DD DISP=SHR,DSN=WASCFG.MDCELL.WBSF.SQL(MDFABSO)
```

---

7. If your DB2 for z/OS system is using RACF for authorization checking, and if you have connected the database J2C authentication alias user ID to the RACF group of the schema (MDWPS for our cell), then the database authentication alias will need no more authority because it has privileges equivalent to the owner of the tables.

If you have not connected the database J2C authentication alias user ID to the schema's RACF group because you do not want the authentication alias to have implicit owner privileges, issue GRANT statements for the Fabric tables and sequences.

We created the necessary grant statements in member GRANTFAB in the WASCFG.MDCELL.WPS.SQL dataset. Then we created sample job MDGRANT in the .CNTL dataset to execute these statements.

Execute MDGRANT now if you need to GRANT table and sequence privileges to the database J2C authentication alias user ID.

8. To create the tables for the Fabric Service Integration Bus, we copied the DDL we had used for one of the other buses and changed the schema to MD01F (where the "F" stands for Fabric). You could also run the sibDDLGenerator.sh to generate new DDL if you wanted to.

We created sample job MDDDL SI (Example 3-6) to execute this DDL.

*Example 3-6 Sample job MDDDL SI to execute the Fabric bus DDL*

---

```
//MDDDL SI JOB (0), 'MDADMIN', CLASS=A, REGION=OM, NOTIFY=&SYSUID
/*JOBPARM S=SC42
//*****
/* Run this job using a userid with DB2 SYSADM authority */
/* or grant DBADM to the <sqlid> on SET CURRENT SQLID. */
/* Set the SYSAFF to the LPAR for this node. */
/* Change <xx> to your two character cell prefix. */
/* For example, MD. (Use upper case.) */
/* The DB2 system administrator should review all the */
/* names and bufferpools used. */
/* Change all the dataset names and plan name to match */
/* those in use with your DB2 system. */
/* Set the SYSIN to the dataset holding the DDL. */
//*****
//JOB LIB DD DISP=SHR, DSN=DB9F9.SDSNEXIT
// DD DISP=SHR, DSN=DB9F9.SDSNLOAD
//*
//* Before running this:
//* The userid on the SET CURRENT SQLID statement must
//* have DB2 SYSADM authority and EXECUTE on PLAN DSNTEP2
//*
```

```

/* Create tables for Fabric SIB
/*
//CREATE EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
DSN SYSTEM(D9FG)
RUN PROGRAM(DSNTEP2) PLAN(DSNTEP91) LIB('DB9FU.RUNLIB.LOAD')
//SYSIN DD *
SET CURRENT SQLID = 'MD01F' ;
// DD DISP=SHR,DSN=WASCFG.MDCELL.WBSF.SQL(MD01F)

```

---

**Note:** There are GRANT statements for the SIB tables at the end of the MD01F DDL member so it is not necessary to run a separate GRANT job for the SIB tables.

## 3.4 Add Fabric resources to WebSphere Process Server for z/OS

This section describes how to configure resources within the cluster scope. The process to configure resources within the cluster scope consists of the following steps:

1. Set WebSphere environment variables.  
See page 246.
2. Create an authentication alias for Fabric.  
See page 247.
3. Create and configure the data sources.  
See page 248.
4. Create and configure the service integration bus.  
See page 252.
5. Create and configure the service integration bus.  
See page 256.
6. Configure the JMS provider.  
See page 259.
7. Configure a mail provider.

- See page 266.
- 8. Configure security.
  - See page 267.
- 9. Configure distributed cache.
  - See page 271.
- 10. Configure a namespace variable for CEI.
  - See page 274.
- 11. Install the Fabric core EAR files.
  - See page 275.
- 12. Grant access to the Fabric Tools Console.
  - See page 277.

At the time of writing, the JACL scripts provided to configure WebSphere Business Services Fabric for z/OS do not support a clustered environment. Therefore this section describes how we configured WebSphere Business Services Fabric for z/OS using the Integrated Solutions Console.

### 3.4.1 Set WebSphere environment variables

The DB2 Universal JDBC Provider used by the WebSphere Business Services Fabric data source uses the WebSphere environment variable `DB2UNIVERSAL_JDBC_DRIVER_PATH`. This variable is already defined at the cluster level because it is created during the configuration of WebSphere Process Server as part of the DB2 Universal Driver JDBC provider. This section describes how to check that the variable exists.

1. Log in to the Integrated Solutions Console using the administrator user (we used MDADMIN).
2. In the Integrated Solutions Console, navigate to **Environment** → **WebSphere Variables**, and in the Scope selection box select the cluster.
  - For our single-cluster topology we selected Cluster = mdcl01\_WPS.
  - For our two-clusters topology we chose the Cluster = mpcl02.AppTarget.
3. Check the value for the `DB2UNIVERSAL_JDBC_DRIVER_PATH`. It has the value of the path to the db2 client JDBC jar files. If the value is not there, set the value to the DB2 Universal JDBC driver's jar file path and save the changes.



### 3.4.2 Create an authentication alias for Fabric

This section describes the steps to create two J2C authentication aliases that are used by WebSphere Business Services Fabric to authenticate to the MDFABDB database and to the Fabric bus.

1. In the Integrated Solutions Console, navigate to **Security** → **Secure administration, applications, and infrastructure**.
2. Under **Authentication**, click **Java Authentication and Authorization Service**.
3. Click **J2C authentication data**.
4. Click **New** and enter the values given in the Database alias values column of Table 3-2. Enter the User ID and Password according to your environment. Click **OK**.
5. Click **New** to define a second alias. Enter the values given in the JMS alias values column of Table 3-2. Enter the User ID and Password according to your environment.

Table 3-2 J2C authentication alias data

Field	Database alias values	JMS alias values
Alias	Fabric_DB_Auth_Alias	Fabric_ME_Auth_Alias
User ID	MDDBU	MDJMSU
Password	MDDBU	MDJMSU
Description	Fabric database authentication alias	Fabric ME authentication alias

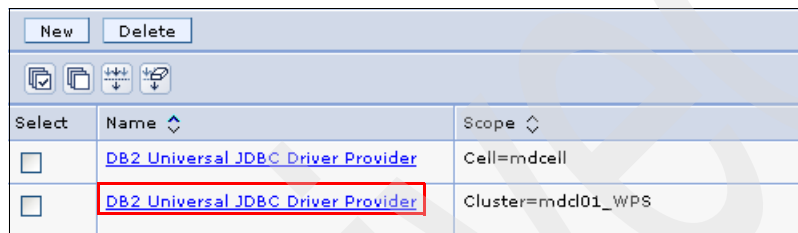
6. Click **Save** to save in the master configuration
7. Click **OK** after the nodes are synchronized.

**Tip:** Navigate to **System administration** → **Console Preferences** and ensure the Synchronize changes with nodes option is selected. This option will ensure all the configuration changes made in the deployment manager are synchronized across all the nodes.

### 3.4.3 Create and configure the data sources

Data sources are resources that include the source of data as well as how to connect to the databases. WebSphere Business Services Fabric for z/OS requires three datasources for the fabricdb and one datasource for the MEDB. This section describes the steps to create the required datasources

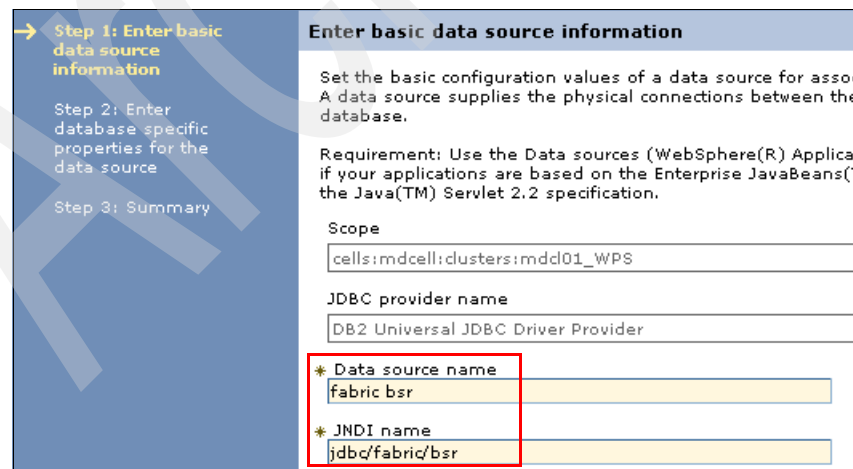
1. In the Integrated Solutions Console, navigate to **Resources** → **JDBC** → **JDBC Providers**. Click the **DB2 Universal JDBC Driver Provider** that is defined at the cluster level. For our single-cluster topology this was the one with the scope equal to Cluster=mdcl01\_WPS (Figure 3-2). For our two-clusters topology we chose Cluster=mpcl02.AppTarget.



<input type="button" value="New"/> <input type="button" value="Delete"/>		
Select	Name	Scope
<input type="checkbox"/>	<a href="#">DB2 Universal JDBC Driver Provider</a>	Cell=mdcell
<input type="checkbox"/>	<a href="#">DB2 Universal JDBC Driver Provider</a>	Cluster=mdcl01_WPS

Figure 3-2 Click the DB2 Universal JDBC Driver Provider defined at cluster scope

2. Click **Data Sources**.
3. Click **New**.
4. Specify Data Source name as **fabric bsr**.  
Specify JNDI name as **jdbc/fabric/bsr** as shown in Figure 3-3.



→ Step 1: Enter basic data source information

Step 2: Enter database specific properties for the data source

Step 3: Summary

**Enter basic data source information**

Set the basic configuration values of a data source for association. A data source supplies the physical connections between the application and the database.

Requirement: Use the Data sources (WebSphere(R) Application Server) if your applications are based on the Enterprise JavaBeans(TM) and the Java(TM) Servlet 2.2 specification.

Scope  
cells:mdcell:clusters:mdcl01\_WPS

JDBC provider name  
DB2 Universal JDBC Driver Provider

\* Data source name  
fabric bsr

\* JNDI name  
jdbc/fabric/bsr

Figure 3-3 Set the data source name and JNDI name

5. Select mddmnode/Fabric\_DB\_Auth\_Alias in the Component-managed authentication alias (Figure 3-4) and click **Next**.

**Component-managed authentication alias and XA recovery authentication alias**

Select a component-managed authentication alias. The selected authentication alias will also be set as the XA recovery authentication alias if your JDBC Provider supports XA. If you choose to [create a new J2C authentication alias](#), the wizard will be canceled.

mddmnode/Fabric\_DB\_Auth\_Alias

Figure 3-4 Set the Fabric database alias

**Coding Server name and Port numbers:** For a type 2 datasource connecting to DB2 z/OS you still have to code the Server name and Port numbers of the DB2 z/OS Distributed Data Facility server (DDF) even though these are normally required by a type 4 datasource. These values do not have to be correct but they cannot be blank. We coded the correct values for our DB2 DDF server to act as documentation.

Find the host name and port that DDF is listening on by looking at the initialization messages of the DB2 MSTR address space. On our system we saw these messages in the joblog of our D9F1MSTR address space:

