



IBM Software Group

WebSphere® Enterprise Service Bus for z/OS® V6.1 WebSphere Process Server for z/OS V6.1

z/OS DB2® configuration



@business on demand.

© 2008 IBM Corporation
Updated February 13, 2008

This presentation will cover the DB2 configuration for WebSphere Process Server for z/OS V6.1 and WebSphere Enterprise Service Bus for z/OS V6.1

Goals

- Describe DB2 requirements for WebSphere Process Server for z/OS V6.1 and WebSphere Enterprise Service Bus for z/OS V6.1
- Describe options for database configuration

The goal of this presentation is to familiarize you with the DB2 requirements to run WebSphere Process Server or WebSphere Enterprise Service Bus. The presentation will look at the required DB2 configuration and options on how to configure it.

DB2 requirements

- DB2 Universal Database Server for z/OS, Version 7 or higher
 - ▶ Need stored procedure builder enabled (DSNTPSMP) if using Relationships
 - ▶ DB2 Universal Database Server for z/OS, Version 8 or higher needed for Business Rules and Selectors
 - New function mode!

3

z/OS DB2 configuration

© 2008 IBM Corporation

In order to implement WebSphere Process Server for z/OS V6.1 or WebSphere Enterprise Service Bus for z/OS V6.1, you must be using DB2 Universal Database Server for z/OS Version 7 or higher. It is possible to use Version 7 (or Version 8 in compatibility mode) but you will not be able to use Business Rules or Selectors. Those functions both require a minimum of DB2 Version 8 running in new function mode. No matter what version of DB2 you are running, if you plan to use Relationships, you will also need to have the stored procedure builder enabled (DSNTPSMP).

Databases (default names)

- WPRCSDB – common database used by:
 - ▶ Relationship services
 - ▶ Mediation
 - ▶ Recovery
 - ▶ Business rules
 - ▶ Selector
 - ▶ Scheduler
 - ▶ ESB mediation events
- EVENT and EVENTCAT – used by Common Event Infrastructure
- BPEDB – Used by Business Process Container and Human Task Manager

No separate
ESBDB



The next couple of slides describe the databases needed by the WebSphere Process Server and WebSphere Enterprise Service Bus. This slide shows the main databases needed with the default names used. While there are four databases shown here, it is important to note that you really only need to configure the databases for the functions you plan to use. The WebSphere Process Server database, or WPRCSDB, is the common database that is the only 'required' database. It is used by many of the functions introduced with the WebSphere Process Server and the WebSphere Enterprise Service Bus products such as Mediation, Relationship Services and Business Rules. The Common Event Infrastructure, or CEI, requires two databases, EVENT and EVENTCAT, but if you do not plan on using CEI, it is not necessary to configure them. ESB mediation events used to have a separate database but with Version 6.1, those tables have been moved into the common database as shown on the slide. The BPEDB is one used by the Business Process Container and the Human Task Manager, so again, it is only needed if you plan to configure those components and use them in the server. Note that WebSphere Enterprise Service Bus only uses the WPRCSDB and EVENT databases.

Databases – System integration bus

- SIBAPP -- used by the SCA.APPLICATION.<cell_name>. Bus messaging engine which is used to define JMS queue destinations and other JMS resources for modules deployed with JMS bindings
- SIBSCA -- used by SCA.SYSTEM.<cell_name>. Bus which is used by the SCA runtime to support asynchronous interactions between components and modules.
- SIBCEI -- used by CommonEventInfrastructure_Bus messaging engine which is used by the CEI for asynchronous event transport
- SIBBPC – used by BPC.<cell_name>. Bus messaging engine

5

z/OS DB2 configuration

© 2008 IBM Corporation

There are also four service integration bus databases that might be used by the WebSphere Process Server or the WebSphere Enterprise Service Bus. Service integration buses support the deployment of service applications and mediation modules in the WebSphere Process Server and WebSphere Enterprise Service Bus environments. The first two databases shown on the slide here, SIBAPP and SIBSCA, are used by the Service Component Architecture so they are required in both environments. The other two databases shown, SIBCEI and SIBBPC, are needed only if you plan to configure CEI and Business Process Choreographer functions.

