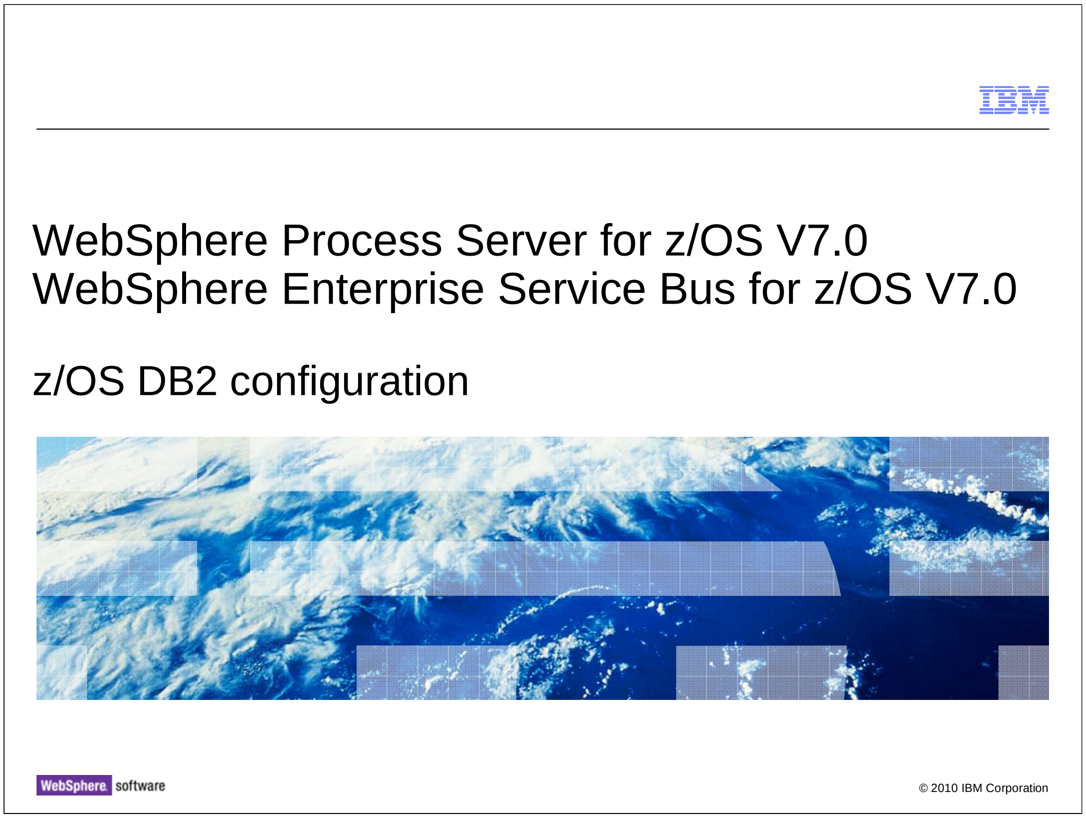


WebSphere Process Server for z/OS V7.0
WebSphere Enterprise Service Bus for z/OS V7.0

z/OS DB2 configuration

WebSphere software

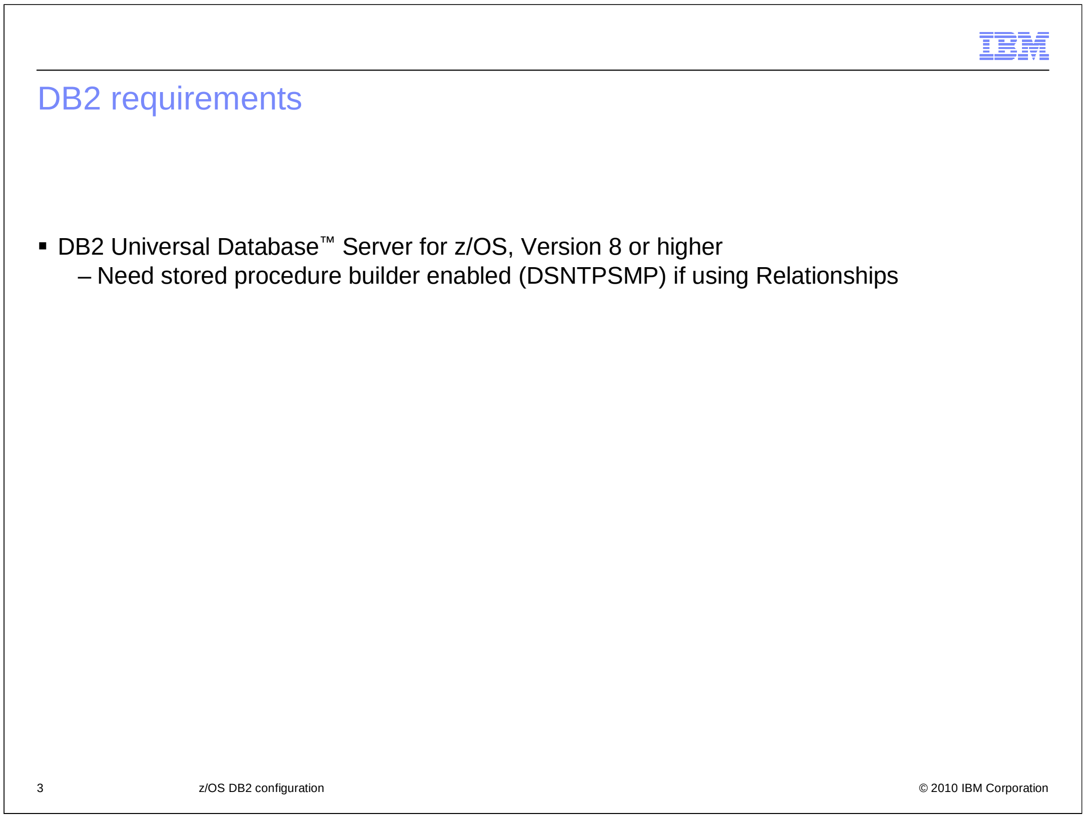
This presentation will cover the DB2® configuration for WebSphere® Process Server for z/OS V7 and WebSphere Enterprise Service Bus for z/OS V7.