```
DSNL004I  -D9F1 DDF START COMPLETE  768
          LOCATION  DB9F
          LU        USIBMSC.SCPD9F1
          GENERICLU -NONE
          DOMAIN    wtsc42.itso.ibm.com
          TCPPORT    37893
          SECPORT    0
          RESPORT    37894
          IPNAME     -NONE
```

6. Specify the following information as shown in Figure 3-5 on page 250 and click **Next**.
  - a. Set Database name to DB9F.

Note that this is the Location name of the DB2 z/OS system, not the Fabric database name within DB2 z/OS.
  - b. Set Driver type to 2.
  - c. Set Server name to wtsc42.itso.ibm.com.
  - d. Set Port number as 37893.
  - e. Clear the Use this data source in container managed persistence (CMP) check box.

Step 1: Enter basic data source information

→ Step 2: Enter database specific properties for the data source

Step 3: Summary

### Enter database specific properties for the data source

Set these database-specific properties, which are required by the database vendor JDBC driver to support the connections that are managed through this data source.

\* Database name  
DB9F

\* Driver type  
2

Server name  
wtsc42.itso.ibm.com

Port number  
37893

☐ Use this data source in container managed persistence (CMP)

Previous Next Cancel

Figure 3-5 Fabric Datasource creation Step 2

- f. Click **Finish** and then click **Save the configuration changes**.
- g. Click **OK** after the nodes are synchronized.
7. Set some custom Custom properties as follows.
  - a. Within the new fabric data source, click **Custom properties** and then click the property fullyMaterializeLobData, which has the default value of true. Set the value to false and click **OK**. Do not save the changes yet.
  - b. **Important:** Do not do perform this step for the **fabric me** datasource.  
 Scroll down the Custom properties list and set the schema name you have used for the Fabric tables in either the currentSchema or currentSQLID properties, depending on how you decided to manage your schemas. For our MD cell the schema was MDWPS. Do not make this change for the fabric me data source.
  - c. Click **OK** and then click **Save the configuration changes**.
8. Create the three other data sources using the values specified in the last three rows of Table 3-3 on page 251.  
 For all the data sources, be sure to set the Database Name field to the Location name of the DB2 z/OS system. For our system we set these as follows:
  - Database name: DB9F
  - Server name: wtsc42.itso.ibm.com
  - Port number: 37893

Table 3-3 Data Sources

Data source name	JNDI name	Database name (Location name)	Usage
fabric bsr	jdbc/fabric/bsr	DB9F	Used by Fabric Business Service Repository
fabric gm	jdbc/fabric/gm	DB9F	Used by Fabric Governance Manager
fabric pm	jdbc/fabric/pm	DB9F	Used by Fabric Performance Manager
fabric me	jdbc/fabric/me	DB9F	Used by Fabric Messaging Engine

**Note:** It is normal that Test Connection will fail, as shown in Figure 3-6. Test Connection does not work on a Type 2 datasource when it is defined at cluster level.

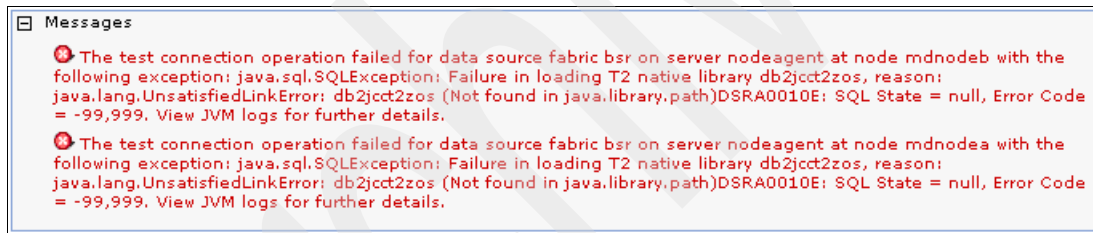


Figure 3-6 Test Connection Error messages for Type 2 datasources

When you have finished, there are four fabric data sources as shown in Figure 3-7 on page 252.

<a href="#">fabric_bsr</a>	jdbc/fabric/bsr	Cluster=mdcl01_WPS
<a href="#">fabric_gm</a>	jdbc/fabric/gm	Cluster=mdcl01_WPS
<a href="#">fabric_me</a>	jdbc/fabric/me	Cluster=mdcl01_WPS
<a href="#">fabric_pm</a>	jdbc/fabric/pm	Cluster=mdcl01_WPS

Figure 3-7 Four fabric datasources listed

### 3.4.4 Create and configure the service integration bus

A service integration bus supports applications using message-based and service-oriented architectures. A bus is a group of one or more interconnected servers. Each server is a member of the bus. Applications connect to a bus at one of the messaging engines associated with its bus members. WebSphere Business Services Fabric uses a service integration bus called **fabricbus**. This section provides the steps to create and configure the fabricbus.

When configuring WebSphere Business Services Fabric in a cluster, the bus needs to use a shared data store. Therefore, we configured fabricbus to use a DB2 z/OS data store like the other WebSphere Process Server buses.

1. In the Integration Solutions Console, navigate to **Service integration** → **Buses** and click **New**.
2. Specify the name of the new bus as fabricbus, as shown in Figure 3-8 on page 253. The WebSphere Business Services Fabric Information Center recommends configuring the buses with security disabled. We decided to enable security because this is nearly always a requirement in a z/OS installation. We checked the Bus security check box.

→ **Step 1: Create a new bus**

Step 2: Confirm create of new bus

**Create a new bus**

Configure the attributes of your new bus

\* Enter the name for your new bus.  
fabricbus

Bus security  
☒

Next Cancel

Figure 3-8 Fabric bus creation Step 1

3. Click **Next**.
4. Click **Finish** and click **Save to the master configuration**.
5. Click **OK** after the synchronization of the nodes.
6. Navigate to **Service integration** → **Buses** → **fabricbus** and click the Local Topology tab. Expand the tree and click **Add** under **Bus Members** (Figure 3-9).

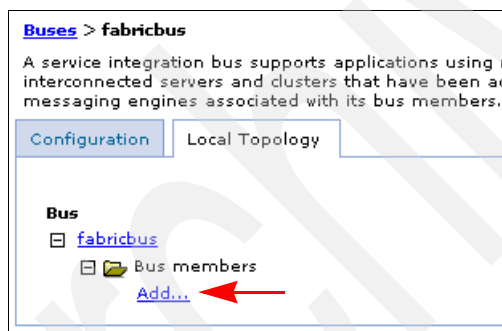


Figure 3-9 Expand the topology and Add a bus member

7. In the Step 1: Select server, cluster or WebSphere MQ server window (Figure 3-10), click the button for **Cluster**, select the cluster, and click **Next**.
- For our single-cluster topology we chose Cluster=mdcl01\_WPS.
  - For our two-clusters topology we chose Cluster=mpcl02.AppTarget because we want to run the Fabric message engine in the same cluster as Fabric.

Add a server, cluster or a WebSphere MQ server as a new member of the bus.

→ **Step 1: Select server, cluster or WebSphere MQ server**

Step 2: Confirm the addition of a new bus member

**Select server, cluster or WebSphere MQ server**

Choose the server, cluster or WebSphere MQ server to add to the bus

☐ Server  
mdnodeb:mdsr01b\_WPS

☒ **Cluster**  
mdcl01\_WPS

☐ WebSphere MQ server  
(none)

Next Cancel

Figure 3-10 Map the bus to the cluster

8. In the Step 2: Select the type of message store window (Figure 3-11), click **Data Store** and then click **Next**.

Add a server, cluster or a WebSphere MQ server as a new member of the bus.

Step 1: Select server, cluster or WebSphere MQ server

→ **Step 2: Select the type of message store**

Step 3: Provide the message store properties

Step 4: Confirm the addition of a new bus member

**Select the type of message store**

Choose the type of message store for the persistence of message state

☐ File store

☒ **Data store**

Previous Next Cancel

Figure 3-11 Choose Data Store



9. In the Step 3: Provide the message store properties window (Figure 3-12), set the following fields:
  - a. Set the Data Source JNDI name to jdbc/fabric/me. (You created this data source earlier.)
  - b. Set the Schema Name to MD01F.

This follows the convention used for the other bus data stores. The “F” in the Schema Name stands for “Fabric.”
  - c. Select Fabric\_DB\_Auth\_Alias from the Authentication alias list box.
  - d. Clear the Create tables check box, because you have already created the tables. Click **Next**.

Step 1: Select server, cluster or WebSphere MQ server

Step 2: Select the type of message store

→ Step 3: Provide the message store properties

Step 4: Confirm the addition of a new bus member

**Provide the message store properties**

Select properties for the data store

\* Data source JNDI name  
jdbc/fabric/me

Schema name  
MD01F

Authentication alias  
mpdmnode/Fabric\_DB\_Auth\_Alias

☐ Create tables

Previous Next Cancel

Figure 3-12 Set data source properties

10. Review the Summary in the Step 4: Confirm the addition of a new bus member window (Figure 3-13), and click **Finish**.

Step 1: Select server, cluster or WebSphere MQ server

Step 2: Select the type of message store

Step 3: Provide the message store properties

→ Step 4: Confirm the addition of a new bus member

**Confirm the addition of a new bus member**

The following is a summary of your selections. To complete the bus member creation, click Finish. If there are settings you wish to change, click Previous to review bus member settings.

Summary of actions:

New bus member "Cluster=mdd01\_WPS" will be added to bus "fabricbus" using a data store as the message store.

Previous Finish Cancel

Figure 3-13 Review the summary and then click Finish

- 11. Click **Save to the master configuration**.
- 12. Click **OK** after the synchronization of the nodes.

3.4.5 Create destinations in the service integration bus

A bus destination is a virtual location within a service integration bus, to which applications attach as producers, consumers, or both, to exchange messages. This section defines how to create a topic space and a queue in the fabricbus Service Integration Bus.

Creating a topic space

Perform the following steps to create a topic space.

- 1. Navigate to **Service integration** → **Buses** → **fabricbus**.
- 2. Under Destination resources, click **Destinations** and then click **New** (Figure 3-14).



Figure 3-14 Create a new destination

- 3. Select the Topic Space radio button as your destination type and click **Next** (Figure 3-15).

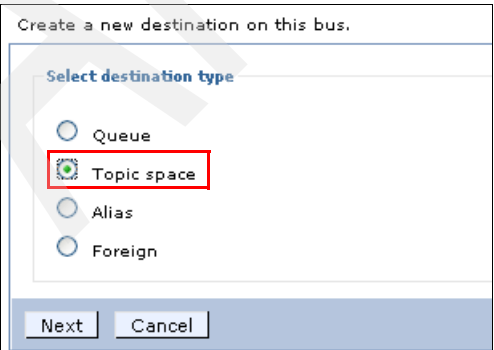


Figure 3-15 Choose Topic Space

4. In the Step 1: Set topic space Attributes window (Figure 3-16), enter **DA.Event.Topic** in the Identifier field, and click **Next**.

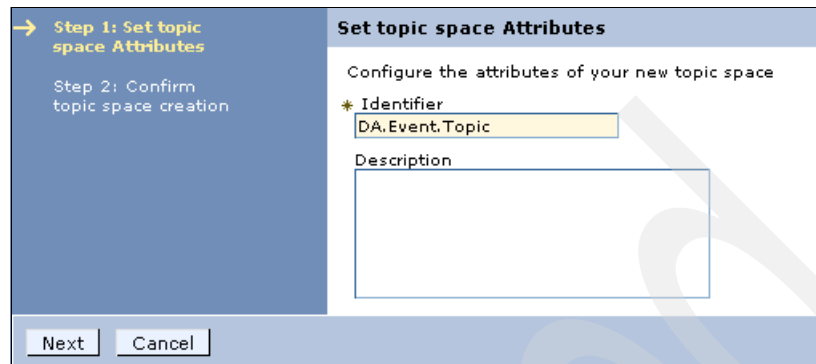


Figure 3-16 Set the Topic Space attributes

5. In the Step 2: Confirm topic space creation window, review the Summary, and click **Finish**.
6. Click **Save to the master configuration**.
7. Click **OK** after the synchronization of nodes.

## Creating a queue

Perform the following steps to create a queue.

1. In the Integrated Solutions Console, navigate to **Service integration** → **Buses** → **fabricbus**.
2. Under Destination resources, click **Destinations** and then click **New**.
3. Select the Queue radio button in the Select destination type panel, as shown in the Figure 3-17, and click **Next**.

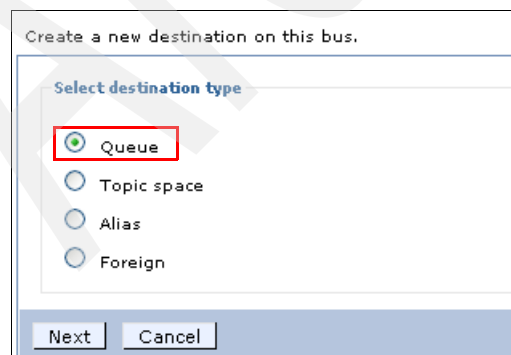


Figure 3-17 Select the Queue destination type

4. Enter Hub.Request.Queue in the Identifier field as shown in the Figure 3-18 and click **Next**.

Create a new queue for point-to-point messaging.

→ **Step 1: Set queue attributes**

Step 2: Assign the queue to a bus member

Step 3: Confirm queue creation

**Set queue attributes**

Configure the attributes of your new queue

\* Identifier  
Hub.Request.Queue

Description

Next Cancel

Figure 3-18 Queue creation for the fabric bus

5. In the Step 2: Assign the queue to a bus member window (Figure 3-19), choose the appropriate Bus member for your topology:

For our single-cluster topology, there was only one Bus member to choose, so we selected Cluster=mdcl01\_WPS.

For our two-clusters topology we chose Cluster=mpcl02.AppTarget because the fabricbus was defined in the AppTarget cluster, local to Fabric.

Create a new queue for point-to-point messaging.

Step 1: Set queue attributes

→ **Step 2: Assign the queue to a bus member**

Step 3: Confirm queue creation

**Assign the queue to a bus member**

Assign the queue to a bus member that

Bus member  
Cluster=mdcl01\_WPS

Previous Next Cancel

Figure 3-19 Assign the queue to a bus member

6. Click **Next**.
7. In the Step 3: Confirm queue creation window, review the Summary, and click **Finish**.
8. Click **Save to the master configuration**.
9. Click **OK** after the synchronization of the nodes.

### 3.4.6 Configure the JMS provider

A Java Messaging Service (JMS) provider enables messaging based on the Java Messaging Service (JMS). It provides J2EE connection factories to create connections for JMS destinations. This section describes the JMS resources you need to define. It contains the following sections:

- ▶ “Create the DAEventConnectionFactory connection factory” on page 259
- ▶ “Create the HubRequestQueue queue” on page 261
- ▶ “Create the DA.Event.Topic topic” on page 262
- ▶ “Create the activation specifications” on page 262

#### Create the DAEventConnectionFactory connection factory

Perform the following steps to create the DAEventConnectionFactory connection factory.

1. Navigate to **Resources** → **JMS** → **JMS Providers**.

In the Scope selection box, select the cluster that will host the Connection Factory.

- For our single-cluster topology that was Cluster= mdc101\_WPS.
- For our two-clusters topology we chose Cluster=mpcl02.AppTarget.

2. Click **Default Messaging Provider** (Figure 3-20).

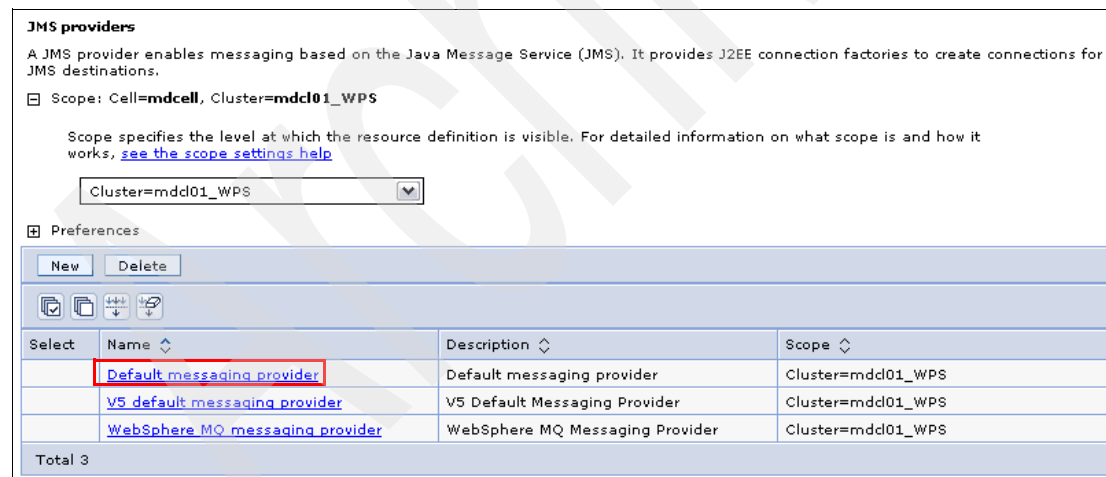


Figure 3-20 Choose the Default Messaging Provider in the cluster's scope

3. Click **Connection factories** and click **New**.
4. Specify the following properties in the General Properties panel (Figure 3-21):
  - Name: DAEventConnectionFactory

- JNDI Name: jms/fabric/DAEventConnectionFactory
- Bus Name: fabricbus
- Target Type: Bus member name

**General Properties**

**Administration**

Scope  
Cluster=mdcl01\_WPS

Provider  
Default messaging provider

\* Name  
DA EventConnectionFactory

\* JNDI name  
jms/fabric/DAEventConnectio

Description

Category

**Connection**

\* Bus name  
fabricbus

Target

Target type  
Bus member name

Figure 3-21 Connection Factory General Properties

5. Set the Component-managed authentication alias to the one created earlier for authentication to the fabricbus (Figure 3-22). For our MD cell this was mddmnode/Fabric\_ME\_Auth\_Alias.

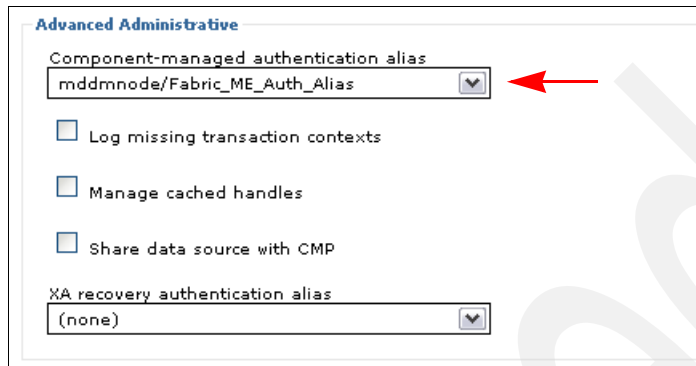


Figure 3-22 Set the Connection factory Component-managed authentication alias

6. Click **OK** and click **Save to the master configuration**.
7. Click **OK** after the synchronization of nodes.

## Create the HubRequestQueue queue

Perform the following steps to Create the HubRequestQueue queue.

1. Navigate to **Resources** → **JMS** → **JMS Providers**.  
In the Scope selection box, select the cluster.
  - For our single cluster topology we chose Cluster= mdcl01\_WPS.
  - For our two-clusters topology we chose Cluster=mpcl02.AppTarget.
2. Click **Default Messaging Provider**.
3. Click **Queues** under Additional Properties and click **New**.
4. Specify the following properties in the Queues creation panel:
  - Name: HUBRequestQueue
  - JNDI Name: jms/fabric/HubRequestQueue
  - Bus Name: fabricbus
  - Queue Name: Hub.Request.Queue
  - Delivery mode: Application
5. Click **OK** and click **Save to the master configuration**.
6. Click **OK** after the synchronization of nodes.

## Create the DA.Event.Topic topic

Perform the following steps to Create the DA.Event.Topic topic.

1. Navigate to **Resources** → **JMS** → **JMS Providers**.

In the Scope selection box select the cluster.

- For our single cluster topology we chose Cluster= mdcl01\_WPS.
- For our two-clusters topology we chose Cluster=mpcl02.AppTarget.

2. Click **Default Messaging Provider**.
3. Click **Topics** under Additional Properties and click **New**.
4. Specify the following properties in Queues creation panel:
  - Name: DA Event Topic
  - JNDI Name: jms/fabric/jms/fabric/HubRequestQueue
  - Bus Name: fabricbus
  - Topic space: DA.Event.Topic
  - Delivery mode: Application
  - Read Ahead: Inherit from connection factory
5. Click **OK** and click **Save to the master configuration**.
6. Click **OK** after the synchronization of nodes.

## Create the activation specifications

Three activation specifications are required:

- ▶ DAPerfMon Activation
- ▶ Hub Request Activation
- ▶ Hub Event Activation

Perform the following steps to create the activation specifications.

1. Navigate to **Resources** → **JMS** → **JMS Providers**.

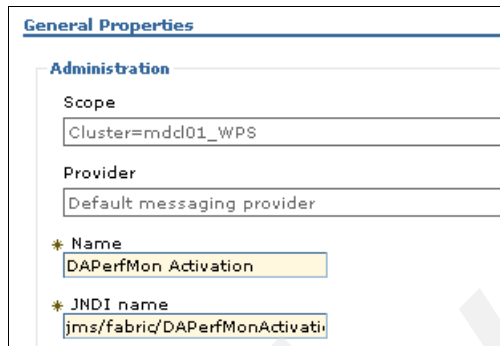
In the Scope selection box select the cluster.

- For our single cluster topology we chose Cluster= mdcl01\_WPS.
- For our two-clusters topology we chose Cluster=mpcl02.AppTarget.

2. Click **Default Messaging Provider**.
3. Click **Activation specifications** and click **New**.



4. Specify the following properties.
  - a. In the General Properties box (Figure 3-23) of the Activation specification creation panel, enter the following values:
    - Name: DAPerfMon Activation
    - JNDI Name: jms/fabric/DAPerfMonActivation



General Properties	
<b>Administration</b>	
Scope	Cluster=mdcd01_WPS
Provider	Default messaging provider
* Name	DAPerfMon Activation
* JNDI name	jms/fabric/DAPerfMonActivation

Figure 3-23 Activation specification General Properties

b. In the Destination box (Figure 3-24) of the Activation specification creation panel, enter the following values:

- Destination Type: Topic
- Destination JNDI Name: jms/fabric/DAEventTopic
- Bus Name: fabricbus
- Acknowledge mode: Auto acknowledge

The screenshot shows the 'Destination' tab of a configuration panel. It contains several fields and dropdown menus: 'Destination type' is set to 'Topic'; 'Destination JNDI name' is 'jms/fabric/DAEventTopic'; 'Message selector' is empty; 'Bus name' is 'fabricbus'; 'Acknowledge mode' is 'Auto-acknowledge'; 'Target' is empty; 'Target type' is 'Bus member name'; 'Target significance' is 'Preferred'; and 'Target inbound transport chain' is empty.

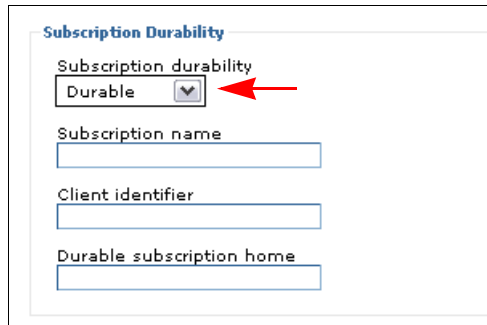
Figure 3-24 Activation spec Destination properties

c. In the Additional box (Figure 3-25) of the Activation specification creation panel, select the JMS Authentication alias from the Authentication alias list box. For our cell this was mddmnode/Fabric\_ME\_Auth\_Alias.

The screenshot shows the 'Additional' tab of a configuration panel. It contains three fields: 'Authentication alias' is set to 'mddmnode/Fabric\_ME\_Auth\_Alias' and is highlighted with a red rectangle; 'Maximum batch size' is empty; and 'Maximum concurrent endpoints' is empty.

Figure 3-25 Activation spec Additional properties

d. In the Subscription durability box (Figure 3-26 on page 265) of the Activation specification creation panel, enter **Durable**.



**Subscription Durability**

Subscription durability  
Durable ▼

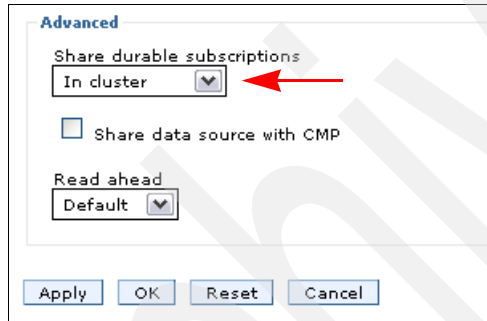
Subscription name  
[ ]

Client identifier  
[ ]

Durable subscription home  
[ ]

Figure 3-26 Activation spec Subscription Durability

- e. In the Advanced box (Figure 3-27) of the Activation specification creation panel, enter **In Cluster** in the Share durable subscriptions list box.



**Advanced**

Share durable subscriptions  
In cluster ▼

☐ Share data source with CMP

Read ahead  
Default ▼

Apply OK Reset Cancel

Figure 3-27 Activation spec Advanced properties

5. Click **OK** and click **Save to the master configuration**.
6. Click **OK** after the synchronization of nodes.
7. Repeat the above steps to create a second activation specification, specifying the following information in the activation specification creation panel:
  - Name: Hub Request Activation
  - JNDI Name: jms/fabric/HubRequestActivation
  - Destination Type: Queue (This one is a Queue.)
  - Destination JNDI Name: jms/fabric/HubRequestQueue
  - Bus Name: fabricbus
  - Acknowledge mode: Auto acknowledge
  - Authentication alias: mddmnode/Fabric\_ME\_Auth\_Alias
  - Subscription durability: Durable
  - Share durable subscriptions: In Cluster

8. Repeat the above steps to create a third activation specification, specifying the following information in the Activation specification creation panel
  - Name: Hub Event Activation
  - JNDI Name: jms/fabric/HubEventActivation
  - Destination Type: Topic (This one is a Topic.)
  - Destination JNDI Name: jms/fabric/DAEventTopic
  - Bus Name: fabricbus
  - Acknowledge mode: Auto acknowledge
  - Authentication alias: mddmnode/Fabric\_ME\_Auth\_Alias
  - Subscription durability: Non Durable. (Careful! This one is Non Durable.)
  - Share durable subscriptions: In Cluster
9. Save all changes.

### 3.4.7 Configure a mail provider

This section describes the steps to configure mail resources required for WebSphere Business Services Fabric. If you do not configure a Mail provider it will not be possible to deploy the fabric-web-apps.ear later.

1. Navigate to **Resources** → **Mail** → **Mail Providers**.

In the Scope selection box select the cluster.

- For our single cluster topology we chose Cluster= mdcl01\_WPS.
- For our two-clusters topology we chose Cluster=mpcl02.AppTarget.

2. Click **Built-in Mail Provider** (Figure 3-28).

**Mail Providers**

Use this page to create a mail provider, an object that encapsulates the protocol providers that you use to access the three default protocol providers: SMTP, IMAP, and POP3. This page also allows you to delete a mail provider.

☐ Scope: Cell=mdcell, Cluster=mdcl01\_WPS

Scope specifies the level at which the resource definition is visible. For detailed information on how to use scope, [see the scope settings help](#).

Cluster=mdcl01\_WPS

☐ Preferences

New Delete

Select	Name	Scope
<input checked="" type="checkbox"/>	Built-in Mail Provider	Cluster=mdcl01_WPS

Total 1

Figure 3-28 Click the Built-in Mail Provider at the cluster scope

3. Click **Mail Sessions** under Additional properties (Figure 3-29).



Figure 3-29 Click Mail sessions

4. Click **New**.
5. Specify the following properties in the Mail Session creation panel (Figure 3-30).
  - Name: Fabric Mail
  - JNDI Name: mail/fabric
  - Mail transport protocol: smtp

A screenshot of a web interface titled 'General Properties'. It contains four fields, each with a yellow star icon to its left: 'Scope' with the value 'cells:mdcell:clusters:mdcd01\_WPS', 'Provider' with the value 'Built-in Mail Provider', 'Name' with the value 'Fabric Mail', and 'JNDI name' with the value 'mail/fabric'.

Figure 3-30 Name and JNDI name in the Mail provider General Properties

**Note:** Mail transport host, Mail store user id and Mail store password are according to your email environment setup. You can also configure mail provider later in the setup.

6. Click **OK** and click **Save to the master configuration**.
7. Click **OK** after the synchronization of nodes.

### 3.4.8 Configure security

There are many options to secure the WebSphere Business Services Fabric environment including federated repositories, local operating system, and stand-alone LDAP. We configured security using a LocalOS user registry, RACF.

For instructions on configuring security using LDAP see Redbooks publication, *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665, which describes the use of LDAP with WebSphere Process Server on non-z/OS platforms. Although this Redbooks publication describes the use of RACF with WebSphere Process Server for z/OS, there are also some comments about using LDAP on z/OS in Section 2.2.6, “Planning security” on page 38.

### Enable service integration bus security

When messaging security is switched on, all users who connect to a bus must have the required authorization permissions to use the bus resources. The user accessing the bus has a bus connector role.

Because a z/OS installation of WebSphere Process Server will almost certainly require bus security to be enabled, this chapter has shown how to enable the fabric bus with security enabled. The other buses used by WebSphere Process Server also have security enabled.

The following procedure can be used to review security settings to check that you have everything set-up correctly.

1. Ensure that you create a new J2C Authentication Alias for a user ID that is going to have bus connector role. See Section 3.4.2, “Create an authentication alias for Fabric” on page 247.
2. Navigate to **Service Integration** → **Buses** → **fabricbus**, and click **Security**. The general properties panel appears (Figure 3-31 on page 269).
  - a. Ensure the Enable bus security check box selected.
  - b. In the Inter-engine authentication alias and Mediations authentication alias fields, select the J2C authentication alias you will use for authenticating to the fabricbus.

For our cell this was the Fabric\_ME\_Auth\_Alias alias.

Figure 3-31 Set the fabricbus authentication aliases

3. Click **OK** and save your changes.
4. Navigate to **Service Integration** → **Buses** → **fabricbus** → **Security**. The General Properties panel will appear (Figure 3-32). In the Bus Connector Role section, click **Users and groups**. Click **New**.
5. Choose the User name radio button and specify the user ID that is associated with the J2C Authentication alias you created for authenticating to the fabricbus. For our cell the alias was Fabric\_ME\_Auth\_Alias and the user ID was MDJMSU. Click **OK**.

Figure 3-32 Specify the user or group with the Bus Connector Role

**A different approach:** Click **Group name** and specify a group to which MDJMSU user is connected. For example, our RACF user ID MDJMSU is in default group MDJMSG so we could have specified the group MDJMSG.

6. Check that the Connection Factory is using the JMS J2C Authentication Alias by performing the following steps.
  - a. Navigate to **Resources** → **JMS** → **JMS Providers**. In the Scope selection box select Cluster= mdcl01\_WPS.
  - b. Click **Default Messaging Provider** and click **Connection factories**.
  - c. Click **DAEventConnectionFactory**.
  - d. On the Component-managed authentication alias field, select the J2C Authentication alias you created earlier for authenticating to the fabricbus. For our MD cell this was mddmnode/Fabric\_ME\_Auth\_Alias.
  - e. Click **OK** and save the configuration changes.
7. Check that the Activation specifications are using the JMS J2C Authentication Alias by performing the following steps:
  - a. Navigate to **Resources** → **JMS** → **JMS Providers**.  
In the Scope selection box select the cluster.
    - For our single cluster topology we chose Cluster= mdcl01\_WPS.
    - For our two-clusters topology we chose Cluster=mpcl02.AppTarget.
  - b. Click **Default Messaging Provider** and click **Activation specifications**.
  - c. Click **DAPerfMon Activation**.
  - d. Check that the Component-managed authentication alias field specifies the J2C authentication alias that was created for connecting to the bus. For our MD cell this was mddmnode/Fabric\_ME\_Auth\_Alias.
  - e. Click **OK** and save the configuration changes.
8. Repeat the step 7 for the following Activation specifications:
  - Hub Event Activation
  - Hub Request Activation



### 3.4.9 Configure distributed cache

This section contains the following sections:

- ▶ “Define a Replication domain” on page 271
- ▶ “Create Object cache instance WBSF Context Cache” on page 272

#### Define a Replication domain

The dynamic cache of WebSphere Business Services Fabric is replicated using Data Replication Service to all the cluster members. Follow the steps below to create the replication domain.

1. In the Integrated Solutions Console, navigate to **Environment** → **Replication domains**, and click **New**.
2. In the General Properties section (Figure 3-33), enter WBSF DA Replication in the Name field.
3. In the **Number of replicas** section, select the Entire Domain radio button.

**Replication domains > New**

Use this page to configure the replication properties that

Configuration

**General Properties**

\* Name  
WBSF DA Replication

\* Request timeout  
5 seconds

**Encryption**

☐ Encryption type  
none

Regenerate encryption key

**Number of replicas**

☐ Single replica

☒ Entire Domain

☐ Specify  
Number of replicas

Apply OK Reset Cancel

Figure 3-33 Creation of Replication Domain for Fabric

4. Click **OK** and then save the configuration changes.

## Create Object cache instance WBSF Context Cache

An object cache instance is a location, in addition to the default shared dynamic cache, where J2EE applications can store, distribute, and share data. Follow the steps below to create an object cache instance.

1. Navigate to **Resources** → **Cache Instances** → **Object cache instances**.

In the Scope selection box, select the cluster.

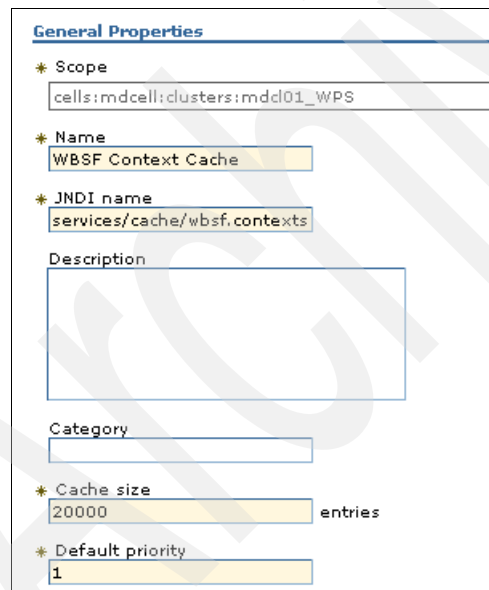
- For our single cluster topology we chose Cluster= mdcl01\_WPS.
- For our two-clusters topology we chose Cluster=mpcl02.AppTarget.

2. Click **New**.

3. Specify the following properties. in the Object cache instances creation panel.

- a. In the General Properties box (Figure 3-34) of the Object cache instances creation panel, enter the following values:

- Name: WBSF Context Cache
- JNDI Name: services/cache/wbsf.contexts
- Cache size: 20000

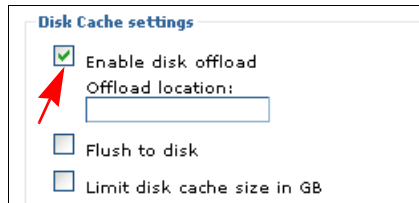


The screenshot shows the 'General Properties' dialog box for creating an Object cache instance. The fields are as follows:

Property	Value
Scope	cells:mdcell:clusters:mdcl01_WPS
Name	WBSF Context Cache
JNDI name	services/cache/wbsf.contexts
Description	
Category	
Cache size	20000 entries
Default priority	1

Figure 3-34 Object cache General Properties

- b. In the Disk Cache Settings box (Figure 3-35) of the Object cache instances creation panel, select the Enable disk offload check box.



The 'Disk Cache settings' dialog box contains the following elements:

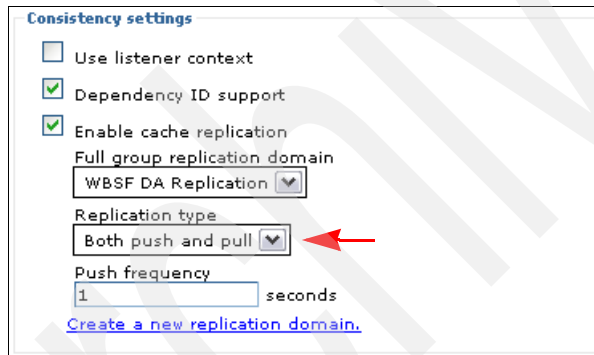
- ☒ Enable disk offload
- Offload location:
- ☐ Flush to disk
- ☐ Limit disk cache size in GB

A red arrow points to the 'Enable disk offload' checkbox.

Figure 3-35 Check Enable disk offload

- c. In the Consistency settings box (Figure 3-36) of the Object cache instances creation panel, enter the following values:

- Check: Enable cache replication
- Full group Replication domain: WBSF DA Replication
- Replication type: Both Push and Pull
- Push frequency: 0



The 'Consistency settings' dialog box contains the following elements:

- ☐ Use listener context
- ☒ Dependency ID support
- ☒ Enable cache replication
- Full group replication domain:
- Replication type:
- Push frequency:  seconds
- [Create a new replication domain.](#)

A red arrow points to the 'Replication type' dropdown menu.

Figure 3-36 Enabling cache replication

4. Click **OK**, then save the configuration changes.

### 3.4.10 Configure a namespace variable for CEI

WebSphere Business Services Fabric events are emitted to the JMS destinations present in the cluster running the message engines.

The WebSphere Business Services Fabric engine looks-up destination values in a namespace `wbsf-cbe-emitter-factory`, which references the CBE emitter factory JNDI name. When CEI is configured in a cluster the JNDI name of the CBE emitter factory is not fully qualified and a JNDI lookup error will occur because the look-up defaults to the node level. The error is described in Section 4.7.14, “NMSV0610I: A NamingException for wbsf-cbe-emitter-factory” on page 349.

This procedure describes how to configure a namespace binding to resolve this issue.

1. In the Integrated Solutions Console, navigate to **Servers** → **Clusters** and then select the cluster where Fabric is configured.
  - For our single-cluster topology that was `mdcl01_WPS`.
  - For our two-clusters topology it was `mpcl02.AppTarget`.
2. Expand **Common Event Infrastructure** and click **Common Event Infrastructure Destination**.
3. Note the JNDI Name under Event Infrastructure emitter factory JNDI name. In our scenario it is `/com/ibm/events/configuration/emitter/Default`.
4. Paste this string into Notepad and append `cell/clusters/<cluster_name>` to create the fully-qualified JNDI name. For example:  
**`cell/clusters/mdcl01_WPS/com/ibm/events/configuration/emitter/Default`**
5. Navigate to **Environment** → **Naming** → **Name Space Bindings**.  
In the Scope selection box select the cluster.
  - For our single cluster topology we chose Cluster= `mdcl01_WPS`.
  - For our two-clusters topology we chose Cluster=`mpcl02.AppTarget`.
6. Click **New**.
7. Select **String** in the Binding type and click **Next**.
8. Specify the following values
  - Binding Identifier: `wbsf-cbe-emitter-factory`
  - Name in name space: `wbsf-cbe-emitter-factory`
  - String Value:  
`cell/clusters/mdcl01_WPS/com/ibm/events/configuration/emitter/Default`
9. Click **Finish** and save the configuration change.

### 3.4.11 Install the Fabric core EAR files

WebSphere Business Services Fabric is packaged into four enterprise application EAR files:

- ▶ fabric-engine.ear
- ▶ fabric-catalog.ear
- ▶ fabric-webtools.ear
- ▶ fabric-webtools-help.ear

These are sometimes describes as the core Fabric EAR files. Although the WebSphere Business Services Fabric recommends using a JACL script to deploy these applications we found that the script did not use the correct application names and would only work as provided when the cell contained only one server or one cluster. For this reason we decided to install these EAR files, using the Integrated Solutions Console. In the two-clusters topology the Fabric EAR files are targeted at the AppTarget cluster, which for our MP cell was mpcl02.AppTarget.

**Tip:** The Fabric applications will require a lot more space in your ZFS file systems so they may need to extend during the deployment process. After installing Fabric, our deployment manager's node ZFS occupied 1300CYLs while each node occupied 1100CYLs.

To ensure a clean deployment, ensure that the volumes that host your cell's ZFS datasets have enough free space to allow them to obtain secondary extents and grow to the sizes mentioned above.

1. In the Integrated Solutions Console, navigate to **Applications** → **Install New Application**.
2. Choose **Remote file system** and click **Browse**.
3. Click the deployment manager's node. Navigate up to the root file system and then to the Fabric product's /runtime directory. On our system this was /usr/lpp/webify/V6R1M2/runtime.

4. Select the **fabric-catalog.ear** radio button (Figure 3-37) and click **OK**.

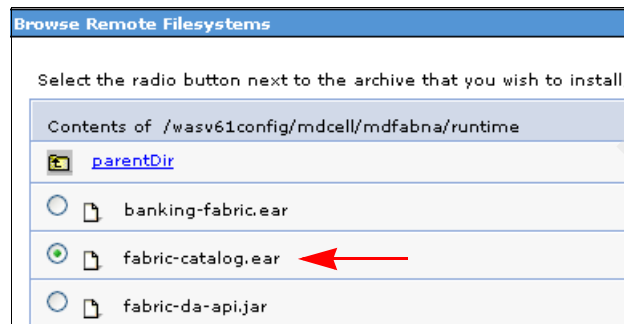


Figure 3-37 Selecting *fabric-catalog.ear*

5. Click **Next** in the Preparing for the application installation window.
6. Click **Next** in the Step 1: Select installation options window.
7. In the Step 2: Map modules to servers window (Figure 3-38), ensure that the modules are shown mapped to the correct cluster and click **Next**.

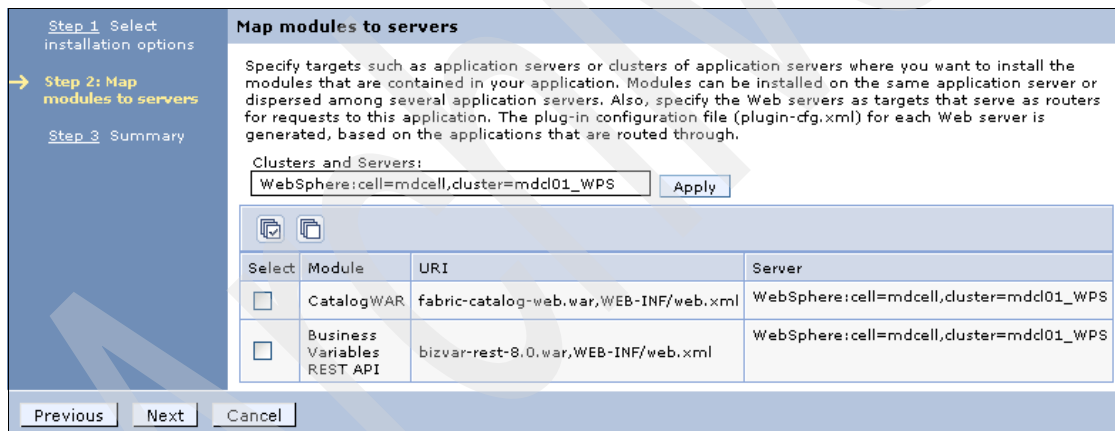


Figure 3-38 Mapping *fabric-catalog.ear* to the *mdcl01\_WPS* cluster

**Note:** When you only have one cluster in the cell, all the modules are mapped to that cluster by default. If you have other clusters in the cell, select all the modules, and under Clusters and Servers, select the cluster in which you wish to configure WebSphere Business Services Fabric, for example, WebSphere:cell=mpcell,cluster=mpcl02.AppTarget. Then click **Apply**.

8. In the Step 3 Summary window, click **Next** and then click **Finish**.
9. Save the configuration changes.
10. Repeat these steps to install the other three applications:
  - fabric-engine.ear
  - fabric-webtools.ear
  - fabric-webtools-help-ear.ear

In a production environment you do not need to install fabric-webtools-help.ear.

**Note:** Make sure all your nodes and servers are running when you are saving the changes and synchronizing the nodes.

11. Do not attempt to start the applications until you have restarted the cluster. To do so, navigate to **System Administration** → **Clusters**. Select the cluster in which you have configured WebSphere Business Services Fabric for z/OS. Stop and restart the cluster.

### 3.4.12 Grant access to the Fabric Tools Console

When using a LocalOS, SAF-based user registry such as RACF with WebSphere Process Server, the role-based authorization checks are normally performed by SAF-based user registry. When using RACF, roles are defined in RACF profiles in class EJBROLE and user ID and groups are permitted access to these profiles. When using a non-SAF user registry, the Integrated Solutions Console is used to permit users and groups to roles.

For example, only users with FabricAdministrator role for the Fabric\_Tool can access the Fabric console. For our cell called MD we created a RACF EJBROLE profile called MD.FabricAdministrator and a RACF group called MDFADMG was permitted access to this profile. We created a user called MDFABADM, which has MDFABADG as its DFLTGRP. Therefore, user ID MDFABADM had the FabricAdministrator role. See the jobs MDRACF (Example 3-2 on page 234) and MDEJBROL (Example 3-3 on page 237) for details of this setup for all the roles.

If you are not using a LocalOS -SAF-based user registry like RACF, then it is necessary to permit groups to roles using the Integrated Solutions Console. For details of how to do this, see Section 3.4.12, “Grant access to the Fabric Tools Console” on page 277.

## 3.5 Verify the Fabric installation and configuration

This section describes the steps to verify the installation and configuration of WebSphere Business Services Fabric.

1. Log in to the Integrated Solutions Console and verify the Fabric applications are started as shown in Figure 3-39.

<input type="checkbox"/>	<a href="#">Fabric Catalog</a>	
<input type="checkbox"/>	<a href="#">Fabric Engine</a>	
<input type="checkbox"/>	<a href="#">Fabric Tools</a>	
<input type="checkbox"/>	<a href="#">Fabric Tools Help</a>	

Figure 3-39 Fabric core applications status

2. In the deployment manager console navigate to **Service integration** → **Buses** → **fabricbus** and click **Messaging engines**. The fabricbus messaging engine will be up and running as shown in Figure 3-40.

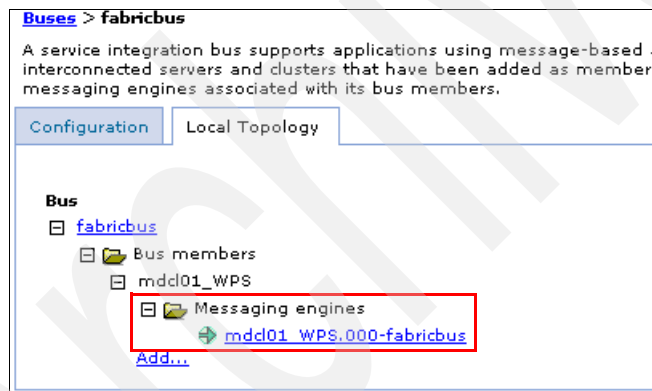


Figure 3-40 fabricbus Messaging Engine Status



3. Type the following url in the browser `http://<XXX>:<ZZZ>/fabric`, where XXX is the IP or host name of any one of the members of mdcl01\_WPS Cluster and ZZZ is the default-host port of the cluster member, for example `http://wtsc42.itso.ibm.com:20048/fabric`.

You will see the login window as shown in Figure 3-41. Log in with the user ID you added to RACF as the Fabric Administrator user ID.

Address <http://wpsplex.itso.ibm.com:20048/fabric/login.jsp> Go Links

## WebSphere Business Services Fabric

Login

Please enter your **User Id** and **Password**.

Fields marked with an asterisk (\*) are required fields.

\* **User Id** MDFABADM

\* **Password** \*\*\*\*\*

Login

© Copyright IBM Corp. 2006-2008. WebSphere Business Services Fabric \$FABRIC\_VERSION\$

Figure 3-41 Log in to the Fabric console

**Note:** When we tried to log in to Fabric for the first time we found that some of the text on the login panel was uninitialized. The text of fields like User ID and Password, for example, did not contain valid data. These fields are initialized from meta-data in the Fabric tables that appears to be inserted into the tables when fabric is started for the first time.

This problem occurs only when you try to log in the first time after starting Fabric for the first time. Simply stopping and restarting the Fabric\_Tools application will resolve the problem. See also Section 4.7.9, “Garbage on Fabric login window at first login” on page 341.

If the user ID does not have access to the required role, you receive the error shown in Figure 3-42.



Figure 3-42 Error received if not authorized

When the WebSphere Process Server cell is using a RACF user registry, go to the z/OS SYSLOG, where you see a related ICH408I error as shown in Example 3-7.

*Example 3-7 RACF ICH408I error when unauthorized to Fabric role(s)*

---