DB2 databases – Naming considerations

Database	Dbase Name	Storage Group	Schema (owner)	VCAT DSN-hlq
WPS	xxWPSDB	xxWPSSTO	xxCELL	xxWPS
BPE	xxBPEDB	xxBPESTO	xxCELL	xxWPS
SIBs	xxSIBSDB	xxSIBSTO	xxSIBS	xxWPS
	xxSIBADB	xxSIBSTO	xxSIBS	xxWPS
	xxSIBCDB	xxSIBSTO	xxSIBC	xxWPS
	xxSIBBDB	xxSIBSTO	xxSIBB	xxWPS
CEI	xxEVTDB	xxEVTSTO	xxCELL	xxWPS
	xxEVTCAT	xxEVTSTO	xxCELL	xxWPS

where xx=cell identifier (for instance, SCELL)

- One possibility is to use separate databases
 - SIB databases have common tables/tablespaces so need unique schema

6

One of the first things you need to do when configuring the WebSphere Process Server or WebSphere Enterprise Service Bus is to figure out a naming convention for the many databases needed. This slide shows one option where unique databases are used for each of the various components but a common schema name is used. One exception to the common schema name is the schema name used by the databases for the four possible Service Integration Buses. Since the Service Integration Bus databases use common table names, a unique schema name is needed for each bus to distinguish the tables. The convention shown here assumes a 2-letter cell identifier has been used for the WebSphere Application Server cell and that is pre-pended to each of the database component names. You'll note the unique schema names for the Service Integration Bus databases.

DB2 databases – Naming considerations

Database	Dbase Name	Storage Group	Schema (owner)	VCAT DSN-hlq
WPS	xxCELL	xxDBSTO	xxCELL	xxWPS
BPE	xxCELL	xxDBSTO	xxCELL	xxWPS
SIBs	xxCELL	xxDBSTO	xyyyS	xxWPS
	xxCELL	xxDBSTO	xyyyA	xxWPS
	xxCELL	xxDBSTO	xyyyB	xxWPS
	xxCELL	xxDBSTO	xyyyC	xxWPS
CEI	xxCELL	xxDBSTO	xxCELL	xxWPS
	xxCELL	xxDBSTO	xxCELL	xxWPS

where xx=cell identifier (for instance, SScELL), yy=scope (for instance, S1)

- Another possibility is to use one database/storage group/schema name
 - SIB databases have common table names so need unique schema

7

Another option is to use a single database for all the components with a common storage group and schema name. You will find this scheme used, by default, when you look at the createDB sample script that is shipped with the product on some later slides. The createDB sample script can be used to easily generate SQL for the DB2 administrator to run. Again, since the Service Integration Buses use common table names, they need to be distinguished with unique schema names. The cell identifier and scope identifier is appended with a letter denoting the particular service integration bus it is being used for. For instance, 'C' is used to denote the service integration bus used for CEI.

DB2 databases – Naming considerations

Database	Scope	Dbase Name	Storage Group	Schema (owner)	VCAT DSN-hlq
WPS	Cell	xxCELL	xxDBSTO	xxCELL	xxWPS
BPE	Server/Cluster	xxCELL	xxDBSTO	xxSRyyz	xxWPS
SIBs	Server/Cluster	xxCELL	xxDBSTO	xxSRyyzS	xxWPS
	Server/Cluster	xxCELL	xxDBSTO	xxSRyyzA	xxWPS
	Server/Cluster	xxCELL	xxDBSTO	xxSRyyzB	xxWPS
	Server/Cluster	xxCELL	xxDBSTO	xxSRyyzC	xxWPS
CEI	Server/Cluster	xxCELL	xxDBSTO	xxSRyyz	xxWPS
		xxCELL	xxDBSTO	xxSRyyz	xxWPS

- Schema name becomes the Cell, Cluster or Server name, depending on its scope
- For instance, xx=cell identifier (SSCELL), yy=server/cluster identifier (yy=01 in Cluster SSSR01) and optionally z=LPAR identifier (z=D in Server SSSR01D) IF Server scope is used.