Goals

- Describe DB2 requirements for WebSphere Process Server for z/OS V7.0 and WebSphere Enterprise Service Bus for z/OS V7.0
- Describe options for database configuration

2

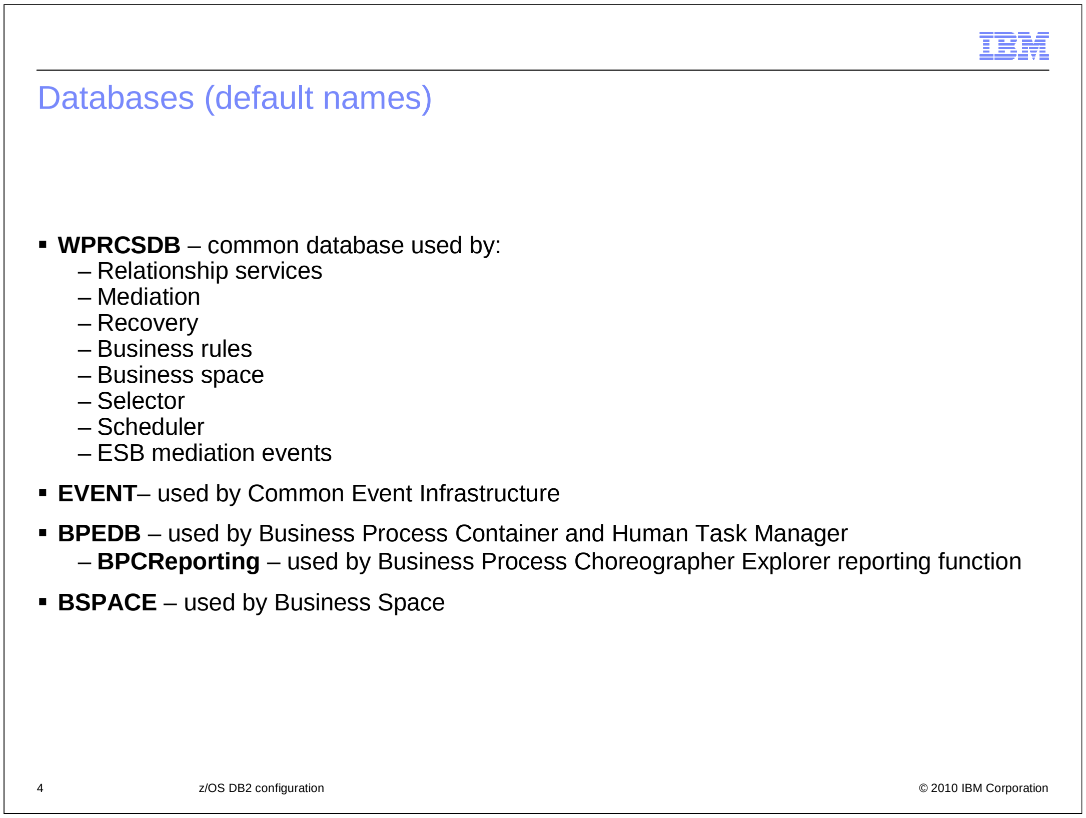
The goal of this presentation is to familiarize you with the DB2 requirements to run WebSphere Process Server or WebSphere Enterprise Service Bus. The presentation will look at the required DB2 configuration and options on how to configure it.



DB2 requirements

- DB2 Universal Database™ Server for z/OS, Version 8 or higher
  - Need stored procedure builder enabled (DSNTPSMP) if using Relationships

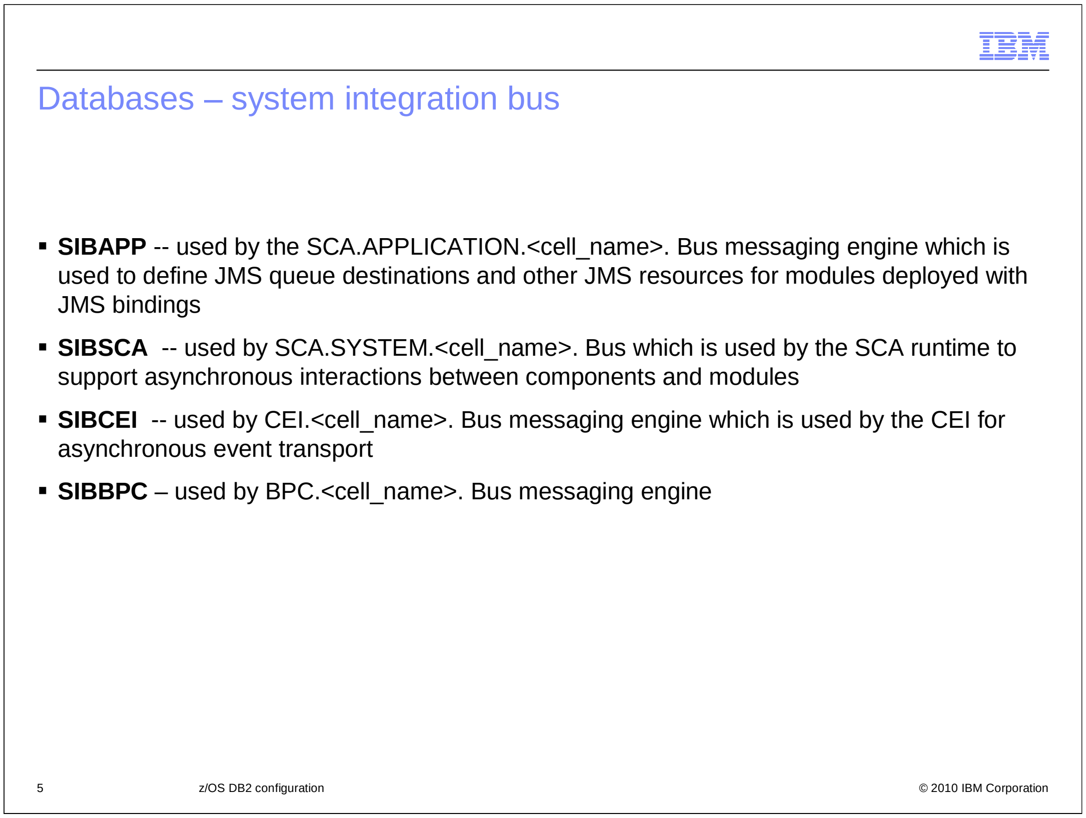
In order to implement WebSphere Process Server for z/OS V7.0 or WebSphere Enterprise Service Bus for z/OS V7.0, you must be using DB2 Universal Database Server for z/OS Version 8 or higher. If you plan to use Relationships, you will also need to have the stored procedure builder enabled (DSNTPSMP).