```
ICH408I USER(MDFABADM) GROUP(MDFADMG ) NAME(MD FABRIC ADMIN      )
      MD.FabricStudioUser CL(EJBROLE )
      INSUFFICIENT ACCESS AUTHORITY
      ACCESS INTENT(READ  ) ACCESS ALLOWED(NONE  )
+BB000222I: SECJ0129E: Authorization failed for MDFABADM while
invoking GET on default_host:fabric/app, Authorization failed, Not
granted any of the required roles: FabricAdministrator
FabricGovernanceAdministrator FabricPerformanceUser
FabricSubscriberManager FabricBasicUser FabricStudioUser
```

---

To resolve this error, permit the user ID to one of the EJBROLE profiles. In this case the user ID MDFABADM, which we wanted to be the Fabric Administrator user ID, had not been granted to any of the roles.

4. The first time you log in, you are prompted to register your name and email address (Figure 3-43).



**User Account Creation**

You are logging in for the first time using account information that is authenticated outside of the WebSphere Business Services Fabric.

Provide information in the following required fields in order to complete the creation of your account.

Fields marked with an asterisk (\*) are required fields.

**User Id**      **MDFABADM**

\* **First Name**      MDFabric

\* **Last Name**      Administrator

\* **Email Address**      MDFABADM@ibm.com

The form is titled 'User Account Creation' and contains instructions for first-time login. It includes fields for User Id (MDFABADM), First Name (MDFabric), Last Name (Administrator), and Email Address (MDFABADM@ibm.com). A 'Submit' button and a 'Cancel' button are at the bottom. To the right of the form is a graphic showing a hierarchy of levels: Application Level, Service Level, and a diagram of interconnected nodes.

Figure 3-43 First-time login user account creation

5. After registering, the WebSphere Business Services Fabric welcome panel is displayed as shown in Figure 3-44.



**WebSphere Business Services Fabric**

Welcome, MDFabric • [Logout](#)  
Login Time: 22 Aug 2008 23:12:30

**My Services**

- My Inbox
- Business Services Repository
- Governance Manager
- Performance Manager
- Subscriber Manager
- Help

**My Inbox**

Welcome to the WebSphere™ Business Services Fabric

The WebSphere™ Business Services Fabric is a comprehensive SOA offering to model, assemble, deploy, manage and govern composite business applications. It provides the design-time tooling, run-time environment, industry reference models, and pre-built SOA assets to enable rapid development of loosely coupled composite business applications.

**Inbox Messages**

View  rows at a time

0 rows

From	Message	Date
------	---------	------

Page 1 of 1

© Copyright IBM Corp. 2006-2008.      WebSphere Business Services Fabric \$FABRIC\_VERSION\$.

The screenshot shows the 'WebSphere Business Services Fabric' welcome panel. It includes a 'My Services' sidebar with links like 'My Inbox', 'Business Services Repository', 'Governance Manager', 'Performance Manager', 'Subscriber Manager', and 'Help'. The main content area has a 'My Inbox' section with a welcome message and a description of the fabric. Below this is an 'Inbox Messages' section with a table showing 0 rows. The footer contains copyright information and a version string.

Figure 3-44 WebSphere Business Services Fabric welcome panel

## 3.6 Install and test the sample application

We tested the WebSphere Business Services Fabric environment by using the sample enterprise application described in Redbooks publication *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665.

The enterprise application installed in this section is supplied with the additional materials provided with this book. Refer to Appendix A, “Additional material” on page 351. Perform the following steps to install and test the sample application:

1. Update the endpoint URLs of ITSOBanCBAPrj.  
See page 283.
2. Change endpoint URLs by editing the .owl file.  
See page 283.
3. Import the FCA files.  
See page 284.
4. Assign MDFABADM to the ITSOBanCBAPrj team.  
See page 284.
5. Create an External LDAP federation project.  
See page 286.
6. Set Enrollments.  
See page 291.
7. Install the sample application's EAR Files.  
See page 295.
8. Map modules to the Web server.  
See page 298.
9. Update the SCA Import URLs.  
See page 298.
10. Test the sample application.  
See page 300.

**Note:** When you are using the Fabric console, each operation you select from the menu on the left is displayed as a series of tabbed pages. Each tab usually has its own save button. It is important to click **Save** on each tab. Clicking **Save** on one tab does not save changes made on the other tabs. We found that forgetting to save changes on each tab was the source of a lot of confusing errors.

### 3.6.1 Update the endpoint URLs of ITSOPBankCBAPrj

The sample application provided with the Additional Materials in code\Scenarios\Fabric was originally tested and developed for the non-z/OS part of this Redbooks publication. All the endpoints defined in the application point to host name itsodmgr, using the default HTTP port of 80.

When testing the sample on your z/OS system, you will need to set a different host name and port from the ones hard-coded in the sample application. This update must be made before you import the application into Fabric. There are two ways to update the application Endpoint URLs:

- ▶ Update the file `http_www.ibm.com_vehicleloan_inst.owl` in `ITSOPBankCBAPrj20080804-owl.zip`
- ▶ Update the endpoints using WID

The best way to do this is using WID. The process for updating the application's endpoint URLs using WID is described in Section 3.6.2, "Change endpoint URLs by editing the .owl file" on page 283.

A z/OS systems programmer configuring the ITSO Fabric sample application for the first time is unlikely to have WID on their workstation. Therefore we describe how to edit the .owl file in Section 3.6.2, "Change endpoint URLs by editing the .owl file" on page 283 so you can install the ITSO sample application into your WebSphere Business Services Fabric for z/OS cluster without the need for WID.

### 3.6.2 Change endpoint URLs by editing the .owl file

This section describes how to update the endpoint URLs that are coded within the `http_www.ibm.com_vehicleloan_inst.owl` file, which is in the `ITSOPBankCBAPrj20080804-owl.zip`.

1. Extract the `ITSOPBankCBAPrj20080804-owl.zip` file on your workstation.
2. Open `http_www.ibm.com_vehicleloan_inst.owl` in Textpad or WordPad.
3. Search for String `http://itsodmgr`. Replace the host name `itsodmgr` with your `<host>:<port>` (where `<host>` is the WebSphere Business Services Fabric for z/OS host and `<port>` is the HTTP port). There is no port following the host name `itsodmgr` in the original file because it was defaulting to port 80.
4. Save the .owl file.
5. Compress the updated owl file back into `ITSOPBankCBAPrj20080804-owl.zip`.

**Note:** Make sure the file is directly in the zip file and not added under a folder within the zip file.

### 3.6.3 Import the FCA files

Perform the following steps to import the FCA files.

**Note:** Before you import the FCA files, make sure you have updated the host name:port in the ITSOBankCBAPrj20080804-owl.zip as described in Section 3.6.1, “Update the endpoint URLs of ITSOBankCBAPrj” on page 283.

1. Log in to the WebSphere Business Services Fabric console by using the following URL:  
`http://<hostname>:<port>/fabric/app`  
For example, for our system the URL was as follows:  
`http://wpsplex.itso.ibm.com:20048/fabric/app`
2. Navigate to **Governance Manager** → **Import/Export**.
3. Click **Browse** and locate the code\Scenarios\Fabric\FCA folder in the additional materials supplied with this book. Choose `OrganizationsUsersandRoles20080804-owl.zip`, and click **Import file**.
4. Repeat these steps to install the FCA files in the following order:
  - FabricGovernance20080804-owl.zip
  - ITSOBankOntPrj20080804-owl
  - ITSOBankCBAPrj20080804-owl.zip

### 3.6.4 Assign MDFABADM to the ITSOBankOrg team

Perform the following steps to assign MDFABADM to the ITSOBankOrg team:

1. In the Fabric console, expand **Governance Manager** and click **Manage User Accounts**. The Manage User Accounts window opens (Figure 3-45).

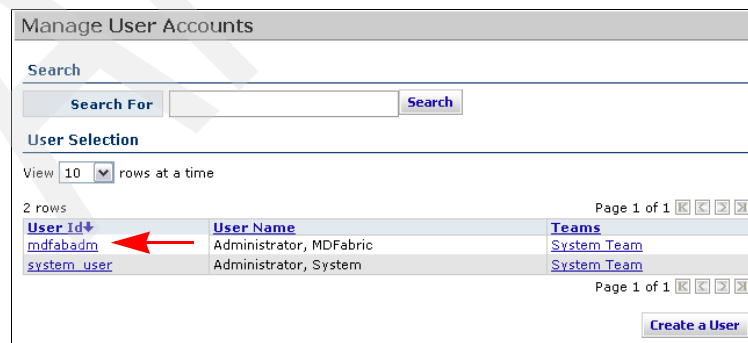


Figure 3-45 Click a user ID to assign to the ITSOBankOrg

2. Click the user ID you want to add to the ITSOBankOrg project. This will probably be the user ID with which you just logged in. In our case this was mpfabadm.
3. Click the team ITSOBankOrg and then click the right-arrow to move the ITSOBankOrg team into the Selected Teams box on the right, as shown in Figure 3-46.

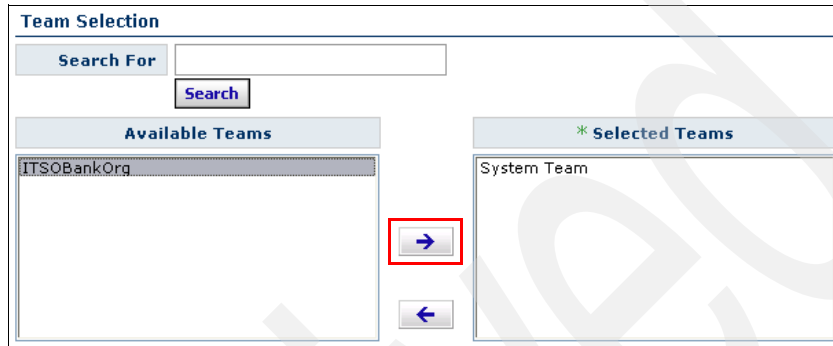


Figure 3-46 Select the ITSOBankOrg team and move to Selected Teams

4. When the ITSOBankOrg team is in the Selected Teams box (Figure 3-47), click **Save**.

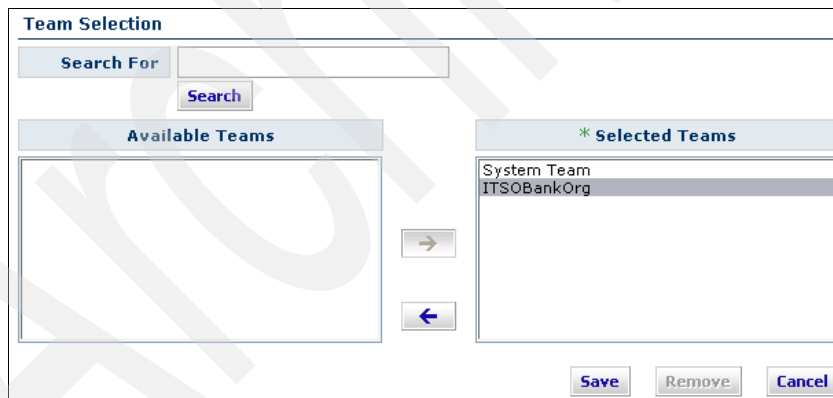


Figure 3-47 Save the Team assignment change

### 3.6.5 Create an External LDAP federation project

When WebSphere Business Services Fabric for z/OS is configured into a WebSphere Process Server for z/OS cell that is using a LocalOS user registry (RACF in our case), authentication to Fabric and verification of Fabric J2EE roles like FabricAdministrator is performed against RACF.

When you are configuring projects inside Fabric (assigning roles, managing enrollments, and managing subscriptions for example), Fabric will need to use some other user registry (such as LDAP).

We configured a Tivoli Directory Server for z/OS V6 LDAP server to use in conjunction with WebSphere Business Services Fabric for z/OS.

You need to configure a new External LDAP federation project that will allow Fabric to work with the external LDAP server.

1. In the Fabric console, navigate to **Governance Manager** → **Configure Projects**.
2. Click **Create Project**.
3. Complete the new project details (Figure 3-48 on page 287) in the Create a Project window as follows.
  - a. In the Project Details section, enter a name for the project (we chose TDSzOS) and a description.
  - b. In the Project Specification section, choose **External LDAP** in the Project Type field.
  - c. In the Source Name text box, enter a name to uniquely identify the LDAP repository. We set the started task name for our LDAP server, MDLDAP.
  - d. Enter a value in the User NSI URI text box. The format of this is different than is stated in the Information Center. The required format is <http url>/dir/NSx# where x is some number. The URI follows web semantics but a real hostname:port is not required. Nevertheless, we entered the following URI:  
`http://wpsplex.itso.ibm.com:20048/TDSzOS/NS1#`
  - e. Click **Create Project**.



**Create a Project**

Fields marked with an asterisk (\*) are required fields.

**Project Details**

\* **Project Name** TDSzOS **Description** TDS zOS LDAP

**Project Specification**

\* **Project Type** External LDAP

**External Details**

\* **Source Name** MDLDAP \* **User NS URI** so.ibm.com:20048/TDSzOS/NS1#

**Project Team**

\* **Team** ITSOBankOrg

**Create Project** **Cancel**

Figure 3-48 Create the external LDAP federation project

- Click **Save** to save the new project (Figure 3-49).

**Configure Projects**

**Project** TDSzOS **Back to Projects**

**Project Details** **Project Team** **Namespaces** **Federation Settings** **Federation Mapping**

Fields marked with an asterisk (\*) are required fields.

**Project Details**

\* **Project Name** TDSzOS **Project Type** External LDAP

**Description** TDS zOS LDAP

**Save** **Remove Project**

Figure 3-49 Save the new project

5. Click the Project Team tab. Notice in Figure 3-50 that user ID mdfabadm is a Team Member because it was assigned to ITSOBankOrg team and the ITSOBankOrg team was assigned to the TDSzOS project.

**Configure Projects**

Project: TDSzOS [Back to Projects](#)

Project Details | **Project Team** | Namespaces | Federation Settings | Federation Mapping

**Info**

- Select a **Team** to view its **Team Members**.
- Click **Save** to associate the Team to the Project.

**Team Selection**

Team: ITSOBankOrg

**Team Members**

View 10 rows at a time

1 rows Page 1 of 1

User Id	Name
mdfabadm	Administrator, MDFabric

Page 1 of 1

[Save](#) [Reset](#)

Figure 3-50 The Project Team showing Team Members

6. Click the Namespaces tab. Notice in Figure 3-51 that the Namespace name is a concatenation of the project name and the string External NS 1.

**Configure Projects**

Project: TDSzOS [Back to Projects](#)

Project Details | Project Team | **Namespaces** | Federation Settings | Federation Mapping

**Owned Namespaces**

1 rows

Namespace	Type	Description
TDSzOS External NS 1	External	

[Create a Namespace](#)

**Imported Namespaces**

0 rows

Namespace	Type	Description	Primary Project	Remove
-----------	------	-------------	-----------------	--------

[Import Namespaces](#) [Remove Imports](#)

Figure 3-51 The Namespaces tab showing the Namespace name

7. Click the Federation Settings tab.
8. Enter the details of the LDAP server as shown in Figure 3-52. When running Tivoli Directory Server for z/OS in a sysplex, the hostname must be a Distributed VIPA.

**Configure Projects**

Project: TDSzOS [Back to Projects](#)

Project Details | Project Team | Namespaces | **Federation Settings** | Federation Mapping

Fields marked with an asterisk (\*) are required fields.

**Connection Details**

\* Hostname: wpsplex.itso.ibm.com    \* Port: 2389  
 \* Username: uid=LDAPADM,o=ibm    \* Password: ••••••

**External Details**

Source Name: MDLDAP    \* User: TDSzOS External NS 1  
 Namespace:   
[Save](#)    [Test Connection](#)

Figure 3-52 Setting LDAP server details on the Federation Settings tab

9. Click **Test Connection**. If this does not succeed as shown in Figure 3-53, recheck the details you entered.

**Configure Projects**

Project: TDSzOS [Back to Projects](#)

Project Details | Project Team | Namespaces | **Federation Settings** | Federation Mapping

Fields marked with an asterisk (\*) are required fields.

**Info**

- The connection test was successful.

Figure 3-53 Test connection was successful

10. Click **Save** to save the changed on the Federation Settings tab once you have tested the connection successfully.

11. Click the Federation Mapping tab. When using Tivoli Directory server, it is necessary to code the BaseDN attributes that apply to the users in the LDAP server. For our LDAP server the users were defined under **cn=People**. The suffix was o=ibm, so we coded BaseDN:cn=People,o=ibm as shown in Figure 3-54.



The screenshot shows the 'Configure Projects' interface. At the top, there's a 'Project' dropdown set to 'TDSzOS' and a 'Back to Projects' button. Below this are several tabs: 'Project Details', 'Project Team', 'Namespaces', 'Federation Settings', and 'Federation Mapping'. The 'Federation Mapping' tab is selected. On the left, there's a 'Mapping Data' section. The main area shows a text input field containing 'BaseDN:cn=People,o=ibm', which is highlighted with a red rectangle. At the bottom right, there is a 'Save' button, also highlighted with a red rectangle.

Figure 3-54 Mapping LDAP attributes on the Federation Mapping tab

12. Click **Save**. The TDSzOS project is now complete.
13. Stop and restart the four fabric applications.
14. During the restart of the applications look in the servant logs for messages like those in Example 3-8, which show a successful federation of the LDAP server. If there are problems with the external LDAP project or with the LDAP server, you will see errors in the logs at this point. Any errors must be resolved before proceeding.

*Example 3-8 Messages showing successful federation of external LDAP server*

---