8

z/OS DB2 configuration

© 2008 IBM Corporation

A third option is to use a single database for all the components with a common storage group but to use the database scope in the construction of the schema names. This will, for instance, allow you to create more than one server or cluster in a cell that hosts business process choreography applications. As explained on the slide, the schema names use a server or cluster identifier and a possible LPAR identifier to distinguish the tables. Keep in mind that the naming possibilities are endless and you will need to take your installation's conventions into account. These are just shown as a few possibilities to get you thinking about your own conventions.

DB2 database creation

- Talk to your DB2 administrator
- Need information about database and storage group names to continue
- Configuration scripts will NOT create database or storage groups for you
- Sample SQL for creation (if using one database/stogroup):

```
DROP DATABASE xxCELL;
DROP STOGROUP xxDBSTO;
CREATE STOGROUP xxDBSTO VOLUMES ('*') VCAT xxWPS;
CREATE DATABASE xxCELL
  STOGROUP xxDBSTO
  BUFFERPOOL BP0
  CCSID UNICODE
  INDEXBP BP0;
COMMIT;

GRANT USE OF STOGROUP xxDBSTO TO xxCELL WITH GRANT OPTION;
GRANT DBADM ON DATABASE xxCELL TO xxCELL;
GRANT USE OF BUFFERPOOL BP0 TO xxCELL;
GRANT USE OF BUFFERPOOL BP1 TO xxCELL;
GRANT USE OF BUFFERPOOL BP8K0 TO xxCELL;
```

Early in the WebSphere Process Server or WebSphere Enterprise Service Bus configuration process, you should talk to your DB2 administrator. You will need to decide on the conventions that are used before getting too far into the configuration process. The configuration scripts, zWPSConfig or zWESBConfig, will configure the needed tables for you if you allow them to but they will not create the database or storage groups for you. This needs to be done ahead of time. Shown here is some sample SQL to create the database and storage group for the case where you plan to use one database name and storage group as shown on slide 7.

DB2 database configuration

- Configuration scripts will create tailored DDL and SQL for the database tables that are needed
- **dbDelayConfig** response file parameter
 - ▶ Determines whether DDL/SQL is run
- Tailored SQL/DDL provided in the DB2 directories shown here:

```
<app_server_root>/dbscripts/*
```

- ▶ CEI_xxCELL
- ▶ CommonDB
- ▶ ProcessChoreographer


As part of the configuration, the zWPSConfig and zWESBConfig scripts will create tailored DDL or SQL for the database tables that are needed by the various components you configure. One of the parameters specified in the response file provided on the zWPSConfig or zWESBConfig script invocation is the **dbDelayConfig** parameter. This parameter determines whether the DDL or SQL is run during configuration. If you are using DB2, the dbDelayConfig parameter is set to 'true', by default, since the DB2 administrator will most likely be the one responsible for actually running the DDL or SQL. If the dbDelayConfig parameter is set to 'false', the SQL or DDL for the tables will be automatically run for you. Again, the databases and storage groups are NOT created while running the zWPSConfig or zWESBConfig scripts. If the dbDelayConfig parameter is set to 'false' in the response file, you must create the databases and storage groups used by WebSphere Process Server or WebSphere Enterprise Service Bus before running the scripts. Regardless of how the dbDelayConfig parameter is set, tailored SQL or DDL is created for you in the dbscripts directory under the subdirectories shown here, with each component creating its own directory.

DB2 database configuration...

- CommonDB (WPRCSDB)

- ▶ Multiple SQL files created and must be run

- CommonDB/DB2zOSV8/xxCELL
 - createTable_AppScheduler.sql
 - createTable_CommonDB.sql
 - createTable_EsbLoggerMediation.sql
 - createTable_Recovery.sql
 - createTable_Relationship.sql
 - createTable_RelationshipMetadataTable.sql
 - createTable_customization.sql
 - createTable_lockmanager.sql
 - createTable_mediation.sql
 - insertTable_CommonDB.sql



Order is important

If you drill down into the component directories, you will see multiple SQL or DDL files that need to be run for DB2 configuration. This slide shows the ten files created for the common database. You can combine these files into one file to run but you must be careful to run the insertTable last.