Databases (default names)

- **WPRCSDB** – common database used by:
  - Relationship services
  - Mediation
  - Recovery
  - Business rules
  - Business space
  - Selector
  - Scheduler
  - ESB mediation events
- **EVENT**– used by Common Event Infrastructure
- **BPEDB** – used by Business Process Container and Human Task Manager
  - **BPCReporting** – used by Business Process Choreographer Explorer reporting function
- **BSPACE** – used by Business Space

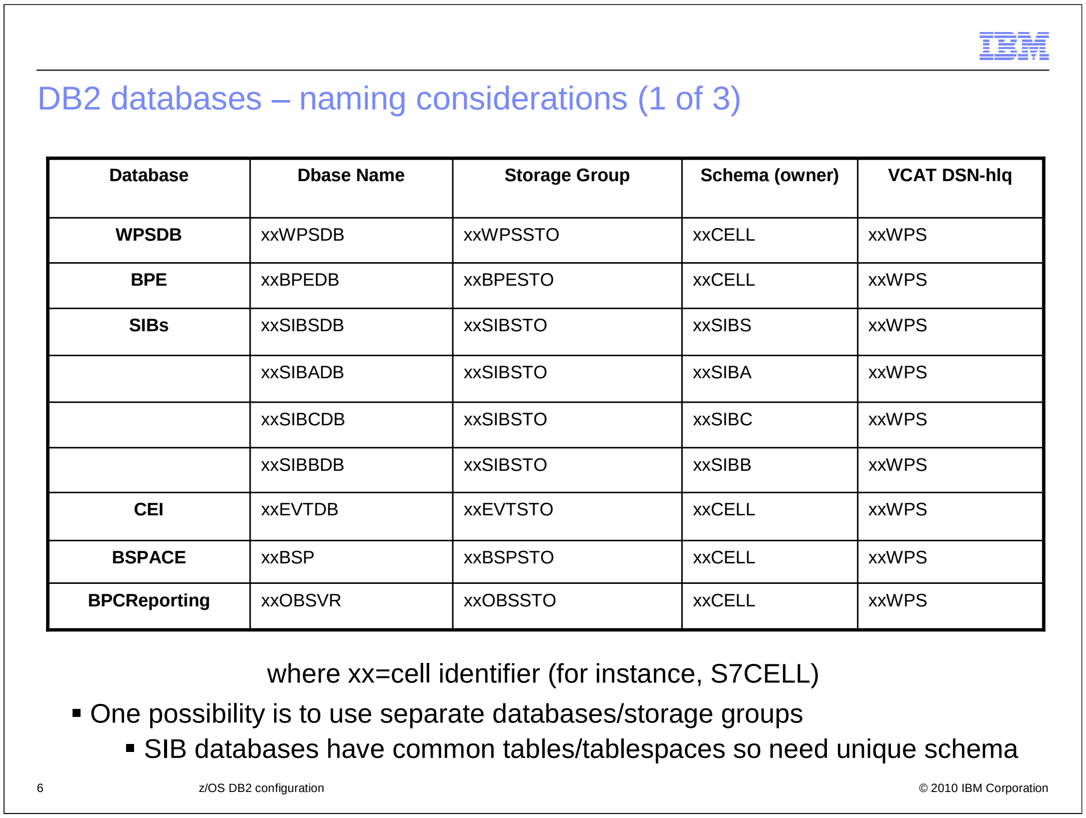
The next couple of slides describe the databases needed by the WebSphere Process Server and WebSphere Enterprise Service Bus. This slide shows the main databases needed with the default names used. While there are four databases shown here, it is important to note that you really only need to configure 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 an EVENT database but if you do not plan on using CEI, it is not necessary to configure it. 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. By default, the Business Process Choreographer Explorer reporting function also uses the BPEDB database. That can also be configured as a separate database however. 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 CEI.<cell_name>. Bus messaging engine which is used by the CEI for asynchronous event transport
- **SIBBPC** – used by BPC.<cell_name>. Bus messaging engine

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.