```
Trace: 2008/09/04 16:00:52.799 01 t=7B1988 c=UNK key=P8 (13007002)
      ThreadId: 0000005b
      FunctionName: internalInfo
      SourceId: com.webify.fabric.catalog.federation.host.FederationHost
      Category: INFO
      ExtendedMessage: Registering federated source
      'ldap(wpsplex.itso.ibm.com:2389)' as 'MPLDAP'
```

---

### 3.6.6 Set Enrollments

The next step is to grant roles, enrollments and subscriptions. See the following relevant sections:

- ▶ “Grant Administrator role within ITSOBankOrg” on page 291
- ▶ “Enroll ITSOBankOrg to ITSOBankLoadApp” on page 294
- ▶ “Subscribe the user ID to ITSOBankLoan App” on page 294

#### Grant Administrator role within ITSOBankOrg

Perform the following steps to Grant Administrator role within ITSOBankOrg.

1. In the Fabric console, navigate to **Subscriber Manager** → **Manage Subscribers**. Click **ITSOBankOrg** → **Users** → **Grant User Roles**.

If Fabric is able to contact the external LDAP Server and search for users you will see all the users Fabric finds in the Available users table on the left as shown in Figure 3-55.

**Note:** If the users in your external LDAP Server are not listed in the Available Users list, check that you stopped and started all the fabric applications after creating the external LDAP server project. Also, go back and check that you clicked **Save** on every tab in external LDAP project.

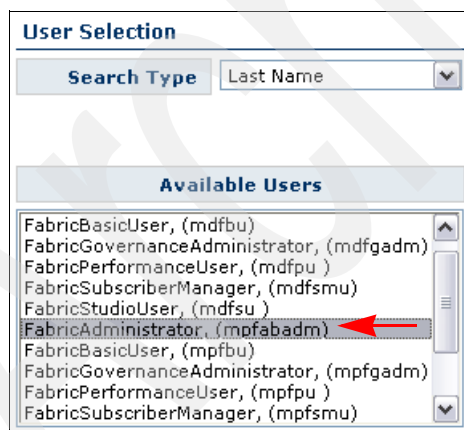


Figure 3-55 Users from the LDAP server listed in Available Users

2. Select the **FabricAdministrator (mpfabadm)** user and click the right arrow to move the selection into the Selected Users box (Figure 3-56).

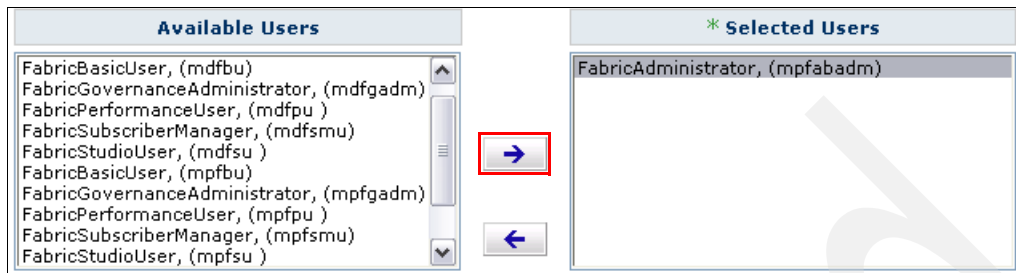


Figure 3-56 FabricAdministrator moved to Selected Users

3. Select the **Administrator** role under Available Roles and click the right arrow to move the selection into the Selected Roles box as shown in Figure 3-57.

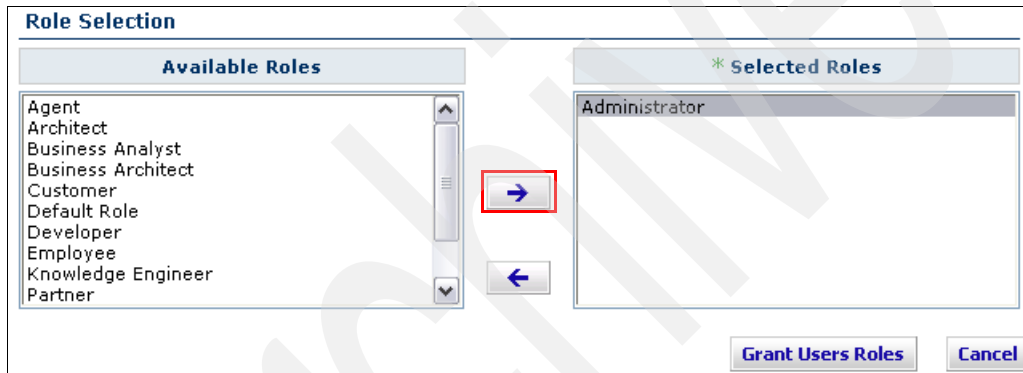


Figure 3-57 Administrator moved to Selected Roles

4. Click **Grant Users Roles** as shown in Figure 3-58.

**Available Users**

- FabricAdministrator, (fabric )
- FabricBasicUser, (mdfbu)
- FabricGovernanceAdministrator, (mdfgadm)
- FabricPerformanceUser, (mdfpu )
- FabricSubscriberManager, (mdfsmu)
- FabricStudioUser, (mdfsu )
- FabricAdministrator, (mpfabadm)
- FabricBasicUser, (mpfbu)
- FabricGovernanceAdministrator, (mpfgadm)
- FabricPerformanceUser, (mpfpu )

**\* Selected Users**

- FabricAdministrator, (mdfabadm)

**Role Selection**

**Available Roles**

- Agent
- Architect
- Business Analyst
- Business Architect
- Customer
- Default Role
- Developer
- Employee
- Knowledge Engineer
- Partner

**\* Selected Roles**

- Administrator

**Grant Users Roles** **Cancel**

Figure 3-58 Click Grant Users Roles

On the Users tab you will now see the selected User ID and its role under the User Selection section, as shown in Figure 3-59.

**User Selection**

View  rows at a time

1 rows Page 1 of 1

User Id	User Name	Email	Roles	Select
<u>mdfabadm</u>	FabricAdministrator,	mdfabadm@itso.ibm.com	Administrator	<input type="radio"/>

Page 1 of 1

Figure 3-59 User mdfabadm listed with its role under User Selection

## Enroll ITSOBankOrg to ITSOBankLoadApp

Perform the following steps to Enroll ITSOBankOrg to ITSOBankLoadApp.

1. Navigate to **Subscriber Manager** → **Manage Subscribers**. Click **ITSOBankOrg**, and then click the Enrollments tab.
2. Select **ITSOBankLoanApp** under the Enrollment Selection section. Click **Save Enrollments**.

When you select **ITSOBankLoanAPP**, the LoanProcessBS Business Service is also selected, as shown in Figure 3-60.

The screenshot shows the 'Manage Subscribers' interface. At the top, there's a header 'Manage Subscribers'. Below it, a filter bar shows 'Organization' set to 'ITSOBankOrg' and a 'Back to Search' button. A tabbed interface has 'Enrollments' selected. The 'Enrollment Selection' section contains a list of services. Two items are checked: 'ITSOBankLoanApp' (Application) and 'LoanProcessBS' (Business Service). A red box highlights the 'Save Enrollments' button at the bottom right.

Name	Type
<input checked="" type="checkbox"/> ITSOBankLoanApp	Application
<input checked="" type="checkbox"/> LoanProcessBS	Business Service

Figure 3-60 Selecting and saving the enrollments

## Subscribe the user ID to ITSOBankLoan App

Perform the following steps to Subscribe the user ID to ITSOBankLoan App.

1. Navigate to **Subscriber Manager** → **Manage Subscribers**. Click **ITSOBankOrg** and then click the Users tab.
2. Click the user **mdfabadm** (Figure 3-61), and then click the Subscriptions tab.

The screenshot shows the 'User Selection' interface. It includes a 'View' dropdown set to '10' and 'rows at a time'. Below this, it says '1 rows'. A table lists the user details. The 'mdfabadm' user ID is highlighted with a red box. The table has columns for 'User Id', 'User Name', 'Email', 'Roles', and 'Select'. Navigation buttons for 'Page 1 of 1' are visible at the bottom right.

User Id	User Name	Email	Roles	Select
mdfabadm	FabricAdministrator,	mdfabadm@itso.ibm.com	Administrator	<input type="radio"/>

Figure 3-61 Select the User Id mdfabadm

3. Select **ITSOBankLoanAPP** in the Subscription Selection section and click **Save Subscriptions** (Figure 3-62 on page 295).



Manage Subscribers

User

FabricAdministrator (mpfabadm) - mpfabadm@itso.ibm.com

Back to Search

User Details

Organizations

Subscriptions

Organization Selection

Organization

ITSOBankOrg

View Subscriptions

Subscription Selection

A list of Services that the current User can be Subscribed to for the selected Organization.

Name	Type
<input checked="" type="checkbox"/> ITSOBankLoanApp	Application
<input checked="" type="checkbox"/> LoanProcessBS	Business Service
<input checked="" type="checkbox"/> Loan Portal Channel	Portal Channel

Save Subscriptions

Figure 3-62 Select ITSOBankLoanApp and click Save Subscriptions

### 3.6.7 Install the sample application's EAR Files

Perform the following steps to Install the sample application's EAR Files

**Important:** Ensure that at least one cluster member in the WebSphere Process Server for z/OS cluster is started before attempting to deploy these applications.

1. Log in to the Integrated Solutions Console and navigate to **Applications** → **Install New Application**.
2. Choose **Local file system** and click **Browse**.
3. In the Additional Material, navigate to the folder code\Scenarios\Fabric\EAR, choose **ITSO\_implApp.ear** and click **Next**.
4. In the Step 1: Select installation options window, click **Next**.

5. In the Step 2: Map modules to servers window (Figure 3-63), check the boxes next to all the modules, then select the cluster and any web servers you may have configured from the **Clusters and Servers** list box. Click **Apply**. In our single-cluster topology the application was already mapped to correct cluster.

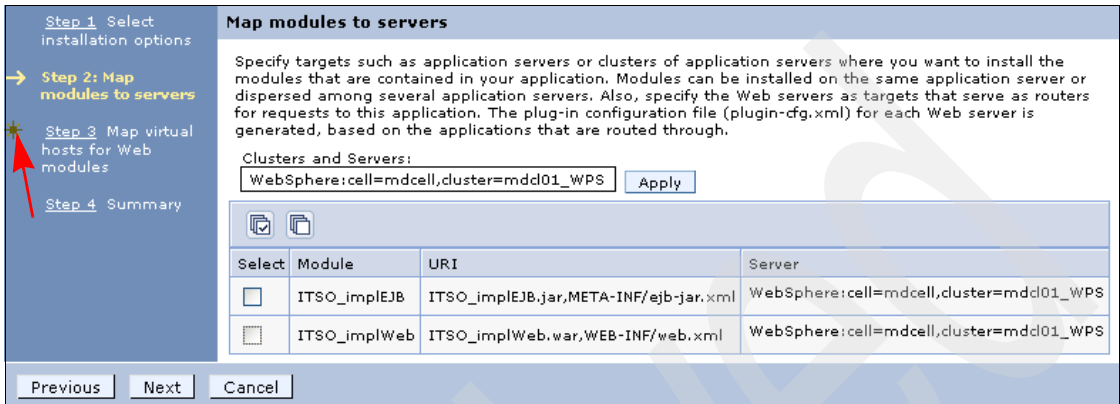


Figure 3-63 Mapping modules to the cluster

**Important:** Notice in Figure 3-63 the small circle with a cross to the left of Step 3. This is to draw your attention to this step. Do not skip past Step 3.

Check the mapping shown in the Server column on the right of the table as shown in Figure 3-64.

Server
WebSphere:cell=mdcell,cluster=mdcd01_WPS
WebSphere:cell=mdcell,cluster=mdcd01_WPS

Figure 3-64 Checking that the modules are mapped to the correct cluster

6. Click **Next**.

7. On the Step 3: Map virtual hosts for Web modules window (Figure 3-65), in the Virtual host list box, select **default\_host**, and click **Next**.

When you have done this, the circular icon with a cross next to Step 3 (mentioned in the note after Figure 3-63 on page 296) disappears.

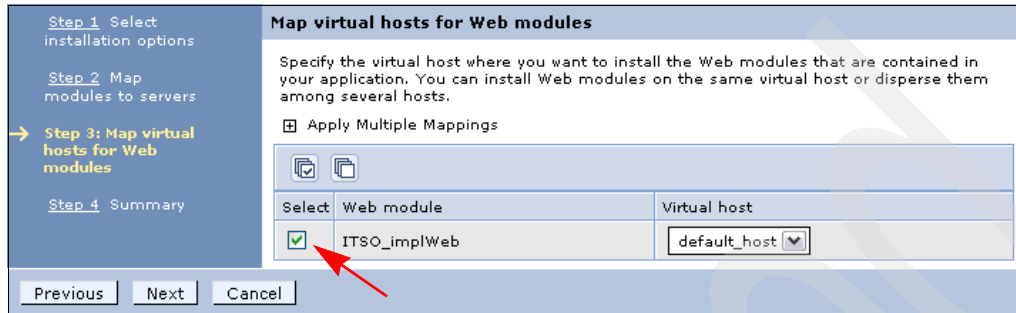


Figure 3-65 Be sure to check the box to select the default\_host

8. On the Step 4: Summary window, click **Finish**.
9. Click **Save**.
10. Repeat these steps to install WebSphereEnvUtilApp.ear and ITSOApp.ear from the folder code\Scenarios\Fabric\EAR directory of the Additional Materials.

**Note:** If you receive an error **CWWBF0029E: Process components of ITSOApp cannot be configured in WebSphere configuration repository** when deploying the ITSOApp, see Section 4.7.12, “CWWBF0029E: Process components of ITSOApp cannot be configured in WebSphere configuration repository” on page 343.

11. Navigate to **Applications** → **Enterprise Applications**.
12. Choose ITSOAPP, ITSO\_implApp, and WebSphereEnvUtilApp and click **Start**.

### 3.6.8 Map modules to the Web server

**Note:** This step may not be necessary if you checked the box to map all the applications to the Web server at the time you configured the Web servers.

When using a Web server to balance work across the servers and maintain affinity, it is necessary to map some of the applications to the web server in order for them to be in the plugin-cfg.xml file used by the WebSphere plugin in the HTTP Server.

Perform the following steps to map modules to the Web server.

1. In the Integrated Solutions Console, navigate to **Applications** → **Enterprise Applications** → **BPCExplorer\_mdcl01\_WPS** → **Manage Modules**.
2. Select all the modules.
3. Choose your Web server (in our environment this is itsowebserver) and the cluster where the application is already mapped under Clusters and Servers
4. Click **Apply** and click **OK**.
5. Save your changes.
6. Repeat these steps for the following applications:
  - Fabric\_Catalog
  - Fabric\_Engine
  - Fabric\_Tools
  - Fabric\_Tools\_Help (if installed)
7. Navigate to **Servers** → **Web servers** → **<webserver>** → **Generate Plug-in**.
8. Select **<webserver>** → **Propagate Plug-in**.

### 3.6.9 Update the SCA Import URLs

In the sample application, all the Web service SCA Imports are bound to URLs that point to itsodmgr. There is no port explicitly defined in the URLs, so the default HTTP port of 80 is used. You will likely need to change these to match your hostname:port.

To do so, perform the following steps:

**Note:** Ensure that at least one WebSphere Process Server for z/OS cluster member is started and that the three ITSO sample applications are started.

1. In the Integrated Solutions Console, navigate to **Applications** → **SCA Modules** and click **ITSO**.
2. Navigate to **Imports** → **CreditCheckImport** → **Bindings** as shown in Figure 3-66.

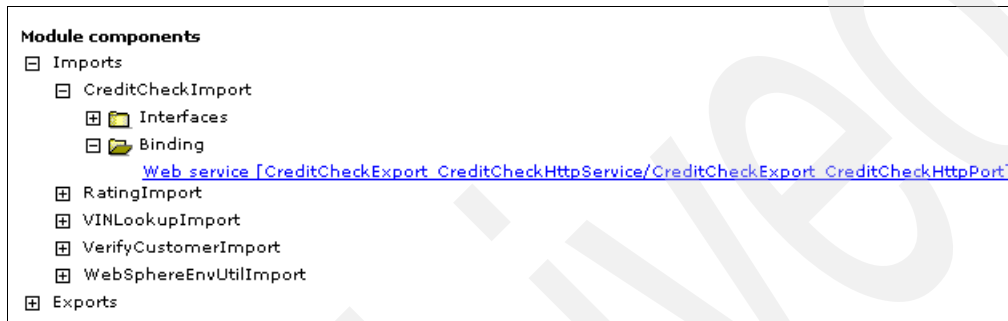


Figure 3-66 Expanding the CreditCheckExport bindings

3. Click **CreditCheckExport\_CreditCheckHttpService**.
4. Change the hostname and port number in the Endpoint field. Click **OK** as shown in Figure 3-67.

When you have a cluster with web servers running the WebSphere plug-in, you would normally specify the Distributed VIPA and port of the web server. Here we specified the HTTP port of the WebSphere Process Server for z/OS servers because we had not finished testing the web servers at the time we deployed the sample Fabric applications.

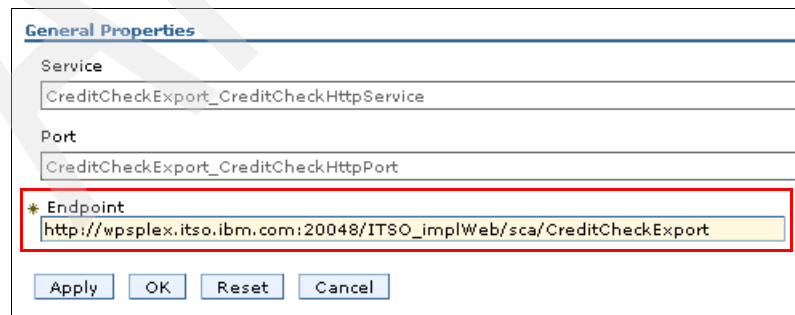


Figure 3-67 Specify the appropriate hostname and port

5. Repeat these steps for RatingImport, VinLookImport, and VerifyCustomerImport in the SCA module ITSO.
6. Exit the ITSO SCA module.
7. Navigate to **Applications** → **SCA Modules**. Click the **WebSphereEnvUtil** module and repeat these steps for the SubscriberManagerImport present in the module WebSphereEnvUtil (Figure 3-68).

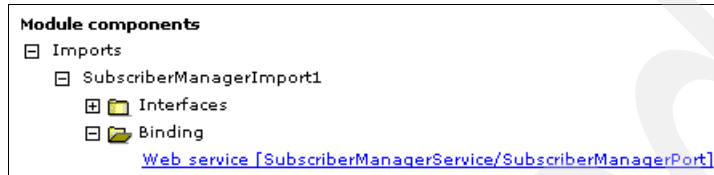


Figure 3-68 Updating SubscriberManagerService in WebSphereEnvUtil

8. Save the configuration changes.

### 3.6.10 Test the sample application

Perform the following steps to test the sample application:

1. Ensure that the user MDFABADM has the WebClientUser role and the eventCreator role. We connected user ID MDFABADM to the RACF groups MDWCUG and MDEVTCRG, which were on the access list for the RACF EJBROLE profiles called MD.WebClientUser and MD.eventCreator.

**Note:** If you try to log in to the BPC Explorer with a user ID that does not have access to the WebClientUser role, then login will fail, returning the following message:

You are not authorized to view this page

There will also be the following messages in the z/OS SYSLOG:

```

ICH408I USER(MDFABADM) GROUP(MDFADMG ) NAME(MD FABRIC ADMIN      )
MD.WebClientUser CL(EJBROLE )
INSUFFICIENT ACCESS AUTHORITY
ACCESS INTENT(READ  ) ACCESS ALLOWED(NONE  )

```

If you try to run the TestLoanProcess from BPC Explorer while logged in as MDFABADM, and that user is not authorized to create events, the following errors are issued:

```

ICH408I USER(MDFABADM) GROUP(MDFADMG ) NAME(MD FABRIC ADMIN      )
MD.eventcreator CL(EJBROLE )
INSUFFICIENT ACCESS AUTHORITY
ACCESS INTENT(READ  ) ACCESS ALLOWED(NONE  )

```

2. Log in to BPC Explorer with user MDFABADM. In our environment we used the following URL:  
`http://wpsplex.itso.ibm.com:20048/bpc`
3. Click **My Process Templates**, select the TestLoanProcess check box, and click **Start Instance**, as shown in Figure 3-69.

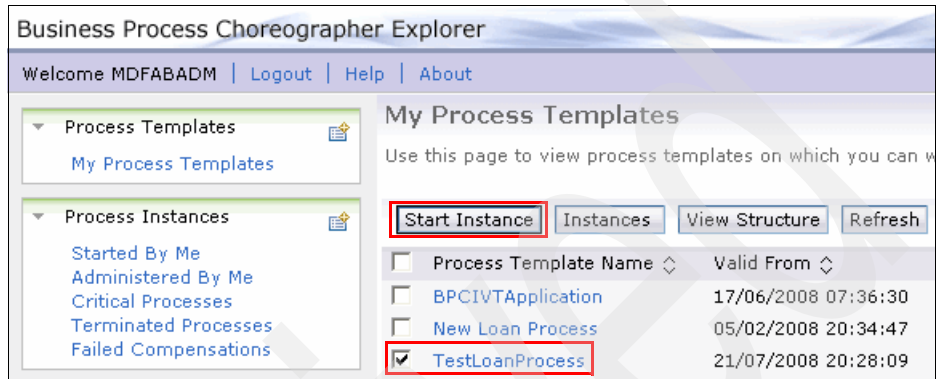


Figure 3-69 Start an instance of TestLoadProcess

**Note:** If you do not see the New Loan Process and TestLoadProcess listed in BPC Explorer, check that you have started ITSOApp.

4. Specify the following values:
  - CustomerIdentificationNumber: 100
  - CustomerAddress: <Any string value>
  - VIN: 12345678901234567
  - LoanAmountRequested: 1000
  - BankID: ITSO

5. Click **Submit**. The results look like Figure 3-70.

### Process Output Message

Use this page to view the results of a business process that you started. [i](#)

Process Template Name TestLoanProcess  
Operation processLoan  
Process Input Message

#### Form View

Input	CustomerIdentificationNumber	100
	CustomerAddress	Davis Rd
	VIN	12345678901234567
	LoanAmountRequested	1000.0
	BankId	ITSO

[View Source](#)

Process Output Message

#### Form View

Output	LoanAmountSanctioned	1000.0
	LoanStatus	APPROVE
	InterestRate	4.565

[View Source](#)

Figure 3-70 Sample application test result

**Note:** In the sample application, the TestLoanProcess acts as a proxy to the New Loan Process. The TestLoanProcess calls the New Loan Process through a context injector, which injects the required context for WebSphere Business Services Fabric.

If you receive the error **CWWBU0001E: A communication error occurred when the 'class com.ibm.bpe.clientmodel.command.CallCommand' function was called** when running TestLoanProcess, see Section 4.7.13, “CWWBU0001E in BPC Explorer with TestLoanProcess” on page 345.



## 3.7 Troubleshooting the WebSphere Business Services Fabric for z/OS installation

This section describes how to diagnose problems with the installation. There are also a series of problems and solutions in Section 4.7, “Errors related to WebSphere Business Services Fabric for z/OS” on page 334.

- ▶ Go to DB2 and verify the creation of 8 tables for the fabricbus messaging engine. For our cell, these were in database MD01FDB and they had a schema of MD01F.
- ▶ Go to the Integrated Solutions Console and verify the scope of the all JMS provider resources created for fabric. For a single-cluster topology there are unlikely to be any problems, but in the two-clusters topology you might have defined one or more resources in the wrong cluster. For the two-clusters topology, all Fabric resources are defined in the AppTarget cluster (for our MP cell, this was **mpcl02.AppTarget**).
- ▶ Verify the JNDI Names are the same for the JMS Provider resources as specified in Section 3.4.6, “Configure the JMS provider” on page 259.
- ▶ Verify the Destination JNDI Names are given properly for the Activation specifications as specified in “Create the activation specifications” on page 262.
- ▶ If you have enabled security on the buses go through the checks in Section 3.4.8, “Configure security” on page 267.
- ▶ Check the Server logs for any java.io.FilePermisssion exceptions for Fabric applications related to Java security. If exceptions are present, grant the permissions in the policy file or disable Java 2 Security and restart all the servers.

## 3.8 WebSphere Business Services Fabric events

WebSphere Business Services Fabric generates several Common Base Events (CBEs) that can be monitored by WebSphere Business Monitor. For details on how to configure WebSphere Business Services Fabric to generate events, see Redbooks publication *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665.



# Troubleshooting WebSphere Process Server for z/OS

This chapter discusses some of the errors we encountered creating the Business Process Management production topology on z/OS, and describes how we overcame these errors.

This chapter contains the following sections.

- ▶ “Sources of information about an error” on page 306
- ▶ “WebSphere Process Server errors” on page 306
- ▶ “Errors relating to BPC Event Collector and Observer” on page 325
- ▶ “Errors relating to BPC Event Collector and Observer” on page 325
- ▶ “Errors relating to Business Space” on page 328
- ▶ “Errors related to People Directory Provider” on page 333
- ▶ “Errors related to WebSphere Business Services Fabric for z/OS” on page 334

## 4.1 Sources of information about an error

If you experience a problem with one of the configuration tasks, there are four main sources of information about the problem:

- ▶ The error messages issued by the task.
- ▶ Error messages in the WebSphere Deployment Manager or WebSphere Process Server for z/OS job logs. If you are federating a node you might also find messages in the node agent's logs.
- ▶ When running zSMPEInstall.sh and zWPSConfig.sh, first look in the installconfig.log at <wps\_home>/logs/wbi/install/installconfig.log.
  - Find SEVERE to determine error
  - Find Buildfile previous to see which .ant task was running
  - Look in corresponding .ant.log file(s) at <wps\_home>/logs/wbi for more information
- ▶ FFDC logs in <wps\_home>/profiles/default/logs/ffdc.

## 4.2 WebSphere Process Server errors

In this section we have recorded some of the errors we encountered when configuring WebSphere Process Server for z/OS V6.1.2 and how we resolved them. These errors tend to be human errors such as a failure to set the schema properly, or a failure to grant required permissions to the schema or to the database J2C authentication alias user ID.

If the schema is not correct, the error messages issued in the logs can be numerous and varied. Sometimes they do not contain anything to indicate a schema error. In this instance, look in the FFDC logs, because certain SQLCODE errors are reported there.

## 4.2.1 zSMPEInstall.sh failed RC=12

**Problem:** When running the zSMPEInstall.sh for the second node, it failed with a RC=12 and the messages shown in Example 4-1.

*Example 4-1 zSMPEInstall.sh failed RC=12 error message*

---