DB2 database configuration...

- CEI (EVENT and EVENTCAT)

- CEI_SSCELL/ddl
 - catalogSeed.ddl
 - cr_db.ddl
 - cr_db_catalog.ddl
 - cr_tbl.ddl
 - cr_tbl_catalog.ddl
 - ins_metadata.ddl

- ProcessChoreographer

- ProcessChoreographer/DB2zOSV8/xxCELL/xxCELL
 - createSchema.sql
 - createSchema_Observer.sql

Common Event Infrastructure and ProcessChoreographer components also have a set of DDL/SQL files that need to be run.

DB2 database configuration...

- Need to create DB2 tables for the message stores used by the service integration buses
 - ▶ SIBSCA, SIBAPP, SIBBPC, SIBCEI
- sibDDLGenerator.sh script generates DDL for service integration bus message stores in DB2
 - ▶

```
./sibDDLGenerator.sh -system db2 -version 8.1  
-platform zos -schema xxyyS -user DB2D -create  
-database xxCELL -storagegroup xxDBSTO -statementend ";"  
> /u/wsuser/wpswork/SIBSCA.ddl
```
- Sample generated DDL statements require over 500 cylinders for each database

The DDL needed to create the message stores used by the service integration buses is NOT automatically created for you. The sibDDLGenerator script can be used to generate tailored DDL. An example using the script to create the SIBSCA DDL is shown. Again, note that this follows the database naming conventions shown on slide 7. Also note that the DDL created by this script requires over 500 cylinders for each database. You can tailor the generated DDL to specify smaller allocations.

DB2 database configuration...

- Can use DBUtility.sh script to run tailored SQL/DDL:

```

cd <app_server_root>/bin
  for example, cd /WebSphere/V6R0M0/AppServer/bin
export LIBPATH=/usr/lpp/db2810/jcc/lib:$LIBPATH

./DBUtility.sh createTable
-DsqlScriptPath.default=/u/wsuser/wpswork/SIBSCA.ddl
-DdbType=DB2UDBOS390_V8_1 -DdbName=xxCELL
-DprofilePath=<profile_path> -DdbJDBCProperties=/etc/db2cfg
-DdbConnectionLocation=MVS215D1 -DprofileName=default
-DdbJDBCClasspath=/usr/lpp/db2810/jcc/classes
-DdbUserId=db2d -DdbPassword=frrlday
-DdbDelayConfig=false -DdbCreateNew=false      >/tmp/DBUtility.output
2>/tmp/DBUtility.err

```

- DDL/SQL copied and run as SQL from:

```
<profile_path>/dbscripts/CommonDB/DB2zOSV8/xxCELL/SIBSCA.sql
```

- ▶ Need to delete this file if original is changed



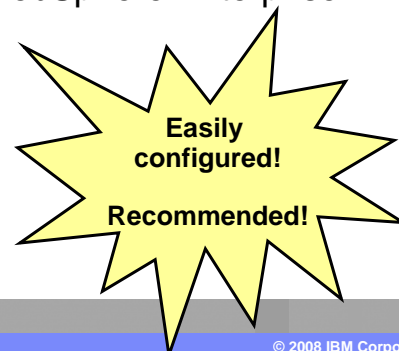
A script called DBUtility is shipped as part of WebSphere Process Server and WebSphere Enterprise Service Bus. You can use this script to run the SQL or DDL that has been tailored for your configuration. An example is shown creating the tables that were generated using the sibDDLGenerator script on the previous slide. You want to be sure to set the dbDelayConfig parameter to 'false' in order to actually run the SQL or DDL statements you specify on the sqlScriptPath. Notice that in the example, a DDL file was specified which is an EBCDIC file by definition. That file is actually copied to the path shown under the dbscripts/CommonDB directory as an SQL file encoded in ASCII instead. The copied file is the one that is actually run. Note that if the original file changes, you must delete the copy found under the dbscripts/CommonDB directory to have the changes take effect.