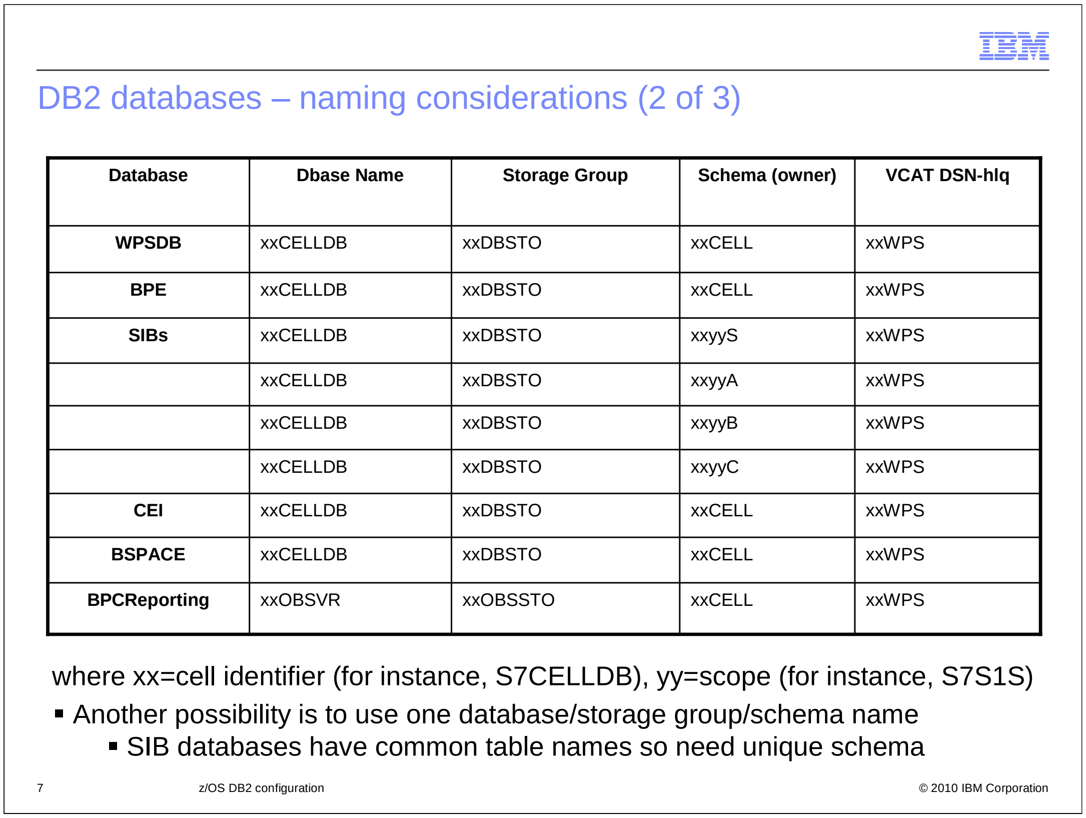
DB2 databases – naming considerations (1 of 3)

Database	Dbase Name	Storage Group	Schema (owner)	VCAT DSN-hlq
WPSDB	xxWPSDB	xxWPSSTO	xxCELL	xxWPS
BPE	xxBPEDB	xxBPESTO	xxCELL	xxWPS
SIBs	xxSIBSDB	xxSIBSTO	xxSIBS	xxWPS
	xxSIBADB	xxSIBSTO	xxSIBA	xxWPS
	xxSIBCDB	xxSIBSTO	xxSIBC	xxWPS
	xxSIBBDB	xxSIBSTO	xxSIBB	xxWPS
CEI	xxEVTDB	xxEVTSTO	xxCELL	xxWPS
BSPACE	xxBSP	xxBSPSTO	xxCELL	xxWPS
BPCReporting	xxOBSVR	xxOBSSTO	xxCELL	xxWPS

where xx=cell identifier (for instance, S7CELL)

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

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 and storage groups 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.



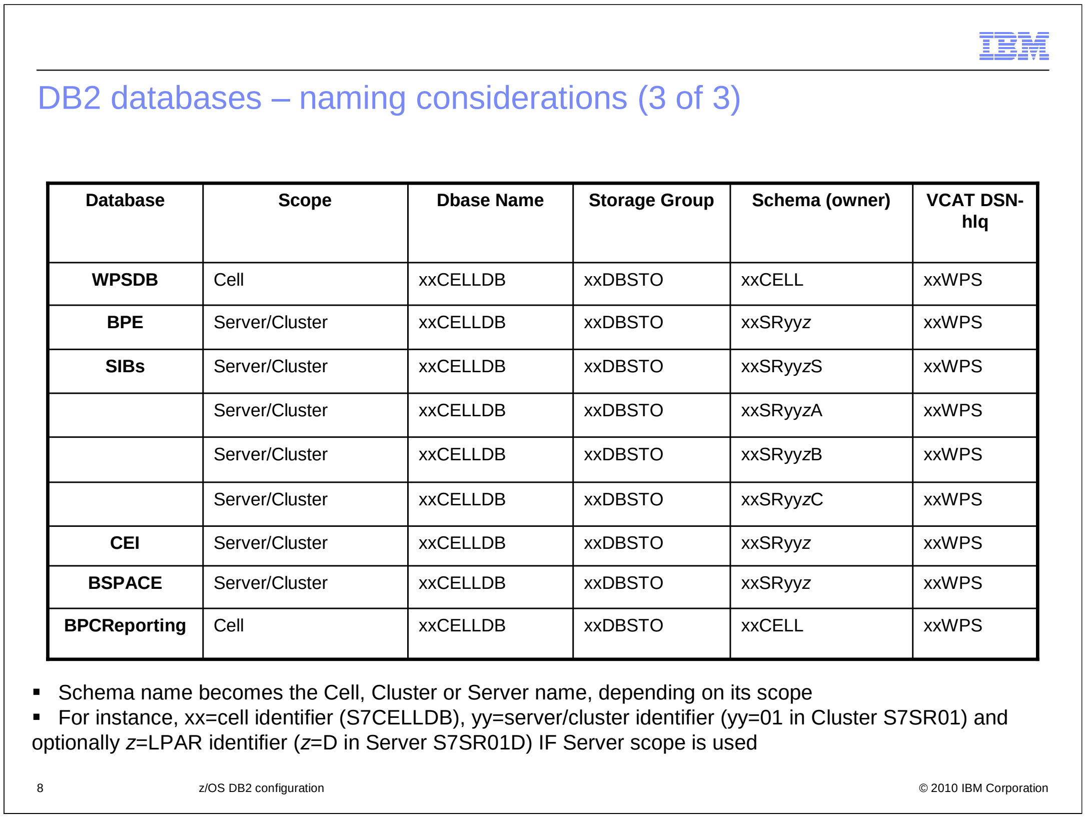
DB2 databases – naming considerations (2 of 3)