```
BPXBATCH SH cd /wasv61config/mdcell/mdnodeb/AppServer/bin; export
PATH=.:$PATH;
/wasv61config/mdcell/mdnodeb_wps_smpe/zos.config/bin/zSMPEInstall.sh
'-smproot /wasv61config/mdcell/mdnodeb_wps_smpe' '-runtime
/wasv61config/mdcell/mdnodeb/AppServer'
'-install' 1> /tmp/mdnodebinstall_0511.out 2>
/tmp/mdnodebinstall_0511.err
READY
END
IKJ56644I NO VALID TSO USERID, DEFAULT USER ATTRIBUTES USED
READY
      BBOHFSWR '/tmp/mdnodebinstall_0511.out'
ERROR: Unable to read file PATH. Error: 81 5620062
READY
      BBOHFSWR '/tmp/mdnodebinstall_0511.err'
ERROR: Unable to read file PATH. Error: 81 5620062
READY
```

---

**Cause:** The problem was caused by having an invalid UNIX environment for the user ID executing the job. The user ID (MDADMIN) did not have a home directory on the secondary LPAR because the BBOSBRAM job, which creates the home directory for MDADMIN, had only been run on the primary LPAR.

**Solution:** Ensure the user ID running zSMPEInstall.sh has a valid UNIX environment by logging on to a telnet client using the user ID.

Alternatively, use an OMVS session and try to su to that user ID. Echo \$PATH, \$CLASSPATH, and so forth, and try commands like **java -fullversion** to ensure that the user ID is working normally in UNIX on the LPAR where you are running the job.

## 4.2.2 Failure in loading T2 native library db2jcct2zos

**Problem:** Failure in loading T2 native library db2jcct2zos when initializing the servant or adjunct. This error can have several causes.

**Cause 1:** The SDSNLOAD and SDSNLOD2 libraries are not present on the STEPLIB of the address space reporting the error.

**Solution 1:** Add the SDSNLOAD and SDSNLOD2 to the STEPLIB

**Cause 2:** You are trying to use the Test Connection function on a Type 2 data source using the Integrated Solutions Console and the data source is defined at cluster level. Typically the error also includes the text shown in Example 4-2.

*Example 4-2 Error text for second cause of failure in loading T2 native library db2jcct2zos*

---

```
reason: java.lang.UnsatisfiedLinkError: db2jcct2zos (Not found in
java.library.path)DSRA0010E: SQL State = null, Error Code = -99,999
```

---

**Solution 2:** There is a limitation in WebSphere Application Server for z/OS that Test Connection will not work for data sources defined at node or cluster level. You will find that you can test the connection of the jdbc/WPSDB data source that is defined at cell level but not the other data sources defined at cluster level.

## 4.2.3 -922 SQLCODEN922 connecting to DB2

**Problem:** The SQLException in Example 4-3 is issued when connecting to DB2.

*Example 4-3 SQLException issued when connecting to DB2*

---

```
java.sql.SQLException:
[IBM/DB2][T2zos/3.6.67]T2zosReusableConnection.flowConnect:execConnect:
DB2 engine SQL error, SQLCODE = -922, SQLSTATE = 42505, error tokens =
CONNECT;00D31024DSRA0010E: SQL State = 42505, Error Code = -922
```

---

**Cause:** The most common reason for this error is specifying an incorrect Location name in the Database name field of a Type 2 data source definition. This error is likely to happen because several of the wizards will set the database name in the Database name field. For a Type 2 data source, you need to set the DB2 Location name in the Database name field.

**Solution:** Make sure that the Database name property on the data source is set to the DB2 z/OS Location name rather than the database name.

## 4.2.4 -551 SQLCODEN551 connecting to DB2

**Problem:** When trying to start the BPC Event Collector and Observer applications, they failed to start giving -551 errors in the servant logs.

The message in Example 4-4 appears in the servant log:

*Example 4-4 -551 SQLCODEN551 connecting to DB2 servant log message*

---

```
ExtendedMessage: CWWB04005E: A failure occurred during the check of
database catalog:
com.ibm.db2.jcc.t2zos.cb: [IBM/DB2] [T2zos/3.6.67]
T2zosPreparedStatement.readPrepareDescribeOutput_:
nativePrepareInto:1512: DB2 engine SQL error, SQLCODE = -551, SQLSTATE
= 42501,
error tokens = MDASRU;SELECT;SYSIBM.SYSTABLES.
```

---

**Cause:** The Data source called Business Process Choreographer data source was created without the schema being set on either currentSQLID or currentSchema.

**Solution:** Set the currentSQLID property on the Business Process Choreographer Event Collector data source to the schema of the BPC Event Collector and Observer tables and then restart the server.

## 4.2.5 Basic authentication failed for user ...

**Problem:** When starting the server for the first time we received an error (Example 4-5) in the servant log.

*Example 4-5 Authentication failure error message*

---

```
SECJ6219I: Basic authentication failed for user "MDJMSU".
The native service results related to the authentication failure are:
WebSphere service=BBOSSNAP, WebSphere returnCode=33,
SAF service=IRRSIA00 INTA_CREATE, SAF returnCode=8,
SAF product returnCode=8, SAF product reasonCode=32.
```

---

**Cause:** There could be something wrong with the way the password for the user ID MDJMSU was set up. For example, the RACF password for the user ID might not match the password set on the J2C Authentication Alias entry. Or you might have forgotten to set the initial password to NOEXPIRED when you defined the user ID to RACF.

The error message can be misleading because a Basic authentication failure can also be caused when RACF APPL class is active and the user ID does not have access to the RACF profile in class APPL that controls access to the cell. When the cell is created, a profile in class APPL is defined with a name equal to the name of the security domain.

For example, for our cell called MDCELL, the security domain prefix was MD so a RACF profile was created in class APPL called MD.

The APPL profile controls which users can connect to the system. This also applies to the J2C Authentication Alias user IDs.

**Solution:** Permit the J2C Authentication Alias user IDs to the APPL profile and then restart the server.

For example, we issued the following commands to permit RACF groups MDJMSG and MDDBG (the default groups of the two J2C Authentication Alias user IDs, MDJMSU and MDDBU respectively) to the profile called MD in class APPL:

```
PE MD CLASS(APPL) ID(MDJMSG,MDDBG)
SETR RACLIST(APPL) REFRESH
```

Diagnosing this problem is hard without detailed traces, so it is useful to remember the symptoms described above and know to check the APPL profile.

## 4.2.6 SCHD0128E: The daemon for scheduler ... could not be started

**Problem:** The error shown in Example 4-6 was seen in the servant log when it was started for the first time after configuring BPC.

*Example 4-6 Servant log error after configuring BPC*

---

```
ExtendedMessage: BB000220E: SCHD0128E: The daemon for scheduler
BPEScheduler could not be started:
com.ibm.ws.scheduler.exception.SchedulerDataStoreException:
SCHD0124E: Unable to initialize TaskStore due to error:
SCHD0046E: The table MDWPS.SCHED_TREG in datasource jdbc/BPEDB used by
scheduler resource BPEScheduler
(BPEScheduler) is missing or is not accessible.
```

---

**Cause:** When you receive messages like those in Example 4-6, check that the schema in the message is correct. Often the reason the table is not found is that you are using an incorrect schema to access DB2.



In this case, the schema MDWPS was correct and the message was telling the truth: The table MDWPS.SCHED\_TREG did not exist in DB2.

The table was missing because when we copied the BPC DDL from the UNIX file system into a partitioned dataset, we did not notice that the target dataset was out of space. Part of the DDL for BPC was not copied so the table SCHED\_TREG had not been created.

**Solution:** Enlarge the dataset holding the BPC DDL, then re-copy it from the UNIX file system. Repeat any editing performed on the DDL. Drop the BPC database and re-create it. Then re-run the BPC DDL again. Restart the server.

#### 4.2.7 Resources not seen in the Administration console

**Problem:** Sometimes you do not see Application Servers after creating a cluster with cluster members, or you do not see applications after installing them.

**Cause:** When you are logged in to the Integrated Solutions Console while configuration changes are being made, the cached pages may not reflect the new state of the configuration.

**Solution:** Log off the Integrated Solutions Console and log in again.

#### 4.2.8 Events not seen in the CBE Browser during verification

**Problem:** When trying to verify the function of the CBE Browser you always see zero events listed. There are two common causes of this problem.

**Cause 1:** The user ID executing the task that you expect to be generating the events does not have access to the CEI role eventCreator.

**Solution 1:** When using a RACF user registry, ensure that you have created RACF profiles in CLASS EJBROLE or GEJBROLE for all the CEI roles and that the user ID trying to create events is permitted to the eventCreator role.

Remember that if the cell has security enabled, but the application is not secured and therefore is not forcing you to authenticate before using it, then the application is running UNAUTHENTICATED. For WebSphere Process Server for z/OS, this means the application is running under the guest user ID. Because the guest user ID is a RESTRICTED user ID in RACF, it is not allowed to gain access to profiles which are UACC(READ). This means that you must explicitly permit the guest user ID to any RACF EJBROLE profiles.

**Cause 2:** Unless the application you are testing creates CEI events explicitly, you will need to enable traces in WebSphere Process Server for z/OS in order for CEI events to be created when the business process executes.

When you use the ZOSBookOrderApp, for example, you have to enable traces of WBILocationMonitor.CEI.SCA.com\*.

The process for verifying the CBE Browser is described in detail in the White Paper WP101218, *Performing Installation Verification for WPS on z/OS V6.1*, available at the following Web page:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101218>

**Solution 2:** Remember to enable trace for WBILocationMonitor.CEI.SCA.com\* when you expect to generate events using ZOSBookOrderApp or bpcivt as part of verifying the CBE browser.

**Cause 3:** The CEI bus is not started. There are clear errors in the adjunct's log if this is the case.

**Solution 3:** Resolve the errors being issued so that the CEI bus starts normally.

## 4.2.9 Could not start TM\_PredefinedTasks\_V612\_mdcl01\_WPS

**Problem:** The errors shown in Example 4-7 were seen at several points in the servant log.

*Example 4-7 Servant log errors*

---

```
com.ibm.ws.exception.RuntimeWarning: could not start application
HTM_PredefinedTasks_V612_mdcl01_WPS because of filter
com.ibm.bpe.framework.ProcessContainer@33fe33fe
```

---

Although the message says the application cannot be started, it did show as partially started on the WebSphere Administration console.

**Cause:** The message appears to be another symptom of having an incorrect schema. The problem disappeared after we had resolved the -551 and -922 errors in the logs.

**Solution:** Ensure all the buses specify the correct schema. Ensure all the datasources specify the correct schema on the currentSQLID or currentSchema custom properties.

## 4.2.10 WSVR0220I: Application stopped: TaskContainer\_mdcl01\_WPS

**Problem:** The application TaskContainer\_mdcl01\_WPS attempted to start during server initialization but then was seen to stop.

**Cause:** The application started normally once the database J2C Authentication Alias user ID had been granted SELECT on SYSIBM.SYSTABLES, so it appears to be a symptom of the problem described in Section 4.2.12, “-551 MDDBU;SELECT;SYSIBM.SYSTABLES” on page 314.

**Solution:** Grant SELECT on SYSIBM.SYSTABLES to the database J2C Authentication Alias user ID.

Ensure all the buses specify the correct schema. Ensure all the datasources specify the correct schema on the currentSQLID or currentSchema custom properties.

## 4.2.11 BBOO0220E: J2CA0081E: Method cleanup failed

**Problem:** The errors shown in Example 4-8 were issued together with the previous two when the server was started.

*Example 4-8 Server start errors*

---

```
ExtendedMessage: BBOO0220E: J2CA0081E: Method cleanup failed while
trying to execute method cleanup on ManagedConnection
WSRdbManagedConnectionImpl@4e004e from resource jdbc/BPEDB.
Caught exception: com.ibm.ws.exception.WsException:
DSRA0080E: An exception was received by the Data Store Adapter.
See original exception message: Cannot call 'cleanup' on a
ManagedConnection while it is still in a transaction..
```

---

**Cause:** Not known for certain. It is likely related to switching from using currentSQLID to using currentSchema on the data sources. The error was noticed after making that change. We suspect that the WebSphere transaction manager was trying to clean up a failed connection, but because the mode of accessing DB2 had changed, there was a problem.

**Solution:** We deleted everything under the <wps\_home>/profiles/default/tranlog directory for both nodes and restarted the cluster. Restarting the cluster on its own had not resolved the problem, but deleting the contents of /tranlog resolved it.

## 4.2.12 -551 MDDBU;SELECT;SYSIBM.SYSTABLES

**Problem:** The -551 SQLCODEN551 error in Example 4-9 was seen in FFDC logs issued in servant.

*Example 4-9 -551 SQLCODEN551 servant log error*

---

```
Exception = com.ibm.db2.jcc.t2zos.cb
Source = com.ibm.ws.rsadapter.jdbc.WSJdbcConnection.prepareStatement
probeid = 1584 Stack Dump = com.ibm.db2.jcc.t2zos.cb:
[IBM/DB2] [T2zos/3.6.67] T2zosPreparedStatement.readPrepareDescribeOutput
_:nativePrepareInto:1512: DB2 engine SQL error, SQLCODE = -551,
SQLSTATE = 42501, error tokens = MDDBU;SELECT;SYSIBM.SYSTABLES
```

---

**Cause:** The obvious cause is that SELECT on the DB2 catalog tables is not granted to PUBLIC and the J2C Authentication Alias user ID, MDDBU, has not been granted SELECT on SYSIBM.SYSTABLES.

However, that an attempt is made to access SYSIBM.SYSTABLES may indicate that a component of WebSphere Process Server for z/OS has failed to locate its tables. The underlying cause could also be one of the following:

- ▶ The correct schema name is not specified on a bus definition.
- ▶ The correct schema name is not specified on either the currentSQLID or correctSchema custom property of a datasource.

**Solution:** Grant select on SYSIBM.SYSTABLES to the database J2C Authentication Alias user ID.

Ensure that all the buses specify the correct schema. Ensure that all the data sources specify the correct schema on the currentSQLID or currentSchema custom properties. We included the necessary GRANT in the sample job MDCREDB that is provided with the additional material in Appendix A, “Additional material” on page 351.

## 4.2.13 SECJ4062W: Cannot find the credential information

**Problem:** The errors shown in Example 4-10 were seen in the adjunct logs just after running the BPC Configuration wizard.

### *Example 4-10 Adjunct log errors*

---

```
Trace: 2008/08/24 21:34:31.698 01 t=7BF2D8 c=UNK key=P8 (13007002)
  ThreadId: 00000030
  FunctionName: com.ibm.ws.security.auth.j2c.WSDefaultPrincipalMapping
  SourceId: com.ibm.ws.security.auth.j2c.WSDefaultPrincipalMapping
  Category: WARNING
  ExtendedMessage: BB000221W: SECJ4062W: Cannot find the credential
information.
Trace: 2008/08/24 21:34:31.698 01 t=7BF2D8 c=UNK key=P8 (0000000A)
  Description: Log Java Message
  Message: BB000221W: SECJ4062W: Cannot find the credential information.
Trace: 2008/08/24 21:34:31.703 01 t=7BF2D8 c=UNK key=P8 (13007002)
  ThreadId: 00000030
  FunctionName: com.ibm.ejs.j2c.PrivExAction
  SourceId: com.ibm.ejs.j2c.PrivExAction
  Category: SEVERE
  ExtendedMessage: BB000220E: J2CA0044E: The ConnectionManager failed to get a
Subject from the security service associated with connection factory
jdbc/com.ibm.ws.sib/mpcl01_WPS-BPC.mpcell.Bus.
Received exception javax.security.auth.login.LoginException:
Incorrect authDomainEntry and alias is: BPCME_00_Auth_Alias
```

---

**Cause:** The credential in question is the user ID and password in the J2C authentication alias called BPCME\_00\_Auth\_Alias. At the time the error occurred, this alias had just been created by running the BPC configuration wizard but the cluster had not been restarted. RACF-related control blocks for the user IDs on J2C authentication aliases are built during server initialization so a restart is needed before they can be used.

**Solution:** Restart the cluster.

## 4.2.14 Errors starting Event Collector and Observer

**Problem:** After deploying the Event Collector and Observer application, we received the errors in Figure 4-1 when we tried to start them.

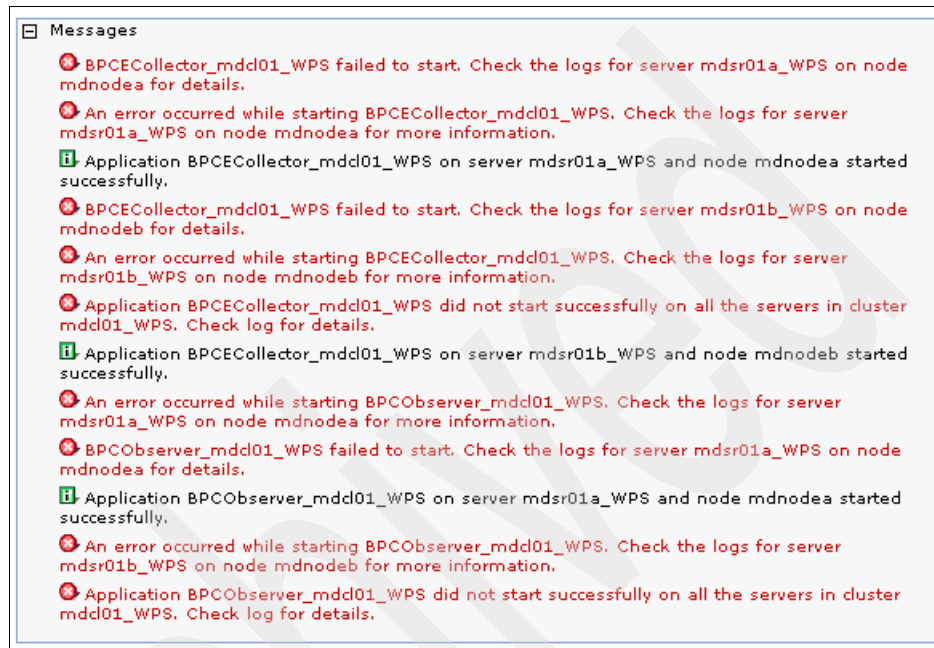


Figure 4-1 Errors when trying to start Event Collector and Observer

The servant logs showed the error messages in Example 4-11.

### Example 4-11 Servant log errors

```
ExtendedMessage: CWWB04005E: A failure occurred during the check of
database catalog: com.ibm.db2.jcc.t2zos.cb: [IBM/DB2][T2zos/3.6.67]
T2zosPreparedStatement.readPrepareDescribeOutput_:
nativePrepareInto:1512: DB2 engine SQL error, SQLCODE = -551, SQLSTATE
= 42501, error tokens = MDASRU;SELECT;SYSIBM.SYSTABLES.
```

**Cause:** The presence of MDASRU (the servant region user ID) in the message suggested that the Event Collector application might not be using the correct schema or sqldid when accessing the database.

When the Event Collector is deployed a new datasource is created (Figure 4-2 on page 317).

<input type="checkbox"/>	<a href="#">Business Process Choreographer Event Collector data source</a>	jdbc/BPCEventDB_mdcl01_WPS
--------------------------	--	----------------------------

Figure 4-2 Event Collector datasource

When we navigated to **Data sources** → **Business Process Choreographer Event Collector data source** → **Custom properties** we found that, unlike the other datasources, the currentSQLID property had not been set, nor was currentSchema set. This explained why the database access was being attempted under the servant region's user ID.

**Solution:** Set the currentSQLID property or currentSchema property to the schema name. For our MD cell this was MDWPS.

Save the changes and restart WebSphere Process Server for z/OS.

When the server re-starts the Event Collector and Observer applications start successfully.

#### 4.2.15 -812 SQLCODEN812 when starting BPC Observer

**Problem:** When BPC Observer was started we saw the error in Example 4-12 on one node only.

Example 4-12 Error when starting BPC Observer

---

```
Trace: 2008/08/13 10:19:06.700 01 t=7A0D90 c=UNK key=P8 (0000000A)
  Description: Log Java Message
  Message:
[IBM/DB2] [T2zos/3.6.67] T2zosPreparedStatement.readPrepareDescribeOutput_:
nativePrepareInto:1512: DB2 engine SQL error, SQLCODE = -812, SQLSTATE = 22508,
error tokens = SYSLN303.5359534C564C3031;DSNRRSAF
BPCObserverModelCat BPCObserverModelCat
```

---

**Cause:** The NULLID.\* collection had not been bound on the DB2 instance running on the LPAR where we saw the error.

**Solution:** The DB2 system administrator re-bound the NULLID.\* collection, which is one of the required steps in the configuration of the DB2 Universal JDBC Driver.

Refer to *Installing the IBM Data Server Driver for JDBC and SQLJ* in the DB2 Information Center, which can be found at the following Web page:

<http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp>

## 4.2.16 Product registration failed for ProcessServerZOS

**Problem:** RegistrationException error in the servant log.

```
com.ibm.ws390.product.RegistrationException: Return code 4 issued from
product registration
    at com.ibm.ws390.product.Registration.register(Unknown Source)
```

... and in the Adjunct ...