DB2 database option - createDB.sh

- Alternatively, use createDB.sh script
 - ▶ Sample shipped with the product to generate tailored SQL

```
<pathPrefix>/usr/lpp/zWPS/V6R1/zos.config/samples
```

- ▶ Allows DB2 administrator to work independently of WebSphere Process Server or WebSphere Enterprise Service Bus configuration



15

z/OS DB2 configuration

© 2008 IBM Corporation

Alternatively, a sample script is shipped with the product in the `zos.config/samples` directory that allows you to customize the required SQL or customize and run the required SQL at any time. This allows the DB2 administrator to work independently of the product configuration since the `createDB` script can be run before any of the configuration scripts are run. Depending on the parameters you specify, it will either just generate the needed SQL or it will generate the SQL and run it all in one step. During creation of the tailored SQL, it also combines the component SQL or DDL files into one file, automatically making sure the order is correct which makes the SQL easier to run independently. In order to run the `createDB` script, you either update the script with some important parameters, such as the SQLID you want to use, or provide those parameters on the command line as overrides. We'll look at this on the next few slides.

DB2 database option - createDB.sh

■ createDB.sh ?

createDB.sh +All -DBCreate

```
-DBCreate      Bypass Creating Database
+DBCreate      Create Database
-DBSqlid       Bypass Setting SQLID
+DBSqlid       SET CURRENT SQLID
-DBWPS         Bypass WPS
+DBWPS         Define WPS
-DBBPC         Bypass BPC
+DBBPC         Define BPC
-DBCEI         Bypass CEI
+DBCEI         Define CEI
-DBSIB         Bypass SIB
+DBSIB         Define SIB
*DBSIB         Drop/Define SIB
-DBPrefix      <Prefix> DataBase Prefix 2
                characters
-DBScope       <Scope> DataBase Scope 2
                characters, for example S1/N1/C1
-gen           Run SQL
+gen           Regenerate SQL
```

```
-All           Set All Bypass
+All           Set All Define
-CEIBP4K       Buffer pool value
-CEIBP8K       Buffer pool value
-CEIBP16K      Buffer pool value
-CEISIZE       DB Size
-DBPrefix      DB Prefix
-DBScope       DB Scope
-DBJDBCClasspath JDBC Classpath
-DBUSER        DB Userid
-DBPASSWORD    password
-DBJDBCProperties Properties
-DBConnectionLocation Conn location
-DBDelayConfig Delay config
-DBHostName     IP address
-DBServerPort  port number
-DBVOLUMES     DASD Volumes
-DBVCAT        Catalog
```

To see what parameters are available for the createDB script, specify a question mark as a parameter as shown. The plus and minus signs are used to tell the script to either add a task or take one away. For instance, you can specify **'+All -DBCreate'** in order to do everything except actually create the database. In that case, SQL to create the tables for all components and the service integration buses are customized and run.

DB2 database option - createDB.sh

■ createDB script updates

```
LIBPATH=/usr/lpp/db2810/jcc/lib:$LIBPATH  
export LIBPATH
```

```
DBCREATE=true  
DBUSESQLID=true
```

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

```
DBPREFIX=xx  
DBSCOPE=yy
```

```
DBJDBCClasspath=/usr/lpp/db2810/jcc/classes  
DBUSER=db2d  
DBPASSWORD=fr1day  
DBJDBCProperties=/etc/db2cfg  
DBConnectionLocation=MVS225D1  
DBDelayConfig=true  
DBHostName=mvs225.rtp.raleigh.ibm.com  
DBServerPort=8070  
DBVOLUMES="DB2WK1,DB2WK2"  
DBVCAT=DB2Wk8
```

```
CEIBP4K=BP0  
CEIBP8K=BP8K0  
CEIBP16K=BP16K0  
CEISIZE=10
```

If you look inside the createDB script, you'll see that you can influence its path by changing variables directly there as well. The parameters you specify on the command line will then override these variables if you wanted to change them for any reason.