Database	Dbase Name	Storage Group	Schema (owner)	VCAT DSN-hlq
WPSDB	xxCELLDB	xxDBSTO	xxCELL	xxWPS
BPE	xxCELLDB	xxDBSTO	xxCELL	xxWPS
SIBs	xxCELLDB	xxDBSTO	xxyyS	xxWPS
	xxCELLDB	xxDBSTO	xxyyA	xxWPS
	xxCELLDB	xxDBSTO	xxyyB	xxWPS
	xxCELLDB	xxDBSTO	xxyyC	xxWPS
CEI	xxCELLDB	xxDBSTO	xxCELL	xxWPS
BSPACE	xxCELLDB	xxDBSTO	xxCELL	xxWPS
BPCReporting	xxOBSVR	xxOBSSTO	xxCELL	xxWPS

where xx=cell identifier (for instance, S7CELLDB), yy=scope (for instance, S7S1S)

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

Another option is to use a single database for all the components with a common storage group and schema name. 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.



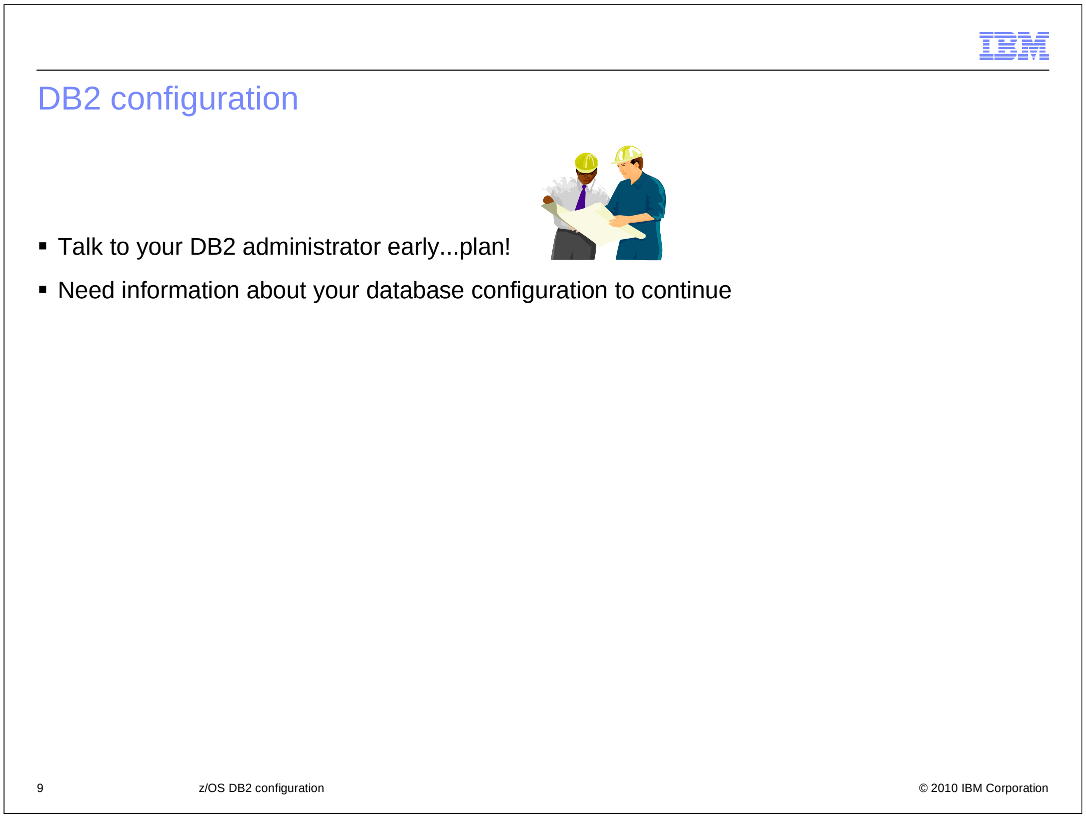
DB2 databases – naming considerations (3 of 3)

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

- Schema name becomes the Cell, Cluster or Server name, depending on its scope
- For instance, xx=cell identifier (S7CELLDB), yy=server/cluster identifier (yy=01 in Cluster S7SR01) and optionally *z*=LPAR identifier (*z*=D in Server S7SR01D) IF Server scope is used

8 z/OS DB2 configuration © 2010 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.