```
Trace: 2008/08/21 19:22:05.247 01 t=7FF028 c=UNK key=P8 (0000000A)
Description: Log Boss/390 Error
from filename: ./bboopreg.cpp
at line: 523
error message: BB000333E PRODUCT 'ProcessServerZOS'
FAILED TO REGISTER WITH IFAUSAGE SERVICE.
Return code: 4
Trace: 2008/08/21 19:22:05.257 01 t=7FF028 c=UNK key=P8 (13007002)
ThreadId: 0000000a
FunctionName: com.ibm.ws.runtime.component.ContainerImpl
SourceId: com.ibm.ws.runtime.component.ContainerImpl
Category: SEVERE
ExtendedMessage: BB000220E: WSVR0501E: Error creating component null
[class com.ibm.ws390.product.ProductRegistrationComponentImpl]

com.ibm.ws.exception.RuntimeWarning: Unable to determine configured
product type
```

**Cause:** WebSphere Application Server APAR PK66900

As of WebSphere Application Server V6.1.0.16, usage reporting for sub-capacity pricing is done through the IFAUSAGE system service, and a non-zero return code from IFAUSAGE is reported as a failure. If WebSphere Service Registry and Repository is installed, two registrations will take place from the same address space, causing IFAUSAGE to return a code of 4. This case can be considered a successful registration.

**Solution:** Install APAR PK66900, which is currently targeted for inclusion in Service Level (Fix Pack) 6.1.0.18 of WebSphere Application Server for z/OS V6.1 for z/OS.



## 4.2.17 CWSIJ0063E: A network connection to host name x.x.x.x, port yyyy cannot be established

When running in a cluster, the following error message may be issued in the servant log when trying to access a bus running on another cluster member.

CWSIJ0063E: A network connection to host name 9.12.4.145, port 22051 cannot be established

Usually there is a related FFDC log in which shows the cause to be an SSL handshake\_failure as shown in Example 4-13.

### Example 4-13 SSL handshake\_failure

---

Caused by: com.ibm.ws.sib.jfapchannel.JFapConnectFailedException: CWSIJ0063E: A network connection to host name 9.12.4.145, port 22051 cannot be established.

..

Caused by: javax.net.ssl.SSLException: javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake\_failure  
at com.ibm.jsse2.qb.a(qb.java:793)

---

**Cause:** The error occurs because the BBOMBRAK job that is generated to define RACF profiles when configuring an empty node does not include commands to generate a personal certificate for the servant region user ID. However a personal certificate is created for the controller user ID.

**Solution:** Copy the RACDCERT commands from .DATA member BBOMBRAK that created the personal certificate for the controller user ID. Edit these to create a personal certificate for the servant region user ID and connect it as the DEFAULT certificate to the servant region's keyring. Example 4-14 shows the commands we created to add a personal certificate for the servant region user ID MDASRU, and to connect that certificate to the keyring WASKeyring.MDCELL.

### Example 4-14 Sample commands to define a personal certificate for the servant

---

RACDCERT ID (MDASRU)	+
GENCERT SUBJECTSDN(CN('wspplex.itso.ibm.com'))	+
O('IBM')	+
OU('MDCELL.MDASRU'))	+
WITHLABEL('DefaultWASServantCert.MDCELL')	+
SIGNWITH(CERTAUTH LABEL('WebSphereCA.MDCELL'))	+
NOTAFTER(DATE(2010/12/31))	
RACDCERT ID(MDASRU)	+
CONNECT	+
(LABEL('DefaultWASServantCert.MDCELL')	+
RING(WASKeyring.MDCELL ) DEFAULT)	
SETR RACLIST(FACILITY) REFRESH	

---

**Note:** After creating a cluster you may not see evidence that this problem exists until you test workload balancing across the cluster.

In our case, we only realized there was a problem when we failed to deploy the ITSOApp application as described in Section 4.7.12, “CWWBF0029E: Process components of ITSOApp cannot be configured in WebSphere configuration repository” on page 343. While investigating that problem we found the FFDC log that indicated an SSL handshake error was occurring.

When all the buses were running in the same server as the servant making the request to the message engine, there was no problem because no SSL handshake is necessary. The issue occurs when a servant on one cluster member has to contact the active message engine running in the adjunct of another cluster member.

## 4.3 Errors when launching bpcivt

This section contains some of the errors we experienced when running bpcivt.

### 4.3.1 Error 403: AuthorizationFailed

**Problem:** When launching the bpcivt from a browser for the first time we received the following error:

Error 403: AuthorizationFailed

In the SYSLOG there were some ICH408I messages which indicated a role authorization problem, as shown in Example 4-15.

*Example 4-15 ICH408I messages indicating a role authorization problem*

---

```
ICH408I USER(MDADMIN ) GROUP(MDCFG ) NAME(WAS ADMINISTRATOR ) 858
MD.BPCIVTUser CL(EJBROLE )
INSUFFICIENT ACCESS AUTHORITY
ACCESS INTENT(READ ) ACCESS ALLOWED(NONE )
+BB000222I: SECJ0129E: Authorization failed for MDADMIN while
invoking GET on default_host:/bpcivt/, Authorization failed,
Not granted any of the required roles: BPCIVTUser
```

---

**Cause:** In this case the problem was caused by invoking bpcivt from a browser that was already logged into the Integrated Solutions Console with user ID MPADMIN on another tab. The LTPA token of MDADMIN was sent to the WebSphere Process Server for z/OS server but MDADMIN had not been permitted to the EJBROLE profile MD.BPCIVTUser.

**Solution:** Open a different browser. You are prompted to log in. Log in with a user that has been permitted to the role BPCIVTUser. For our system we had permitted RACF group MDWCUG to the BPCIVTUser role. Group MDWCUG contained the user MDWCU. Therefore, we logged on with user MDWCU and successfully executed the bpcivt script.

### 4.3.2 NMSV0610I: NamingException, CORBA.NO\_PERMISSION, The WSCredential does not contain a forwardable token ... enable Identity Assertion

**Problem:** When trying to run the bpcivt from a browser the errors in Example 4-16 were seen in the servant log.

*Example 4-16 Servant log errors when trying to run the bpcivt from a browser*

---

```
Trace: 2008/09/04 06:29:13.384 01 t=7B3D90 c=UNK key=P8 (13007002)
ThreadId: 00000025
FunctionName: com.ibm.ws.naming.util.Helpers
SourceId: com.ibm.ws.naming.util.Helpers
Category: WARNING
ExtendedMessage: BB000221W: NMSV0610I: A NamingException is being
thrown from a javax.naming.Context implementation.
Details follow:
Context implementation: com.ibm.ws.naming.jndicos.CNContextImpl
Context method: lookupExt
Context name: mpcell/clusters/mpc101.WPS_SandM
Target name: com/ibm/events/configuration/bus-transmission/Default
Other data: ""
Exception stack trace: javax.naming.NoPermissionException:
NO_PERMISSION exception caught [Root exception is
org.omg.CORBA.NO_PERMISSION: Cannot create service context.
Message: The WSCredential does not contain a forwardable token.
Please enable Identity Assertion for this scenario.
vmcid: 0x0 minor code: 0 completed: No]
```

---

**Cause:** At first sight this appears to be a problem with JNDI naming but the lookup for the CEI emitter is specifying the correct, fully-qualified JNDI name. The cause is the error related to CORBA.NO\_PERMISSION. It is necessary to

enable CSiv2 Identity Assertion when the Business Process Container is in a different cluster from the CEI, because an LTPA token is not automatically forwarded on an RMI/IIOP call when initiated by asynchronous tasks.

**Solution:** Enable CSiv2 Identity Assertion as described in Section 2.18.2, “Enable CSiv2 Asserted Identity” on page 171.

### 4.3.3 CWSIT0086E: Bus SCA.SYSTEM.mpcell.Bus not found

**Problem:** In a multi-cluster configuration, an FFDC log was taken at the end of the startup of the servant running the Business Process Container. The BPC cluster (AppTarget) has a remote SCA configuration. An extract from the FFDC is shown in Example 4-17.

*Example 4-17 FFDC extract*

---

```
Exception = com.ibm.websphere.sib.exception.SIResourceException
Source =
com.ibm.ws.sib.ra.inbound.impl.SibRaStaticDestinationEndpointActivation
.createRemoteListener probeid = 5
Stack Dump = com.ibm.websphere.sib.exception.SIResourceException:
CWSIT0086E: Bus SCA.SYSTEM.mpcell.Bus not found
```

---

**Cause:** The cluster running BPC finished initializing before the Support and Messaging cluster so the SCA bus had not yet started.

**Solution:** When you need to start both clusters, stagger the start commands for the clusters to ensure that the Support and Messaging cluster is initialized before the AppTarget cluster. No problem is caused if you see this FFDC, but it is best to avoid the problem occurring. The AppTarget cluster has dependencies on services that are in the Support and Messaging Cluster, so do not allow the AppTarget cluster to be “Open for e-business” before the Support and Messaging cluster has completed initialization.

### 4.3.4 InstanceNotFoundException running eventbucket.sh

**Problem:** When we suffered the DuplicateGlobalInstanceIdException problem described in Section 4.4.1, “CEIEM0025E, CEIES0010E, -253 SQLCODEN253 with DuplicateGlobalInstanceIdException” on page 326 we decided to run the eventbucket.sh to check the status of the event buckets with a view to changing the buckets and then purging the one causing the problem.

When invoking eventbucket.sh we suffered the problem shown in Example 4-18.

*Example 4-18 Error when invoking eventbucket.sh*

---

```
WASX7017E: Exception received while running file
"/wasv61config/mdcell/mddmnode/DeploymentManager/util/event/eventbucket.jacl";
exception information: javax.management.InstanceNotFoundException: *:*
```

---

**Cause:** This error occurs because eventbucket.sh is executed from one of the node's /bin directory and not from the Deployment Manager's /bin directory as stated in the Information Center. The error has been reported and will be corrected in a future update of the Information Center.

**Solution:** Change to the /bin directory of one of the node hosting CEI before executing eventbucket.sh.

### 4.3.5 Various issues purging the CEI event bucket

**Problem:** The instructions that describe how to purge an event bucket require more detail. See the Information Center article *Purging events from the event database*, available from the following Web page:

[http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.websphere.cei.z.612.doc/doc/tcei\\_db\\_fast\\_purge.html](http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.websphere.cei.z.612.doc/doc/tcei_db_fast_purge.html)

**Solution:** The following instructions help you understand what you need to do. The headings correspond to topics with the same title in the Information Center. This information expands on the information in the Information Center.

#### **Viewing or changing the event database active bucket status**

The log in Example 4-19 on page 324 and Example 4-20 on page 324 shows a successful invocation of eventbucket.sh to change to the other bucket and then view the status to confirm the other bucket is being used. The commands we input are highlighted in bold.

*Example 4-19 Invoking eventbucket.sh to change the CEI event bucket from 00 to 01*

---

```
: >su - mdadmin
FSUM5019 Enter the password for mdadmin:
: > /SC42/var/WebSphere/home/MDCFG> \
cd /wasv61config/mdcell/mdnodea/AppServer/bin
: >/wasv61config/mdcell/mdnodea/AppServer/bin> \
eventbucket.sh -change -user MDADMIN -password MDADMIN
WASX7209I: Connected to process "dmgr" on node mddmnode using SOAP connector;
The type of process is: DeploymentManager
WASX7303I: The following options are passed to the scripting environment and
are available as arguments that are stored in the argv variable: "[-change]"
CEIDS0042I The current bucket is being changed from bucket 0 to bucket 1.
```

---

Now check the bucket status as shown in Example 4-20.

*Example 4-20 Invoking eventbucket.sh to change the CEI event bucket from 00 to 01: Bucket status*

---

```
: >/wasv61config/mdcell/mdnodea/AppServer/bin> \
eventbucket.sh -status -user MDADMIN -password MDADMIN
WASX7209I: Connected to process "dmgr" on node mddmnode using SOAP connector;
The type of process is: DeploymentManager
WASX7303I: The following options are passed to the scripting environment and
are available as arguments that are stored in the argv variable: "[-status]"
CEIDS0054I Number of buckets: 2
Current bucket: 1
Bucket check interval (seconds): 300
```

---

## **Purging the inactive bucket for a DB2 event database**

The path to the files fastpurge00.ctl and fastpurge01.ctl is incorrect. It is the Deployment Manager's profile path:

<profile\_path>/databases/event/<cluster\_name>/dbscripts/db2zos/dd1

We found that the instructions in this section to generate the JCL and to execute the DB2 LOAD utility worked properly. However, before running the JCL to run the purge we had to make the following additional changes to the job.

- Add the JCL additions in Example 4-21 after the job card.

*Example 4-21 JCL additions*

---

```
/*JOBPARM SYSAFF=SC42
//JCLLIB   JCLLIB ORDER=DB9FU.PROCLIB < Where DSNUPROC is
//JOBLIB   DD DISP=SHR,DSN=DB9F9.SDSNEXIT
//         DD DISP=SHR,DSN=DB9F9.SDSNLOAD
```

---

- ▶ Change SYSTEM=D9FG on the //UTIL EXEC card to reference the specific DB2 subsystem ID. This may not be necessary on your system but we had to code the specific SSID of the DB2 on the LPAR where we ran the job. For that reason we also coded the /\*JOBPARM SYSAFF=SC42 card after the job card so that the job ran on the LPAR that hosted that DB2 instance.
- ▶ Change all the table names in the SYSIN so they are fully qualified names with the correct schema as qualifier. We made the following global changes:
 

```
C 'FROM CEI' 'FROM MDWPS.CEI' ALL
C 'TABLE CEI' 'TABLE MDWPS.CEI' ALL
```
- ▶ Change the database name on the CHECKDATA TABLESPACE commands from the location name to the CEI database name in DB2 for z/OS as follows:
 

```
C 'DB9F.' 'MDEVTDDB.' ALL
```

Our job ran successfully after making these changes.

The JCL we used is included in the additional material that accompanies this Redbooks publication as member MDPRG00 in the dataset WASCFG.MDCELL.WPS.CNTL. See Appendix A, “Additional material” on page 351.

After changing to event bucket 01 and purging events in event bucket 00 you are ready to change buckets from 01 back to 00 when you need to.

## 4.4 Errors relating to BPC Event Collector and Observer

The most common error experienced when using BPC Observer is that there are no events displayed in BPC Observer.

There are a number of possible causes for this. The first step is to check the logs and see if there are errors related to CEI, the Event Collector, or BPC Observer. Also the trace option com.ibm.bpe.observer.\* is useful when diagnosing problems.

The WebSphere Process Server for z/OS Information Center article *Troubleshooting Business Process Choreographer Observer* is a good place to start with diagnosing the problem. See the following Web page to access the article:

[http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.websphere.bpc.z.612.doc/doc/bpc/r7observer\\_troubleshooting.html](http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.websphere.bpc.z.612.doc/doc/bpc/r7observer_troubleshooting.html)

#### 4.4.1 CEIEM0025E, CEIES0010E, -253 SQLCODEN253 with DuplicateGlobalInstanceIdException

**Problem:** When we first invoked BPC Observer there were no events displayed.

**Cause:** We were trying to use the bpcivt to initiate some business process instances, expecting the trace WBILocationMonitor.CEI.bpc.\* to cause process-related events to be emitted. However we received an error on the BPC Explorer and found that the errors shown in Example 4-22 had been issued in the servant log.

*Example 4-22 Errors running bpcivt with WBILocationMonitor.CEI.bpc.\* trace active*

---

```
CEIEM0025E The emitter failed to send the events to the event server.  
The local event bus enterprise bean on the event server failed during event  
processing.  
Transaction mode: SAME  
Exception: com.ibm.events.datastore.DuplicateGlobalInstanceIdException  
.  
Trace: 2008/09/20 03:15:35.374 01 t=7C1350 c=UNK key=P8 (13007002)  
ThreadId: 00000045  
FunctionName: datastoreCreateEvents  
SourceId: com.ibm.events.distribution.impl.EventDistribution  
Category: SEVERE  
ExtendedMessage: CEIES0010E The event server failed to store any events in  
the data store referenced by the JNDI name because the configured data store  
threw an exception.  
JNDI name: ejb/com/ibm/events/datastore/impl/DefaultDataStoreEJBLocalHome  
Exception message: CEIDS0036E The relational database management system  
reported the following error.  
Data source resource reference: java:comp/env/EventDataSourceReference  
Database product: DB2  
Database version: DSN09015  
SQL state: null  
Vendor code: -99,999  
Message: [ibm][db2][jcc][102][10040] Non-atomic batch failure.  
The batch was submitted, but at least one exception occurred on an individual  
member of the batch.  
Use getNextException() to retrieve the exceptions for specific batched  
elements.  
Vendor Code: -253 SQL State: 22529  
Message: [IBM/DB2][T2zos/3.6.67]T2zosPreparedStatement.readExecuteBatch_  
nativeExecuteMultiRowInsert:5678:  
Extended Diagnostics Condition #1: DB2 engine SQL error,  
SQLCODE = -253, SQLSTATE = 22529, error tokens = INSERT  
A NON-ATOMIC INSERT STATEMENT SUCCESSFULLY COMPLETED FOR SOME OF THE REQUESTED  
ROWS, POSSIBLY WITH WARNINGS, AND ONE OR MORE ERRORS
```

---



We found that this error disappeared, and the bpcivt process instance started normally, if we disabled the trace WBILocationMonitor.CEI.bpc.\*.

**Solution:** We stopped using bpcivt and used the sample Fabric application (ITSOApp) to create process instances that generated events. When the trace WBILocationMonitor.CEI.bpc.\* was active, the ITSOApp did not suffer a DuplicateGlobalInstanceIdException.

#### 4.4.2 No events displayed in BPC Observer; no errors in logs

**Problem:** We checked the logs for errors and everything mentioned in the Information Center, but there were still no events showing in BPC Observer.

**Cause:** In our case we believe the cause of the problem was that BPC had been configured before CEI. Although the BPC Configuration wizard allows you to enable CEI logging in the State Observer section, it appears that CEI logging is not enabled if CEI is not already installed before you run the BPC configuration wizard.

**Solution:** We uninstalled and re-installed the Event Collector using the setupEventCollector.sh. The process for running setupEventCollector.sh is described in Section 2.19.1, “Change Event Collector thresholds” on page 177. After launching setupEventCollector, select **Remove the Event Collector application and related objects**, as shown in Example 4-23.

*Example 4-23 Removing the Event Collector and its resources*

---

Commands Menu

- 1) Prepare a database for the Event Collector and Observer
- 2) Install the Event Collector application
- 3) Remove the Event Collector application and related objects**
- 4) Change configuration settings of an installed Event Collector
- 5) Drop the database schema of the Event Collector and Observer
- 6) Administer Observer related user-defined functions

0) Exit Menu

Your selection: **3**

==> Remove the Event Collector application and related objects

---

While re-installing the Event Collector the shell script issued the messages in Example 4-24 on page 328. These showed that CEI event logging was not enabled. This was despite the fact that the boxes were checked under State Observers within the BPC Configuration on the Integrated Solutions Console.

#### *Example 4-24 Messages warning that CEI logging is not enabled*

---

Starting install ...

WebSphere Business Process Choreographer Event Collector installed successfully!

Checking if CEI event logging is enabled ...

WARNING: The business process container of mdc101\_WPS has CEI event logging disabled.

To allow the Event Collector to work correctly, CEI event logging is required.

Do you want to enable the CEI event logging on mdc101\_WPS?

y) yes

n) no

c) cancel

Your selection: [y] ☒

=> yes

Note: To activate the changes, you must restart the application server mdc101\_WPS.

---

After the re-installation completed, we stopped and started the cluster and re-tested the TestLoanProcess using the BPC Explorer. When we logged onto BPC Observer, we found that events had been captured by the Event Collector and made available to BPC Observer.

## **4.5 Errors relating to Business Space**

Most of the issues we faced configuring Business Space related to the fact that it was using a fixed schema of IBMBUSSP at runtime. This will be resolved by a future PTF.

### **4.5.1 Error configuring Business Space**

**Problem:** When using the wizard on the Integrated Solutions Console to configure Business Space, we initially tried to configure it using a schema of our choice. We later found that we had to configure Business Space using a schema of IBMBUSSP. When running the wizard for a second time, we received the error shown in Figure 4-3 on page 329.

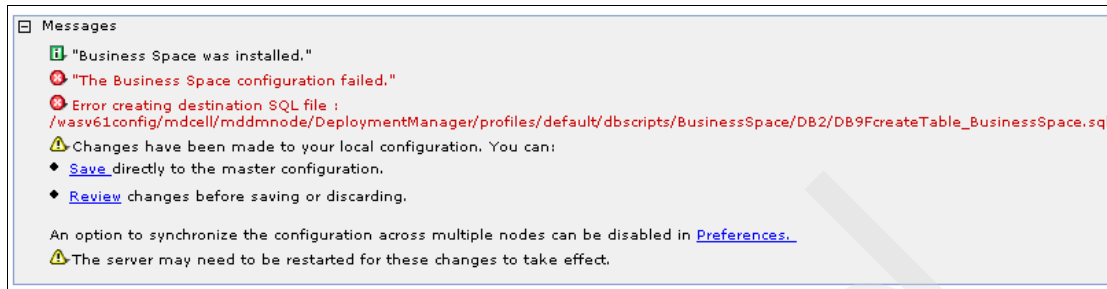


Figure 4-3 Error running the Business Space configuration wizard for a second time

**Cause:** Missing '/' in the directory path of the generated DDL when trying to re-use existing directories.

**Solution:** If you need to run the Business Space configuration wizard for a second time, first delete the directory <dmgr\_home>/profiles/default/dbscripts/**BusinessSpace** and all its sub-directories and contents. This will allow the wizard to finish without errors.

**Note:** If you receive an error like this, be sure to click **Review** and then **Discard** the changes. Do not be tempted to think that the only aspect of the wizard that has failed is the re-writing of the DDL.

We found that if we ran the wizard for a second time and ignored the error above, Business Space did not function normally. We had to uninstall the applications BusinessSpaceManager, IBM\_ BSPACE\_WIDGETS, and Business RulesManager, and then re-create the database before re-running the wizard. We also recommend that you always create the data source first before running the Business Space configuration wizard.

## 4.5.2 -204 SQLCODEN204 starting Business Space applications

**Problem:** When the Business Space applications were started for the first time we received the errors shown in Example 4-25.

*Example 4-25 Business Space application errors*

---

```
ManagedConnectionImpl@47e047e from resource jdbc/bpm/BusinessSpace.  
Caught exception: com.ibm.ws.exception.WsException:  
DSRA0080E: An exception was received by the Data Store Adapter.  
See original exception message: Cannot call 'cleanup'  
.  
.  
com.ibm.db2.jcc.t2zos.cb: [IBM/DB2][T2zos/3.6.67]  
T2zosPreparedStatement.readDescribeInput_:nativeDescribeInput:2211:  
DB2 engine SQL error, SQLCODE = -516, SQLSTATE = 26501, error tokens =  
.  
.  
ExtendedMessage: <Null Message>com.ibm.db2.jcc.t2zos.cb:  
[IBM/DB2][T2zos/3.6.67]T2zosPreparedStatement.readPrepareDescribeOutput_  
_:  
nativePrepareInto:1512: DB2 engine SQL error, SQLCODE = -204, SQLSTATE  
= 42704,  
error tokens = IBMBUSSP.REGISTRY_FILE
```

---

A -204 SQLCODEN204 error is 'resource not available'.

Notice that the schema on the table in the message is IBMBUSSP, which is not the schema we had used to create the tables.

**Cause:** Neither the currentSchema or currentSQLID properties were set when the data source called Business Space Datasource was created by the Business Space wizard. However, the error remained even after setting these properties. The schema name of IBMBUSSP is hard-coded in Business Space.

**Solution:** Create the Business Space tables using the schema IBMBUSSP until a fix is available that allows a different schema to be used.

The fix that allows a custom schema to be used is in WebSphere Process Server for z/OS V6.1.2.1.

### 4.5.3 Webgroup not found when using Business Space

**Problem 1:** After configuring Business Space, we launched it through a browser and created a Business Space. However, in the Create Tasks tab we received an error loading the page and in the servant log there was the error shown in Example 4-26.

*Example 4-26 Servant log error*

---

```
Trace: 2008/08/14 15:21:56.346 01 t=7C14C0 c=UNK key=P8 (0000000A)
Description: Log Java Message
Message: BB000220E: SRVE0255E: A WebGroup/Virtual Host to handle
/iWidget/widget/widgets/htm/taskStatusList_iwidget.xml has not been
defined.
```

---

**Cause 1:** The WebSphere Process Server for z/OS Information Center is slightly confused in the *Enabling widgets in Business Space* article about which Endpoint files need to be updated with the hostname:port of the system. In one place it says to copy and edit wpsWidgets.xml and bpcEndpoints.xml file while in another it says to copy wpsEndpoints.xml and bpcEndpoints.xml.

The problem is caused if you copy and edit wpsWidgets.xml instead of wpsEndpoints.xml.

**Solution 1:** Copy and edit the files wpsEndpoints.xml and bpcEndpoints.xml to add the hostname:port to the <tns:id> tags but do not copy and edit the file wpsWidgets.xml.

**Problem 2:** A problem occurred when trying to add the My Tasks widget to a new page in Business Space before we had added the Business Rules Rest API. The errors shown in Example 4-27 were issued in the servant log.

*Example 4-27 Servant log errors*

---

```
Trace: 2008/09/04 11:48:34.783 01 t=7C66F8 c=UNK key=P8 (0000000A)
Description: Log Java Message
Message: BB000220E: SRVE0255E: A WebGroup/Virtual Host to handle
/rest/bpm/htm/v1/tasks/filter
has not been defined. com.ibm.ws.webcontainer.WebContainer
com.ibm.ws.webcontainer.WebContainer
```

---

**Cause 2:** Business Rules not installed.

**Solution 2:** Install the Business Rules before testing Business Space.

**Cause 3:** In the two-cluster topology, the bpcEndpoints.xml file must be customized for the hostname:port of the cluster running the Business Process Container. If you specify the port of the Support and Messaging cluster it will fail.

**Solution 3:** Specify the correct hostname:port of the cluster hosting BPC in the bpcEndpoints.properties files.

#### 4.5.4 JSPG0235E: The JSP container failed to load the TagExtraInfo

**Problem:** During initialization after configuring Business Space, many occurrences of the error in Example 4-28 were seen in the servant.

*Example 4-28 Servant errors*

---