DB2 database option - createDB.sh

- Assumes single database (xxCELL) for all the components with a common storage group (xxDBSTO) and schema name (xxCELL)
- Schema names for service integration buses are derived from the DBPREFIX and DBSCOPE
 - ▶ SIBSCA: xxyyS
 - ▶ SIBAPP: xxyyA
 - ▶ SIBCEI: xxyyC
 - ▶ SIBBPC: xxyyB
- Must be run from vi-capable window

There are assumptions that are made with the createDB script. The script assumes naming conventions as found on slide 7 where there is one database name with a common storage group and schema name. The service integration buses, which need unique schema names, are derived from the database prefix and scope as shown on the slide. You might need to update the resulting SQL, depending on your naming conventions. In order to tailor the SQL, the createDB script uses the vi editor with an input file to make substitutions. The place where you run the script therefore must be vi-capable. Telnet sessions work well but OMVS sessions do not.

DB2 database option - createDB.sh

- Need to run setupCmdLine.sh before invoking the script in order to set WAS_HOME:

```
cd /WebSphere/V6R1/DeploymentManager/bin
```

```
. setupCmdLine.sh
```

← Period important to have WAS_HOME set

```
cd /u/wsuser/wpswork/createDB
```

```
./createDB.sh +All
```

- SQL generated here:
 - ▶ <profile_path>/dbscripts/CommonDB/DB2zOSV8/xxCELL
- tmp directory will contain output from invocation:
 - error.out
 - output.out



The WAS_HOME variable is required in order to run the createDB script. In order to have it set, run the setupCmdLine script before running the createDB script. The slide shows the steps necessary to invoke the createDB script. The generated SQL will be found in the dbscripts directory under the profile path as shown. The output from running the createDB is written to the tmp directory in the error.out and output.out files. Look there for errors.

DB2 database option - createDB.sh

- `createDB.sh +All -DBPrefix SS`
 - ▶ Creates and runs all SQL using database name SSCELL, storage group SSDBSTO and schema SSCELI
- `createDB.sh -All *DBSIB`
 - ▶ Drops and defines the service integration tables only
- `createDB.sh +All -DBCEI`
 - ▶ Create and run all SQL EXCEPT the tables needed for CEI
- `createDB.sh +All -DBDelayConfig true`
 - ▶ Create but do not run SQL created

20

z/OS DB2 configuration

© 2008 IBM Corporation

This slide shows some examples of using the createDB script. The first example will create and run all the necessary SQL needed for WebSphere Process Server using a prefix on all the names of 'SS'. In the process of configuring WebSphere Process Server or WebSphere Enterprise Service Bus, it is sometimes necessary to clear the service integration bus data stores. The second example comes in handy for that. It will drop all the service integration bus tables and re-create them for you. The third example could be used if you had decided you were not going to configure CEI. Finally, the fourth example shows all the SQL being created but not run at the time of the createDB processing. The SQL could then be given to the DB2 administrator to further tailor and run separately.

Summary

- Many database tables are needed for WebSphere Process Server or WebSphere Enterprise Service Bus configuration
 - ▶ Talk to your DB2 administrator early
 - ▶ createDB script sample can be used for database configuration

In summary, there are many tables that you need to have defined in DB2. Talk to your DB2 administrator early and decide on naming conventions. Using the createDB script sample, the database configuration can be done independently of the WebSphere Process Server or WebSphere Enterprise Service Bus configuration. The names need to be coordinated between the two tasks, however.

Feedback

Your feedback is valuable

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

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

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WBIV61_zOSWPSDB2Configuration.ppt

This module is also available in PDF format at: [../WBIV61_zOSWPSDB2Configuration.pdf](..WBIV61_zOSWPSDB2Configuration.pdf)



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

Trademarks, copyrights, and disclaimers

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

DB2 WebSphere z/OS

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

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice.

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

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

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

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

© Copyright International Business Machines Corporation 2008. All rights reserved.

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