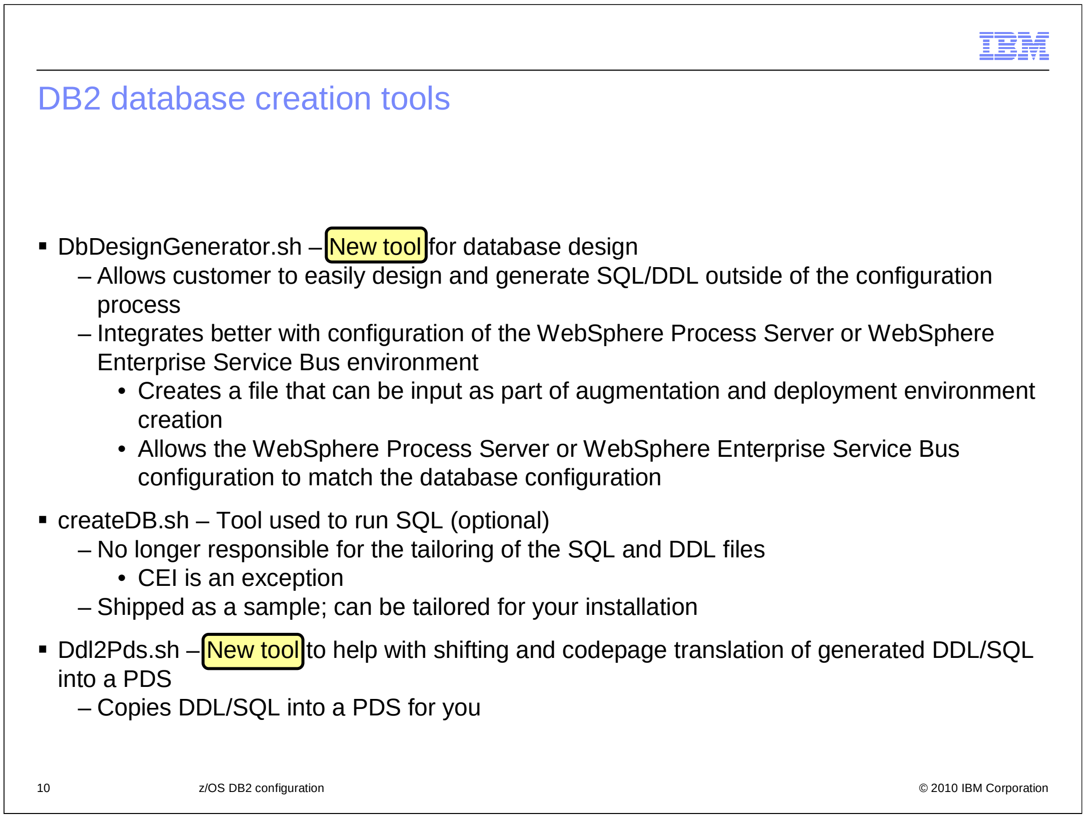


DB2 configuration

- Talk to your DB2 administrator early...plan!
- Need information about your database configuration to continue

9 z/OS DB2 configuration

An important point you should get from the naming consideration slides is that 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. Take some time to do some planning up front! The configuration scripts, zWPSConfig or zWESBConfig, will need information about the DB2 objects that you plan to use. The next few slides will show you some tools to help in the DB2 configuration process.



DB2 database creation tools

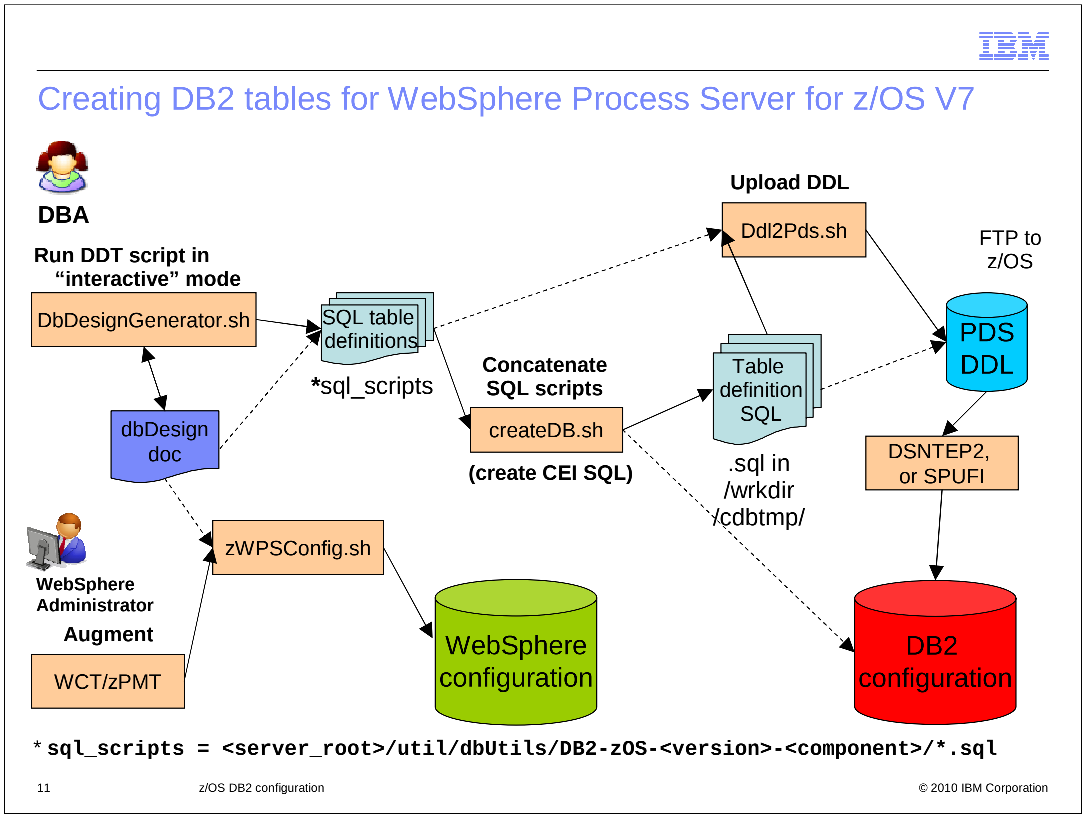
- DbDesignGenerator.sh – New tool for database design
  - Allows customer to easily design and generate SQL/DDL outside of the configuration process
  - Integrates better with configuration of the WebSphere Process Server or WebSphere Enterprise Service Bus environment
    - Creates a file that can be input as part of augmentation and deployment environment creation
    - Allows the WebSphere Process Server or WebSphere Enterprise Service Bus configuration to match the database configuration
- createDB.sh – Tool used to run SQL (optional)
  - No longer responsible for the tailoring of the SQL and DDL files
    - CEI is an exception
  - Shipped as a sample; can be tailored for your installation
- Ddl2Pds.sh – New tool to help with shifting and codepage translation of generated DDL/SQL into a PDS
  - Copies DDL/SQL into a PDS for you