```
Trace: 2008/08/21 20:04:24.494 01 t=7C4658 c=UNK key=P8 (13007002)
  ThreadId: 00000024
  FunctionName: endElement
  SourceId: com.ibm.ws.jsp
  Category: WARNING
  ExtendedMessage: JSPG0235E: The JSP container failed to load the
TagExtraInfo
Trace: 2008/08/21 20:04:24.494 01 t=7C4658 c=UNK key=P8 (0000000A)
  Description: Log Java Message
  Message: JSPG0235E: The JSP container failed to load the TagExtraInfo
class [com.alphablox.server.tags.framework.BloxTagExtraInfo]
```

---

**Cause:** Some classes in the WebSphere Business Space widgets enterprise archive (EAR) file require that the WebSphere Business Monitor Dashboard Alphablox Web application be installed. See the following Web page for more information:

<http://www.ibm.com/support/docview.wss?uid=swg21314482>

**Solution:** You can safely disable the Web application (WBMDashboardABX) that tries to load the JSP™ Taglib classes when WebSphere Business Monitor is not installed. Perform the following steps to disable the Web application.

1. Log in to the Integrated Solutions Console of WebSphere Process Server.
2. Navigate to the **Business Space widgets** application and open the target mapping configuration of the Dashboard application (WebSphere Business Monitor): **Applications** → **Enterprise Applications** → **IBM\_BSPACE\_WIDGETS** → **Manage Modules**.
3. Select DashboardABX check box, which is the war file WBMDashboardABX.war, and click **Target specific application status**.

4. Select the target server or cluster to which the application is mapped, and click **Disable Auto Start**.
5. Click **Apply**, save the changes, and restart the server.

## 4.6 Errors related to People Directory Provider

Although the applications we tested did not need an LDAP People Directory Provider, we configured one to test and document the procedure. We experienced the error described in Section 4.6.1, “WSVR0017E and CWWBS0006E failing to bind the JNDI name of the people directory (staff plug-in configuration)” on page 333.

### 4.6.1 WSVR0017E and CWWBS0006E failing to bind the JNDI name of the people directory (staff plug-in configuration)

**Problem:** When we restarted the cluster after defining a customized LDAP People Directory Provider configuration, we found the errors shown in Example 4-29 in the servant logs.

*Example 4-29 Servant log errors*

---

```
Trace: 2008/09/23 13:13:19.526 01 t=7FF028 c=UNK key=P8 (13007002)
ThreadId: 0000000a
FunctionName: com.ibm.ws.runtime.component.ResourceMgrImpl
SourceId: com.ibm.ws.runtime.component.ResourceMgrImpl
Category: SEVERE
ExtendedMessage: BB000220E: WSVR0017E: Error encountered binding the
J2EE resource, MyLDAPPeopleDirectory, as bpe/staff/MyLDAPPeopleDirectory
from file:/wasv61config/mdcell/mdnodea/AppServer/profiles/default/config/\
cells/mdcell/nodes/mdnodea/resources-bpc.xml
com.ibm.ws.runtime.component.binder.ResourceBindingException:
invalid configuration passed to resource binding logic. REASON:
CWWBS0006E: An error occurred while configuration for the people directory
(staff plug-in configuration) "MyLDAPPeopleDirectory" was being processed.
```

---

**Cause:** In WebSphere Process Server for z/OS V6.0.x, when you placed the customized LDAP transformation xsl file in the deployment manager's <dmgr\_home>/ProcessChoreographer/Staff directory, the wizard that configured the LDAP Staff Plugin Configuration (as it was called in V6.0.x) copied the transformation xsl file to the /ProcessChoreographer/Staff directory in node. However, in WebSphere Process Server for z/OS V6.1.2 it is necessary to manually copy the customized LDAP transformation xsl file into the

/ProcessChoreographer/Staff directory of each node hosting Business Process Choreographer applications that will use that LDAP People Directory provider.

**Solution:** After customizing the LDAP transformation xsl file for your LDAP Server, copy it to the <dmgr\_home>/ProcessChoreographer/Staff directory and also to the <was\_home>/ProcessChoreographer/Staff for each node. Make sure that you set the permissions so that the file is readable by the servant region user ID.

## 4.7 Errors related to WebSphere Business Services Fabric for z/OS

This section details some of the problems we experienced when adding WebSphere Business Services Fabric for z/OS to our topology.

### 4.7.1 Database missing w\_dbversion table and -922 SQLCODEN922

**Problem:** The error in Example 4-30 was seen in the log when starting Fabric for the first time.

*Example 4-30 Log error when starting Fabric for the first time*

---

```
exception is com.webify.wsf.support.spring.dbversion.UnableToMigrateException:
database missing the
  '''w_dbversion''' table?
-- Create the database version table for migration support

create table w_dbversion (
  subsystem varchar(8) not null primary key,
  cur_version integer not null
);
.
.
The exception caught = com.ibm.db2.jcc.t2zos.cb:
[IBM/DB2] [T2zos/3.6.67] T2zosReusableConnection.flowConnect:execConnect:1407: DB2 engine
SQL error, SQLCODE = -922, SQLSTATE = 42505, error tokens = CONNECT;00D31024
SQL Error Code is -922 SQL State is = 42505
```

---

Example 4-31 on page 335 is an extract from the related FFDC log.



#### Example 4-31 FFDC log extract

---

```
Stack Dump = com.ibm.ws.exception.WsException: DSRA8100E: Unable to get a
PooledConnection from the DataSource. with SQL State : 42505 SQL Code : -922
Caused by: java.sql.SQLException:
[IBM/DB2][T2zos/3.6.67]T2zosReusableConnection.flowConnect:execConnect:1407: DB2 engine
SQL error, SQLCODE = -922, SQLSTATE = 42505, error tokens = CONNECT;00D31024DSRA0010E:
SQL State = 42505, Error Code = -922
```

---

**Cause:** The -922 with token CONNECT;00D31024 indicates a problem with the location name. The explanation of the error in the DB2 Information Center suggests that the location name entry is missing from SYSIBM.LOCATIONS but it is no longer necessary to add the location name to that table.

When we had this error it was always because the Database name field on the data source definition had been set to a database name or the DB2 group sharing name rather than the location name.

**Solution:** Make sure that the Fabric type 2 data sources have the DB2 Location name set in the Database name field.

### 4.7.2 Unable to locate MessageSource with name 'messageSource'

**Problem:** There were several instances of the following error in the servant log, as shown in Example 4-32.

#### Example 4-32 Unable to locate MessageSource with name 'messageSource' error

---

```
Trace: 2008/08/21 20:04:32.318 01 t=7C6308 c=UNK key=P8 (13007002)
ThreadId: 00000027
FunctionName: initMessageSource
SourceId:
org.springframework.web.context.support.XmlWebApplicationContext
Category: INFO
ExtendedMessage: Unable to locate MessageSource with name
'messageSource': using default
[org.springframework.context.support.DelegatingMessageSource@509c509c]
. .
Message: Unable to locate ApplicationEventMulticaster with name
'applicationEventMulticaster': using default
[org.springframework.context.event.SimpleApplicationEventMulticaster@440c440c]
```

---

**Cause:** The error is only an INFO category and can probably be ignored.

### 4.7.3 Unable to find resource 'VM\_global\_library.vm'

**Problem:** The VM\_global\_library.vm resource can not be found.

```
Trace: 2008/08/21 21:12:43.236 01 t=7C2510 c=UNK key=P8 (13007002)
  ThreadId: 00000028
  FunctionName: logVelocityMessage
  SourceId: org.apache.velocity.app.VelocityEngine
  Category: SEVERE
  ExtendedMessage: ResourceManager : unable to find resource
'VM_global_library.vm' in any resource loader.
org.apache.velocity.app.Velocity
Trace: 2008/08/21 21:12:43.237 01 t=7C2510 c=UNK key=P8 (0000000A)
  Description: Log Java Message
  Message: ResourceManager : unable to find resource 'VM_global_library.vm' in
any resource loader.
  org.apache.velocity.app.VelocityEngine logVelocityMessage
```

**Cause:** This is APAR PK68864.

**Solution:** Apply the APAR fix when available.

### 4.7.4 Error opening zip file ... soapfabric-2.3.1.jar

**Problem:** The error shown in Example 4-33 was seen in the log when starting the server for the first time after installing the Fabric core applications.

*Example 4-33 Error after starting the server after installing the Fabric core applications*

---

```
ExtendedMessage: CWLAG0201E: The Artifact Loader encountered an error while
indexing the archive file
/wasv61config/mdcell/mdnodea/AppServer/profiles/default/installedApps/mdcell/Fa
bric_Tools.ear/lib/soapfabric-2.3.1.jar.
The system returned the following error: java.util.zip.ZipException: Error
opening zip file
/wasv61config/mdcell/mdnodea/AppServer/profiles/default/installedApps/\mdcell/F
abric_Tools.ear/lib/soapfabric-2.3.1.jar.
```

---

**Cause:** Although the soapfabric-2.3.1.jar was at the directory specified when we downloaded and tried to open it using WinRAR, it gave an error saying that the jar file was invalid or corrupt.

This was probably related to an out-of-space problem that occurred in the node ZFS when the Fabric applications were first installed. We believed that the Fabric Tools application had installed correctly because we had received successful messages from the Integrated Solutions Console, but subsequently noticed that

errors had been issued on one of the servants indicating an out-of-space problem.

**Solution:** Uninstall both the Fabric Tools and the Fabric Tools Help applications.

Re-install the Fabric Tools application and then re-install the Fabric Tools Help application. Then restart both applications.

**Note:** The Fabric Tools application will take some time to initialize and produces many thousands of messages in the logs. When you have fabric working properly you may wish to change the trace setting to reduce the number of info level messages that are being written.

#### 4.7.5 NoClassDefFoundError: com.webify.wsf.spm.mdb.PerformanceMDBBean

**Problem:** The error shown in Example 4-34 was seen on the log when starting the server for the first time after installing the Fabric core applications.

*Example 4-34 Log error when starting the server after installing the Fabric applications*

---

```
Message: BB000220E: CNTR0075E: The user-provided class
"com.webify.wsf.spm.mdb.PerformanceMDBBean" needed by the
EnterpriseBean could not be found or loaded.
com.ibm.ejs.container.BeanMetaData com.ibm.ejs.container.BeanMetaData
com.ibm.ejs.container.ContainerException: Failed to initialize BeanMetaData
instance;
nested exception is: java.lang.NoClassDefFoundError: Error while defining
class:
com.webify.wsf.spm.mdb.PerformanceMDBBean
This error indicates that the class:
com.webify.wsf.model.cbe.CbeOntology$Properties
could not be located while defining the class:
com.webify.wsf.spm.mdb.PerformanceMDBBean
This is often caused by having the class at a higher point in the classloader
hierarchy
Dumping the current context classloader hierarchy:
==> indicates defining classloader
*** indicates classloader where the missing class could have been found
[0] com.ibm.ws.bootstrap.ExtClassLoader@4a684a68
[1] org.eclipse.osgi.internal.baseadaptor.DefaultClassLoader@431f431f
[2] sun.misc.Launcher$AppClassLoader@60dc60dc
[3] sun.misc.Launcher$ExtClassLoader@33a533a5
```

---

**Cause:** This was another symptom of the corrupt Fabric Tools installation.

**Solution:** This problem disappeared after re-installing the Fabric Tools application.

## 4.7.6 SAXPARSEEXCEPTION during Fabric startup

**Problem:** SAXPARSEEXCEPTION seen in log.

**Cause:** The exception does not affect functionality. This issue will be fixed in the WBSF 6.2 release.

**Solution:** The end double quote is missing from the end of the string "6.0.1 in the `spec-version` element in the file **WBSFengine.component**.

There is a copy of the WBSFengine.component file in each node and you could fix those copies, but because this is a trivial fix it is probably a good idea to fix this in the product code so you do not encounter the problem when configuring Fabric in the future. Perform the following steps to fix this situation.

1. Mount the WebSphere Business Services Fabric for z/OS product ZFS read-write.
2. Change to directory `<WBSF_install>/configuration/Runtime`.
3. In the WBSFengine.component file, add the double quote to the right-hand end of the `spec-version` so it will look like Example 4-35.

*Example 4-35 Spec-version with double quote added*

---

```
spec-version="6.0.1"/>
```

---

4. Save the change,
5. Re-mount the product ZFS read-only.

Re-copy this file to each of the configuration file system of each node.

**Note:** The WBSFengine.component file is one of the artifacts copied to each of the node's `/WPS_HOME/properties/version` directory.

Remember to re-copy this file to every node after fixing the missing quotes in the copy of the file that is in the product ZFS.

## 4.7.7 CWSIV0958E: SINotAuthorizedException for DA.Event.Topic

**Problem:** The set of errors shown in Example 4-36 were issued in the servant log with similar errors in the adjunct. This was a large message set occupying several panels, so only the first part of the messages are shown in Example 4-36.

### *Example 4-36 Servant log errors*

---

```
Trace: 2008/08/22 12:15:09.678 01 t=7B4E88 c=UNK key=P8 (13007002)
  ThreadId: 00000027
  FunctionName: com.ibm.ws.sib.utils.ras.SibMessage
  SourceId: com.ibm.ws.sib.utils.ras.SibMessage
  Category: SEVERE
  ExtendedMessage: BB000220E: [fabricbus:mdc101_WPS.000-fabricbus]
CWSIV0555E: The exception javax.resource.ResourceException:
CWSIV0958E: The authorization exception
com.ibm.wsspi.sib.core.exception.SINotAuthorizedException:
CWSIP0303E: No user specified when creating a connection to secure messaging
engine mdc101_WPS.000-fabricbus on bus fabricbus. was thrown while attempting
to create a connection to messaging engine 90162C9C51A37E50 using the
activation specification
[com.ibm.ws.sib.api.jmsra.impl.JmsJcaActivationSpecI
mpl@1721067157 <userName=null> <password=null> <xaRecoveryAlias=null>
<destination=topic://DA Event Topic?topicSpace=DA.Event.Topic
. .
. .
```

---

**Cause:** Notice that the error messages says “No user specified” and that the `userName` and `password` fields are null. This error is caused when security is enabled for the fabricbus but the Activation Specification **Hub Event Activation** does not have a Component-managed authentication alias specified.

This happens because the WebSphere Business Services Fabric for z/OS Information Center does not tell you to assign the JMS authentication alias when creating the Hub Event Activation Specification, although it does say to assign an authentication alias when creating the other Activation Specifications.

**Solution:** Assign the J2C authentication alias you created for JMS authentication to the Activation Specification **Hub Event Activation**.

Restart the cluster after correcting the activation specification.

## 4.7.8 -803 SQLCODEN803 when starting fabric applications

**Problem:** The errors in Example 4-37 were seen in the servant logs when starting the cluster for the first time after installing the Fabric applications.

*Example 4-37 Servant log error when starting the cluster after installing the Fabric applications*

---

```
Trace: 2008/08/25 19:19:24.625 01 t=7C37A0 c=UNK key=P8 (0000000A)
  Description: Log Java Message
  Message: SQLErrorCodes loaded: [DB2, HSQL, MS-SQL, MySQL, Oracle,
    Informix, PostgreSQL, Sybase]
Trace: 2008/08/25 19:19:24.721 01 t=7C37A0 c=UNK key=P8 (13007002)
  ThreadId: 00000036
  FunctionName: error
  SourceId: com.ibm.ws.fabric.ocp.ContentPackInstaller
  Category: SEVERE
  ExtendedMessage: Content import failed, rolling back to :
    2org.springframework.dao.DataIntegrityViolationException:
    PreparedStatementCallback; SQL [INSERT INTO w_version
    (id,username,change_time,cl_gid,cl_lid,schema_rev,schema_ns_id)
    VALUES (?,?,?,?,?,?,?)] [IBM/DB2] [T2zos/3.6.67]
    T2zosPreparedStatement.readExecuteInternal:nativeExecute:7657:
    DB2 engine SQL error, SQLCODE = -803, SQLSTATE = 23505,
    error tokens = WRVERSI0;000000020F;
    nested exception is com.ibm.websphere.ce.cm.DuplicateKeyException:
    [IBM/DB2] [T2zos/3.6.67]
    T2zosPreparedStatement.readExecuteInternal:nativeExecute:7657:
    DB2 engine SQL error, SQLCODE = -803, SQLSTATE = 23505,
    error tokens= WRVERSI0;000000020F
    com.ibm.websphere.ce.cm.DuplicateKeyException: [IBM/DB2] [T2zos/3.6.67]
    T2zosPreparedStatement.readExecuteInternal:nativeExecute:7657:
    DB2 engine SQL error, SQLCODE = -803, SQLSTATE = 23505, error tokens =
    WRVERSI0;000000020F
```

---

**Cause:** The underlying cause was the fact that the database J2C authentication alias user ID did not have USAGE privilege on the SEQUENCES that Fabric uses. The DuplicateKeyException appears to be a symptom of a failure to use the sequence.

**Solution:** GRANT USAGE on all the SEQUENCES to the database J2C authentication alias user ID. If you see this problem it is probably because you failed to consider issuing any grants for either tables or sequences. You may also need to GRANT ALL on the Fabric tables to the database authentication alias user ID.

## 4.7.9 Garbage on Fabric login window at first logon

**Problem:** After installing the Fabric applications and restarting the cluster, there was garbage on the login window in some of the fields. Our login window looked like Figure 4-4. If you continue to log in, you will see garbage on subsequent panels too.



Figure 4-4 Garbage on login window the first time you try to login

**Cause:** Data is loaded into the Fabric tables as part of the first start-up of the applications. It appears that this is not available to be incorporated into fields on the panels until the Fabric\_Tools application is restarted for a second time after the data has been loaded during its first start-up.

**Solution:** Stop and restart the Fabric\_Tools application and then log in again. The panels will now appear without garbage.

Restarting the cluster will also clear the problem, provided that the Fabric applications initialized correctly during the previous cluster restart.

## 4.7.10 com.ibm.ws.fabric.ocp.ContentImportException

**Problem:** When starting the Fabric applications for the first time the error in Example 4-38 was seen in the servant log.

*Example 4-38 Servant log error when starting the Fabric applications*

---

```
Trace: 2008/09/04 15:07:18.447 01 t=7C5030 c=UNK key=P8 (13007002)
ThreadId: 00000025
FunctionName: com.ibm.ws.wsewebcontainer.webapp.WebApp
SourceId: com.ibm.ws.wsewebcontainer.webapp.WebApp
Category: SEVERE
ExtendedMessage: BB000220E: Exception caught while initializing context
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'tools.bootstrapper' defined in class path resource [tools.xml]:
Initialization of bean failed; nested exception is java.lang.RuntimeException:
com.ibm.ws.fabric.ocp.ContentImportException:
Import was interrupted due to intervening catalog writes.
```

---

**Cause:** This error appears to be related to logging on to the Fabric console for the first time.

**Solution:** When the Fabric\_Tools application is restarted for a second time after its initial start, this error is no longer seen in the servant logs at startup.

## 4.7.11 fabric-da-scdl.xsd doesn't refer to a valid schema

**Problem:** The message in Example 4-39 was seen during servant initialization.

*Example 4-39 Servant initialization message*

---

```
Trace: 2008/08/28 10:47:57.698 01 t=7C7078 c=UNK key=P8 (13007002)
ThreadId: 00000010
FunctionName: com.ibm.ws.bo.BOCore
SourceId: com.ibm.ws.bo.BOCore
Category: WARNING
ExtendedMessage: The schemaLocation "scdl.xsd" defined in schema jar:file:
/wasv61config/mpcell/mpnodea/AppServer/profiles/default/\
installedApps/mpcell/Fabric_Engine.ear/lib/\
com.ibm.ws.fabric.da.scdl-8.0.jar!/model/fabric-da-scdl.xsd doesn't refer to a
valid schema, it may result in invalid BusinessObject model
```

---



There was a related FFDC log that contained the information shown in Example 4-40.

*Example 4-40 FFDC log information*

---

```
Exception = java.io.FileNotFoundException
Source = com.ibm.ws.ar.util.ArtifactURIConverter.createInputStream
probeid = 95
Stack Dump = java.io.FileNotFoundException: JAR entry model/scdl.xsd not found
in /wasv6lconfig/mpcell/mpnodea/AppServer/profiles/default/\
installedApps/mpcell/Fabric_Engine.ear/lib/com.ibm.ws.fabric.da.scdl-8.0.jar
```

---

**Solution:** This is only a WARNING message and can be ignored. The message will be removed in a future release.

### 4.7.12 CWWBF0029E: Process components of ITSOApp cannot be configured in WebSphere configuration repository

**Problem:** After successfully deploying two of the three sample applications, the deployment of ITSOApp failed with a CWWBF0029E error as shown in Example 4-41.

*Example 4-41 CWWBF0029E error*

---

```
CWWBF0029E: Process components of ITSOApp cannot be configured in
WebSphere configuration repository
```

---

**Cause 1:** There can be several possible causes of this symptom. Initially, we thought the cause was the problem described in Section 4.2.17, “CWSIJ0063E: A network connection to host name x.x.x.x, port yyyy cannot be established” on page 319. In the Deployment Manager’s servant file, there were the errors shown in Example 4-42 on page 344 during the deployment.

#### Example 4-42 Deployment Manager's servant file errors

---

```
Trace: 2008/09/04 16:26:00.807 01 t=7C4270 c=UNK key=P8 (0000000A)
  Description: Log Java Message
  Message: CWWBV0001I: Validated process model 'NewLoanProcess' successfully: 0
warnings, 0 information.
java.lang.NullPointerException
  at com.ibm.bpe.management.application.process.ProcessSCAArchive.\
addLinkReferencePropertiesToContext(ProcessSCAArchive.java:608)
  at com.ibm.bpe.management.application.process.ProcessSCAArchive.\
createBPELContext(ProcessSCAArchive.java:312)
  at com.ibm.bpe.management.application.process.ProcessSCAArchive.\
initBPELProcessContexts(ProcessSCAArchive.java:222)
  at com.ibm.bpe.management.application.process.ProcessSCAArchive.\
<init>(ProcessSCAArchive.java:134)
  at com.ibm.bpe.management.application.process.ProcessSCAArchive.\
<init>(ProcessSCAArchive.java:169)
  at com.ibm.bpe.management.application.process.ProcessSCAArchive.\
<init>(ProcessSCAArchive.java:145)
  at com.ibm.bpe.processarchive.SCDLProcessComponentConfigureTask.performTask\
(SCDLProcessComponentConfigureTask.java:129)
  at com.ibm.ws.management.application.SchedulerImpl.run(SchedulerImpl.java:262)
  at java.lang.Thread.run(Thread.java:810)
```

---

**Cause 2:** In another cell that suffered the same symptom, the fabricbus had not started in the adjunct because of the error shown in Example 4-43.

#### Example 4-43 Fabricbus not starting error

---

```
Exception: com.ibm.ws.sib.msgstore.MessageStoreRuntimeException:
CWSIS1524E: Data source, /jdbc/fabric/me, not found.
com.ibm.ws.sib.utils.ras.SibMessage
```

---

Notice the JNDI name in the message is `/jdbc/fabric/me` (with the leading '/') instead of `jdbc/fabric/me`. This was a typing error when configuring the fabricbus.

**Cause 3:** On another occasion the cause was a problem in the deployment manager. The deployment manager's ZFS could not extend because the volume it was on was full.

**Solution 1:** After generating a personal certificate for the servant region user ID and restarting the cluster, it was possible to deploy ITSOApp without errors.

**Solution 2:** Check that the fabricbus is started. If not, resolve any problems and restart the Fabric cluster (`mdcl01_WPS` in our scenario). Ensure that the fabricbus starts before trying to install ITSOApp.

**Solution 3:** Ensure there is space on the volumes holding the configuration ZFS datasets to allow the datasets to extend. We cleaned out all FFDC logs and restarted the deployment manager before installing ITSOApp successfully.

### 4.7.13 CWWBU0001E in BPC Explorer with TestLoanProcess

The error shown in Example 4-44 was received when submitting a TestLoanProcess through the BPC Explorer.

*Example 4-44 Error when submitting TestLoanProcess through the BPC Explorer*

---

CWWBU0001E: A communication error occurred when the 'class com.ibm.bpe.clientmodel.command.CallCommand' function was called.

---

This symptom can have many causes. It is important to look in the logs for additional errors and in any FFDC logs that are produced. Here we describe some of the issues we experienced.

**Problem 1:** The error shown in Example 4-45 was seen in the servant log after receiving CWWBU0001E when testing the TestLoanprocess.

*Example 4-45 Servant log error after receiving CWWBU0001E when testing the TestLoanprocess*

---

```
Trace: 2008/09/05 03:41:39.995 01 t=7BF188 c=UNK key=P8 (13007002)
  ThreadId: 0000003c
  FunctionName:
com.ibm.ws.sca.internal.webservice.handler.PortHandler.processMessage#01
  SourceId: SCA.FFDC
  Category: ALL
  ExtendedMessage: Exception: WebServicesFault
  faultCode: {http://schemas.xmlsoap.org/soap/envelope/}Server.generalException
  faultString: WWS3713E: Connection to the remote host itsodmgr failed.
  Received the following error: null
  faultActor: null
  faultDetail:
```

---

**Cause 1:** Notice that the remote host in the message is itsodmgr, which is the host name coded in the application. The SCA bindings had not been updated in the ITSOApp to match the host name of our WebSphere Process Server for z/OS cluster.

**Solution 1:** Update the SCA bindings as described in Section 3.6.9, “Update the SCA Import URLs” on page 298.

**Problem 2:** On another occasion the error shown in Example 4-46 on page 346 was seen in the servant log after receiving CWWBU0001E when testing the TestLoanprocess.

*Example 4-46 Servant log error after receiving CWWBU0001E when testing the TestLoanprocess*

---