10 z/OS DB2 configuration © 2010 IBM Corporation

As you have seen, in order to use WebSphere Process Server for z/OS and WebSphere Enterprise Service Bus for z/OS, some databases, tables and other objects are needed. The tools shown on the slide are useful in the configuration of these objects on DB2. The first tool, dbDesignGenerator, is new in version 7. It is provided to help in the design of the databases needed for various components of the WebSphere Process Server and WebSphere Enterprise Service Bus products. This tool allows the database configuration to proceed in parallel once the zWPSInstall or zWESBInstall scripts have been run, thus allowing better separation of responsibility between the database administration and configuration roles. After using the tool to design the database configuration for your installation, the tool creates a file that can be used as input to the WebSphere Process Server or WebSphere Enterprise Service Bus augmentation process. It is used as input to the augmentation process and the deployment environment creation process. This ensures that your database configuration matches your WebSphere Process Server or WebSphere Enterprise Service Bus configuration. The tool will also generate the corresponding SQL that can then be turned over to your DB2 administrator to run.

The second tool, createDB.sh, is still available to run the SQL but it is no longer responsible for tailoring the SQL and DDL files as it has been in past releases. DbDesignGenerator has taken over that task. SQL for the Common Event Infrastructure (CEI) is an exception to this. You will see on a later slide how it is needed to tailor the SQL for the CEI component. Note that createDB is shipped as a sample so it can also be tailored to meet your installation’s requirements if need be.

Another new tool in version 7 is the Ddl2Pds tool. It allows you to take the SQL that is created for you by dbDesignGenerator in the HFS and move it over to a PDS so that it can be run from there instead. It takes care of codepage translation for you.



Creating DB2 tables for WebSphere Process Server for z/OS V7

DBA

Run DDT script in
“interactive” mode

DbDesignGenerator.sh

SQL table definitions

*sql_scripts

dbDesign doc

Upload DDL

Ddl2Pds.sh

FTP to z/OS

PDS DDL

Concatenate SQL scripts

createDB.sh

(create CEI SQL)