```
Trace: 2008/08/29 12:36:10.541 01 t=7B87B8 c=UNK key=P8 (13007002)
  ThreadId: 00000045
  FunctionName:
com.ibm.ws.sca.internal.webservice.handler.PortHandler.processMessage#01
  SourceId: SCA.FFDC
  Category: ALL
  ExtendedMessage: Exception: WebServicesFault
  faultCode: {http://schemas.xmlsoap.org/soap/envelope/}Server.generalException
  faultString:
WSWS3713E: Connection to the remote host wpsplex.itso.ibm.com failed.
  Received the following error:
EDC8128I Connection refused. (errno2=0x74940000)
```

---

The message contained the correct host name, so the problem must have been with the port, but the message did not report the port number.

At the same time as this error there was an FFDC log written. In this FFDC log, in the transport section, there was more information (Example 4-47) about the remote host that was causing the problem.

*Example 4-47 Remote host of transport section in FFDC log*

---

```
transport =
  DEFAULT_TRANSPORT_NAME = http
  sessionContext = null
  transportName = http
  url = http://wpsplex.itso.ibm.com:22048/fabric-catalog/services/subscriberManager
```

---

**Cause 2:** In this case we had updated the SCA bindings as described in Section 3.6.9, “Update the SCA Import URLs” on page 298 but had coded the HTTP port number of the wrong cluster. The HTTP port number should have been 20048 for the cluster we were testing in, not 22048.

**Solution 2:** Check all the SCA binds for the correct host name and port.

Check especially for any blank characters that might have been accidentally inserted into the url when using copy/paste.

If you find an incorrect url here, it would be a good idea to re-check the host name and port you specified in the **wpsfEndpoint.xml** when you customized it in 3.3.1, “Copy the Fabric artifacts” on page 231.

**Problem 3:** After correcting the SCA binding there was an error (Example 4-48 on page 347) in the logs.

#### Example 4-48 Log error after correcting the SCA binding

---

```
ExtendedMessage: Assembly Failure:
<soapenv:Body xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Fault><faultcode>soapenv:Server.generalException</faultcode>
<faultstring>WSWS3713E: Connection to the remote host wpsplex.itso.ibm.com failed.
Received the following error: EDC8128I Connection refused.
(errno2=0x74940000)</faultstring></soapenv:Fault></soapenv:Body>
(http://wpsplex.itso.ibm.com:22048/ITSO_implWeb/sca/PremiumLoanProviderExport)
```

---

**Cause 3:** Notice the port 22048 in this message. This URL is coming from the application file `http_www.ibm.com_vehicleloan_inst.owl`, which is part of the `ITSOBankCBAPrj20080804-owl.zip`. The correct port was 20048 for the cluster we were testing.

**Solution 3:** When testing with the sample `ITSOApp`, there are some hard-coded URLs in the application which must be updated to match the hostname:port of the fabric cluster with which you are testing.

After updating the URLs in file `http_www.ibm.com_vehicleloan_inst.owl`, re-insert the file into the zip file `ITSOBankCBAPrj20080804-owl.zip`, and re-import that zip file into Fabric. Then re-do the Subscriber Manager subscriptions. Otherwise, you will encounter the problem described in Problem 5.

**Problem 4:** The logs contained the ‘EDC8128I Connection refused’ message but in addition there was the error shown in Example 4-49.

#### Example 4-49 Incorrect security settings error

---

```
Trace: 2008/09/05 04:57:32.773 01 t=7BF188 c=UNK key=P8 (0000000A)
Description: Log Java Message
Message: Problem sending invocation event(s)
org.springframework.jms.JmsSecurityException:
CWSIA0006E: The authorization for the supplied user name was not successful.;
```

---

**Cause 4:** This error is caused by incorrect security settings for the fabricbus.

In our case, when setting the user ID/group in the role authorized to connect to the bus, we had chosen “group” but specified the user ID `MPJMSU` instead of the group `MPJMSG`.

**Solution 4:** Double-check the security settings on the fabric bus by navigating to **Buses → fabricbus → Security for bus fabricbus → Users and groups in the bus connector role**.

**Problem 5:** The error shown in Example 4-50 was seen in the servant log when the CWWBU0001E error was issued in the BPC Explorer.

*Example 4-50 Servant log error when the CWWBU0001E error was issued in the BPC Explorer*

---

```
Trace: 2008/08/29 14:26:42.671 01 t=7BE7D0 c=UNK key=P8 (13007002)
  ThreadId: 00000044
  FunctionName: error
  SourceId: com.ibm.ws.fabric.da.impl.DaContextManagerImpl
  Category: SEVERE
  ExtendedMessage:
com.webify.wsf.engine.context.InvalidContextException:
  The subscription 'NONE' is not fully qualified.
  Subscription ID must be the fully qualified URI.
  The easiest way to get valid IDs is with the copy to clipboard
feature in Subscription Manager
```

---

**Cause 5:** Notice that the message is saying that the subscription is NONE, which suggests that the user has no subscriptions.

We had coded an incorrect port number in the file  
http\_www.ibm.com\_vehicleloan\_inst.owl the first time we customized it for the  
HTTP hostname:port of our WebSphere Business Services Fabric for z/OS  
cluster.

After correcting this and re-importing the FCA zip file,  
ITSOBankCBAPrj20080804-owl.zip, we found that the user ID MPFABADM had  
lost its subscriptions.

**Solution 5:** After re-importing FCA files, re-assign the user ID's subscriptions  
again as described in "Subscribe the user ID to ITSOBankLoan App" on  
page 294.

**Note:** After correcting any incorrect SCA bindings or re-importing the  
ITSOBankCBAPrj20080804-owl.zip, stop and restart the related applications,  
ITSOApp, ITSO\_implApp and WebSphereEnvUtilApp.

## 4.7.14 NMSV0610I: A NamingException for wbsf-cbe-emitter-factory

**Problem:** The error shown in Example 4-51 is issued when testing the TestLocalProcess and CEI is configured in a cluster.

### *Example 4-51 JNDI name error*

---

```
Trace: 2008/09/02 09:37:46.847 01 t=7B9E88 c=UNK key=P8 (13007002)
  ThreadId: 0000004d
  FunctionName: com.ibm.ws.naming.util.Helpers
  SourceId: com.ibm.ws.naming.util.Helpers
  Category: WARNING
  ExtendedMessage: BB000221W: NMSV0610I: A NamingException is being
thrown from a javax.naming.Context implementation. Details follow:
Context implementation: com.ibm.ws.naming.jndicos.CNContextImpl
Context method: lookupExt
Context name: mdcell/nodes/mdnodea
Target name: persistent/wbsf-cbe-emitter-factory
```

---

**Cause:** The JNDI name of the CBE emitter factory is not fully qualified.

**Solution:** Create a namespace binding for wbsf-cbe-emitter-factory that maps it to the fully-qualified JNDI name of the CBE emitter factory. This is described in Section 3.4.10, “Configure a namespace variable for CEI” on page 274.

## 4.7.15 CEIEM0025E The emitter failed to send the events

**Problem:** After successfully testing the TestLoanApp we saw the error in Example 4-52 in the servant log. Notice that the error mentions the ‘Remote event bus’ because this was a two-clusters topology with ITSOApp running in the cluster mpcl02.AppTarget and the CEI running in the cluster mpcl01.WPS\_SandM.

### *Example 4-52 Servant log error after testing TestLoanApp*

---

```
Trace: 2008/09/05 05:17:58.763 01 t=7BF188 c=UNK key=P8 (0000000A)
  Description: Log Java Message
  Message: Failure to create or emit
eventcom.ibm.events.emitter.SendFailureException:
CEIEM0025E The emitter failed to send the events to the event server.
The remote event bus enterprise bean on the event server failed during
event processing.
  Transaction mode: SAME
Exception: java.rmi.AccessException
. .
. .
```

---

We had configured the namespace binding for the CEI emitter factory so that was not the cause.

Looking into the servant logs in the mpcl01.WPS\_SandM cluster we found the error shown in Example 4-53.

*Example 4-53 EJBROLE profile MP.eventCreator error*

---

```
Trace: 2008/09/05 05:17:58.205 01 t=7BEAD0 c=0.A key=P8 (13007002)
  ThreadId: 0000002b
  FunctionName: com.ibm.ws.security.core.SecurityCollaborator
  SourceId: com.ibm.ws.security.core.SecurityCollaborator
  Category: AUDIT
  ExtendedMessage: BB000222I: SECJ0053E: Authorization failed for IBM.COM/MPFABADM
while invoking (Bean)ejb/com/ibm/events/bus/EventBus
createEvent(org.eclipse.hyades.logging.events.cbe.CommonBaseEvent):1
securityName: IBM.COM/MPFABADM;accessID: user:IBM.COM/MPFABADM is not granted any
of the required roles: eventAdministrator eventCreator catalogAdministrator
```

---

After seeing this message, we looked in the syslog and found ICH408I messages that showed that MPFABADM did not have access to the EJBROLE profile MP.eventCreator.

**Cause:** The user ID we were testing ITSOApp with, MPFABADM, had not been given the eventCreator role.

**Solution:** Permit the user ID that is testing ITSOApp to the role eventCreator.

We connected the user ID MPFABADM to the RACF group MPEVTCRG because group MPEVTCRG was on the access list for the EJBROLE profile called MP.eventCreator.

**Note:** When resolving a security problem by connecting a user to a group, note that the user ID's control blocks in WebSphere Application Server for z/OS will have been built with the list-of-groups that applied at login time.

You may find that logging off and logging on again is not enough for the user ID to pick up the new group connection. Try deleting cookies in your browser to remove any existing LTPA token that will bypass the new login process. Otherwise you may have to wait some minutes for the LTPA token to expire before logging in again.



## Additional material

This book refers to additional material that can be downloaded from the Internet as described below.

### Locating the Web material

The Web material associated with this book is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser at:

<ftp://www.redbooks.ibm.com/redbooks/SG24-7703>

Alternatively, you can go to the IBM Redbooks Web site at:

[ibm.com/redbooks](http://ibm.com/redbooks)

Select the **Additional materials** and open the directory that corresponds with the IBM Redbooks form number, SG24-7703.

## Using the Web material

The additional Web material consists of the following file:

<i>File name</i>	<i>Description</i>
<b>SG247703.zip</b>	Additional material

### How to use the Web material

Create a subdirectory (folder) on your workstation, and download the **SG247703.zip** file in **BINARY**. It is about 760 KB.

Unzip the contents of the Web material zip file into this folder.

The zip file contains the following directories:

1. zos\_LDAP\_config
2. zos\_planning\_spreadsheet
3. zos\_wbsf
4. zos\_wps
5. zos\_wpswork

The contents of each of these directories is now described in more detail.

#### **zos\_LDAP\_config**

The zos\_LDAP\_config directory contains the following files:

- ▶ zos\_MD\_itso.ldif
- ▶ zos\_MD\_itso\_wbsf.ldif

You only need to consider using the file zos\_MD\_itso.ldif if you want your cell to use an LDAP user registry.

This ldif file contains sample commands that could be used to update Tivoli Directory Server for z/OS (TDS z/OS) with the user IDs and groups required to support WebSphere and WPS when the cell is using TDS z/OS as a user registry. These commands are identical to those on the WPS\_LDAP worksheet of the spreadsheet provided in the zos\_planning\_spreadsheet directory. Usage of the spreadsheet is described in Section 2.2.5, “Planning using a spreadsheet” on page 33. Choosing user registries is discussed in Section 2.2.6, “Planning security” on page 38.

The file zos\_MD\_itso\_wbsf.ldif is needed if you install WebSphere Business Services Fabric for z/OS.

Even if your cell is using a SAF-based user registry like RACF you will need to use an LDAP server with Fabric. Although Fabric can use RACF for authentication and J2EE role checking, it needs to use an LDAP user registry for various application functions. Therefore, you will need to create an LDAP server and load entries similar to those in file `zos_MD_itso_wbsf.ldif`. This is described in Section 3.6.5, “Create an External LDAP federation project” on page 286.

### **zos\_planning\_spreadsheet**

The `zos_planning_spreadsheet` directory contains the MD - zPMT WPS V6.1.2 Configuration.xls files.

The MD - zPMT WPS V6.1.2 Configuration.xls file is the spreadsheet we used to plan our single-cluster topology MD cell in Section 2.2.5, “Planning using a spreadsheet” on page 33.

You could use the same spreadsheet to plan topology with two clusters. We did not supply the spreadsheet we used for our two-cluster topology cell, MP cell, because there is no difference between the two spreadsheets apart from the cell prefix (MP instead of MD) and the port ranges.

### **zos\_wps**

The `zos_wps` directory contains the following files, which relate to configuring WebSphere Process Server for z/OS:

- ▶ WPS.CNTL.XMIT,
- ▶ WPS.SQL.XMIT
- ▶ WPS.OUTPUT.XMIT

These three files contains all the sample JCL, DDL, and shell scripts that are referred to in Chapter 2, “Configuring WebSphere Process Server for z/OS Network Deployment” on page 25.

The files are partitioned datasets that have been transmitted into a sequential dataset format using the TSO XMIT command. We explain how to upload these files to your z/OS system in “Installing the sample JCL, DDL, and shells on a z/OS system” on page 355.

The WPS.CNTL.XMIT file is the XMITed contents of the dataset WASCFG.MDCELL.WPS.CNTL. This contains all the sample JCL referred to in this Redbooks publication.

The WPS.SQL.XMIT file is the XMITed contents of the dataset WASCFG.MDCELL.WPS.SQL. This contains all the DDL that we generated and customized when configuring our MD cell.

The WPS.OUTPUT.XMITfile is the XMITed contents of the dataset WASCFG.MDCELL.WPS.OUTPUT. This contains the output from most of the jobs we ran when configuring our MD cell. There are also some members that contains failed jobs, together with an explanation of the problem and the solution. We hope that including this output will help you if you have problems executing similar jobs on your system.

### **zos\_wbsf**

The zos\_wbsf directory contains the following files, which relate to configuring WebSphere Business Services Fabric for z/OS:

- ▶ WBSF.CNTL.XMIT,
- ▶ WBSF.SQL.XMIT
- ▶ WBSF.OUTPUT.XMIT

These three files contains all the sample JCL, DDL and shell scripts that are referred to in Chapter 3, “Incorporating WebSphere Business Services Fabric into a production topology on z/OS” on page 227.

We explain how to upload these files to your z/OS system in “Installing the sample JCL, DDL, and shells on a z/OS system” on page 355.

The WBSF.CNTL.XMIT file is the XMITed contents of the dataset WASCFG.MDCELL.WBSF.CNTL. This contains all the sample JCL we created when adding Fabric to our MD cell.

The WBSF.SQL.XMIT file is the XMITed contents of the dataset WASCFG.MDCELL.WBSF.SQL. This contains all the DDL that we generated and customized when adding Fabric to our MD cell.

The WBSF.OUTPUT.XMIT file is the XMITed contents of the dataset WASCFG.MDCELL.WBSF.OUTPUT. This contains the output from most of the jobs we ran when adding Fabric to our MD cell.

#### **Notes:**

1. We do not expect you actually to use the DDL we provide in the additional material. It is better to generate your own using the procedures described in this Redbooks publication. Our DDL is provided as a reference only.
2. Some of the JCL members in WPS.CNTL and WBSF.CNTL have the same names, so be sure to unload the XMIT datasets to different datasets.

## zos\_wpswork

The zos\_wpswork directory contains the sample shell scripts that we mention in this Redbooks publication. There are also the response files we used for configuring our MD cell. The zos\_wpswork directory contains shells relating to both WebSphere Process Server for z/OS and WebSphere Business Services Fabric.

We explain how to upload these files to your z/OS system in “Installing the sample JCL, DDL, and shells on a z/OS system” on page 355.

## Installing the sample JCL, DDL, and shells on a z/OS system

To install the sample JCL, DDL and shell scripts, perform the following steps:

1. Allocate six sequential datasets on your z/OS system with suitable dataset names. For example, hlq.WPS.CNTL.XMIT.

The dataset format is RECFM=FB,LRECL=80,BLKSIZE=3120.

You could use JCL like that shown in Example 4-54.

*Example 4-54 Sample JCL to allocate the .XMIT datasets for additional material*

---

```
//MDALLOC JOB NOTIFY=&SYSUID,CLASS=A,MSGCLASS=A,REGION=0M
//*
/* ALLOCATE FILES TO RECEIVE XMIT DATASETS
/*
//ALLOC PROC DSN=,PRI=4
//IEFBR14 EXEC PGM=IEFBR14
//ALLOC DD DSN=&DSN,
// DISP=(NEW,CATLG),
// UNIT=3390,SPACE=(TRK,(&PRI.,5),RLSE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
// PEND
/*
//S1 EXEC ALLOC,DSN=WASCFG.MDCCELL.WPS.CNTL.XMIT,PRI=8
//S2 EXEC ALLOC,DSN=WASCFG.MDCCELL.WPS.SQL.XMIT,PRI=53
//S3 EXEC ALLOC,DSN=WASCFG.MDCCELL.WPS.OUTPUT.XMIT,PRI=88
//S4 EXEC ALLOC,DSN=WASCFG.MDCCELL.WBSF.CNTL.XMIT,PRI=3
//S5 EXEC ALLOC,DSN=WASCFG.MDCCELL.WBSF.SQL.XMIT,PRI=4
//S6 EXEC ALLOC,DSN=WASCFG.MDCCELL.WBSF.OUTPUT.XMIT,PRI=151
```

---

2. Transfer the .XMIT files to your z/OS system in BINARY mode, choosing dataset names on the transfer that ensures the .XMIT files go into the datasets you just allocated.
3. On TSO, use ISPF option 3.4 to list the .XMIT datasets.

4. Next to each dataset type the line command:

```
receive indsn(/)
```

Example 4-55 shows the command being used to receive the WBSF.CNTL dataset. Execute the same command against each of the datasets listed in the example.

*Example 4-55 Receiving the XMIT files using the TSO RECEIVE command*

---

DSLIST - Data Sets Matching WASCFG.MDCELL.\*.\*.XMIT  
Command ==>

Command - Enter "/" to select action

```
-----  
receive indsn(/)MDCELL.WBSF.CNTL.XMIT  
                  WASCFG.MDCELL.WBSF.OUTPUT.XMIT  
                  WASCFG.MDCELL.WBSF.SQL.XMIT  
                  WASCFG.MDCELL.WPS.CNTL.XMIT  
                  WASCFG.MDCELL.WPS.OUTPUT.XMIT  
                  WASCFG.MDCELL.WPS.SQL.XMIT
```

---

You will receive the following prompt:

Dataset WASCFG.MDCELL.WPS.CNTL from SENIOKJ on WTSCPLX1  
Enter restore parameters or 'DELETE' or 'END' +

5. Press enter to allocate a dataset under your user ID with a similar name. For example, if you are logged on as FRED and you press enter in response to the prompt, the sequential XMIT dataset is expanded into a partitioned dataset called 'FRED.MDCELL.WPS..CNTL'.

If you want to specify a completely different name, enter the DSN(...) option with the dataset you want to create, and then press enter. For example:

```
DSN('WASCFG.MDCELL.WPS.CNTL')
```

6. Create a work directory in the UNIX file system under your user ID. For example, if you are logged on as FRED, create /u/fred/wpswork, where /u/fred is your OMVS home directory.
7. Transfer the contents of the zos\_wpswork directory provided in the additional material from your workstation into a ../wpswork directory using BINARY mode.
8. Change the permission on any files in the wpswork directory to ensure that they include the execution bit. For example:  

```
> cd /u/fred/wpswork  
> chmod 755 *
```

# Abbreviations and acronyms

<b>AST</b>	Application Server Toolkit	<b>TCB</b>	Task Control Block
<b>BPC</b>	Business Process Choreographer	<b>TDS</b>	Tivoli Directory Server
<b>BPCE</b>	Business Process Choreographer Explorer	<b>zPMT</b>	zProfile Management Tool
<b>BPM</b>	Business Process Management		
<b>BRM</b>	Business Rules Manager		
<b>CBE</b>	Common Base Event		
<b>CEI</b>	Common Event Infrastructure		
<b>CMP</b>	Container Managed Persistence		
<b>DDF</b>	Distributed Data Facility		
<b>DDL</b>	Data Definition Language		
<b>DMZ</b>	De-Militarized Zone		
<b>EAR</b>	Enterprise Archive		
<b>HA</b>	High Availability		
<b>HFS</b>	Hierarchical File System		
<b>IBM</b>	International Business Machines Corporation		
<b>ISS</b>	Integrated Security Services		
<b>ITSO</b>	International Technical Support Organization		
<b>IVT</b>	installation verification test		
<b>JMS</b>	Java Message Service		
<b>LNA</b>	LDAP Native Authentication		
<b>LPA</b>	Link Pack Area		
<b>LPAR</b>	Logical Partition		
<b>ODR</b>	On-Demand Router		
<b>PSP</b>	Preventative Service Planning		
<b>SCA</b>	Service Component Architecture		
<b>SDO</b>	Service Data Object		
<b>SIB</b>	Service Integration Bus		





# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks

For information about ordering these publications, see “How to get Redbooks” on page 360. Note that some of the documents referenced here may be available in softcopy only.

- ▶ *WebSphere Business Process Management V6.1.2 Production Topologies*, SG24-7665
- ▶ *Getting Started with IBM WebSphere Business Services Fabric V6.1*, SG24-7614
- ▶ *Getting Started with IBM WebSphere Process Server and IBM WebSphere Enterprise Service Bus Part 1: Development*, SG24-7608
- ▶ *Getting Started with IBM WebSphere Process Server and IBM WebSphere Enterprise Service Bus Part 2: Scenario*, SG24-7642
- ▶ *Getting Started with IBM WebSphere Process Server and IBM WebSphere Enterprise Service Bus Part 3: Run time*, SG24-7643
- ▶ *Security in WebSphere Application Server V6.1 and J2EE 1.4 on z/OS*, SG24-7384
- ▶ *WebSphere Process Server for z/OS: Configuring a Network Deployment Environment*, REDP-4388. (This paper discusses WebSphere Process Server for z/OS V6.0.2.)

## Other publications

The following White Papers from the IBM Washington Systems Center are also relevant as further information sources and are cited in the text of this Redbooks publication. They can be obtained from the following Web page:

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

- ▶ *WebSphere z/OS V6 -- WSC Sample ND Configuration*, WP100653
- ▶ *Introducing the zPMT Configuration Tool for WebSphere z/OS*, WP100871

- ▶ *WebSphere z/OS V6.1 - A Top Down Configuration Approach*, WP101030
- ▶ *WebSphere Process Server for z/OS V.6.1 “Easy” Network Deployment Configuration*, WP101253
- ▶ *WebSphere Process Server/WebSphere Enterprise Service Bus for z/OS V6.1: Network Deployment Configuration Lab*, WP101209
- ▶ *Performing Installation Verification for WPS on z/OS V6.1*, WP101218
- ▶ *WebSphere Application Server for z/OS - Planning for Test, Production and Maintenance*, WP100396
- ▶ *WebSphere z/OS V6.1 - Making the DMGR Mobile*, WP101140
- ▶ *WebSphere z/OS - Comparing Front End HTTP Options*, PRS2663
- ▶ *WebSphere for z/OS V6.1 - 64-bit Addressing Support*, WP100920
- ▶ *The Mixed Platform Stack Project: Deploying a secure SOA solution into z/OS*, WP101300

## How to get Redbooks

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks, at this Web site:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)

**z/OS: WebSphere Business Process Management V6.1.2 Production Topologies**

(0.5" spine)  
0.475" <-> 0.875"  
250 <-> 459 pages







# z/OS: WebSphere Business Process Management V6.1.2 Production Topologies

**Configuring WebSphere Process Server for z/OS in a clustered topology**

**Incorporating WebSphere Business Services Fabric for z/OS**

**Troubleshooting the production topology**

This IBM Redbooks publication describes how to choose and configure a production topology for WebSphere Process Server for z/OS and WebSphere Business Services Fabric for z/OS. This paper book will be useful to infrastructure architects and systems programmers. Because WebSphere Process Server for z/OS interfaces with DB2® for z/OS and with a security manager such as RACF, this book is also relevant to security administrators and to database administrators.

Chapter 1 discusses BPM production topologies for z/OS.

Chapter 2 contains step-by-step instructions for configuring both a single-cluster topology and a two-cluster topology using WebSphere Process Server V6.1.2 for z/OS.

Chapter 3 describes how to add WebSphere Business Services Fabric V6.1.2 for z/OS to your production topology, and how to verify the installation.

Chapter 4 documents errors encountered creating the Business Process Management production topology on z/OS, and describes how to overcome these errors.

This book provides additional material, including the spreadsheet used to plan the cell's topology, sample JCL, and the shell scripts used to simplify some steps.

**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](http://ibm.com/redbooks)