Table definition SQL

.sql in /wrkdir /cdbtmp/

DSNTEP2, or SPUFI

WebSphere Administrator

Augment

WCT/zPMT

zWPSConfig.sh

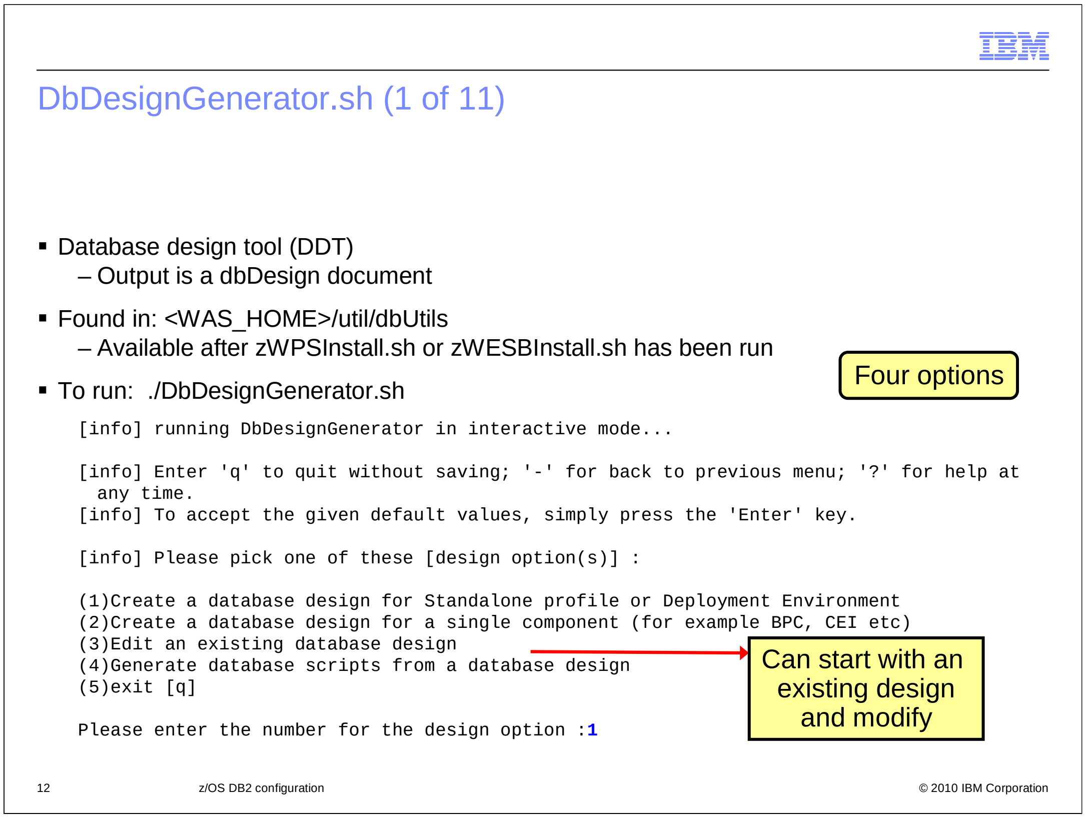
WebSphere configuration

DB2 configuration

`* sql_scripts = <server_root>/util/dbUtils/DB2-zOS-<version>-<component>/*.sql`

11 z/OS DB2 configuration © 2010 IBM Corporation

This slide is meant to show how the various tools work together to give you a configured system. Starting in the top left corner, you see that you need to work with your DB2 administrator to run the DbDesignGenerator ‘tool’, or shell script. You will need information from your DB2 administrator as input to the tool. The output of the DbDesignGenerator tool is a dbDesign file that is the used as input to the WCT for your augment definition. As shown on the slide, that is used by your WebSphere administrator when running the zWPSConfig shell script during augmentation. The dbDesign file is also used as input to generate SQL scripts for database definition as shown to the right of the DBDesignGenerator box. The dbDesign file allows the database definition to easily match the WebSphere configuration that is created when running the zWPSConfig shell script used for augmentation. Once you have the SQL table definitions created using the DbDesignGenerator, those are used as input to the createDB shell script. The createDB shell script does a few things for you. If you are able, it will actually create the needed DB2 configuration by running the SQL that has been created as seen by the dashed line leading to the DB2 configuration. If running the SQL at this point is not an option, you should use it to concatenate the SQL into component files that are more easily processed by your DB2 administrator. You should also use it to create the SQL for CEI at this point. Remember, the SQL for CEI hast has not been created for you by dbDesignGenerator. Once all the SQL has been created, you can pass it to the Ddl2Pds shell script as shown in the top right corner. This will copy it to a PDS that can be used as input to DSNTEP2 or SPUFI to be run by your DB2 administrator. You will see more details on each of these tools on the next slides.



DbDesignGenerator.sh (1 of 11)

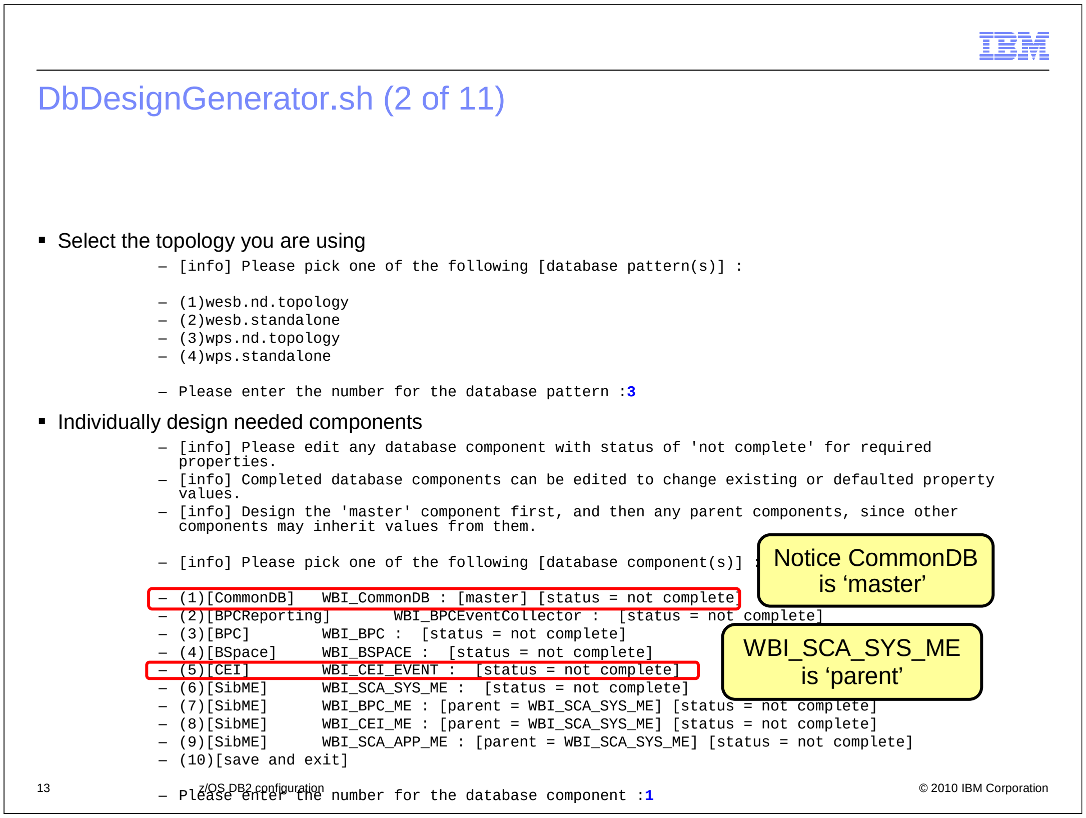
- Database design tool (DDT)
  - Output is a dbDesign document
- Found in: <WAS_HOME>/util/dbUtils
  - Available after zWPSInstall.sh or zWESBInstall.sh has been run
- To run: ./DbDesignGenerator.sh

Four options

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at
  any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of these [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component (for example BPC, CEI etc)
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]

Please enter the number for the design option :1
```

Can start with an existing design and modify

The DbDesignGenerator shell script is known as the database design tool, or DDT. It is found in the util/dbUtils directory under the WAS_HOME directory after you have run zWPSInstall or zWESBInstall. The output from running the DDT is a dbDesign document. The tool can be run interactively to prompt you for various information it needs to create the dbDesign document. As shown on the previous slide, the dbDesign document can then be used as input to the zWPSConfig shell script to augment a Websphere configuration to include WebSphere Process Server or WebSphere Enterprise Service Bus function. The DDT tool can also use the information in the dbDesign document to generate the SQL scripts for the DB2 configuration needed for the products. You see on the slide that the DDT allows you to create a database design for the entire environment, either stand-alone or deployment. It also allows you to create a design for a single component, such as BPC. If you have an existing configuration, option number three allows you to start with that database design and make only needed changes so you do not have to start the design from scratch each time. Option four will take a populated dbDesign document and generate the SQL scripts for you. This presentation will show you the design creation for a deployment environment.



DbDesignGenerator.sh (2 of 11)

- Select the topology you are using

```
– [info] Please pick one of the following [database pattern(s)] :

– (1)wesb.nd.topology
– (2)wesb.standalone
– (3)wps.nd.topology
– (4)wps.standalone

– Please enter the number for the database pattern :3
```

- Individually design needed components

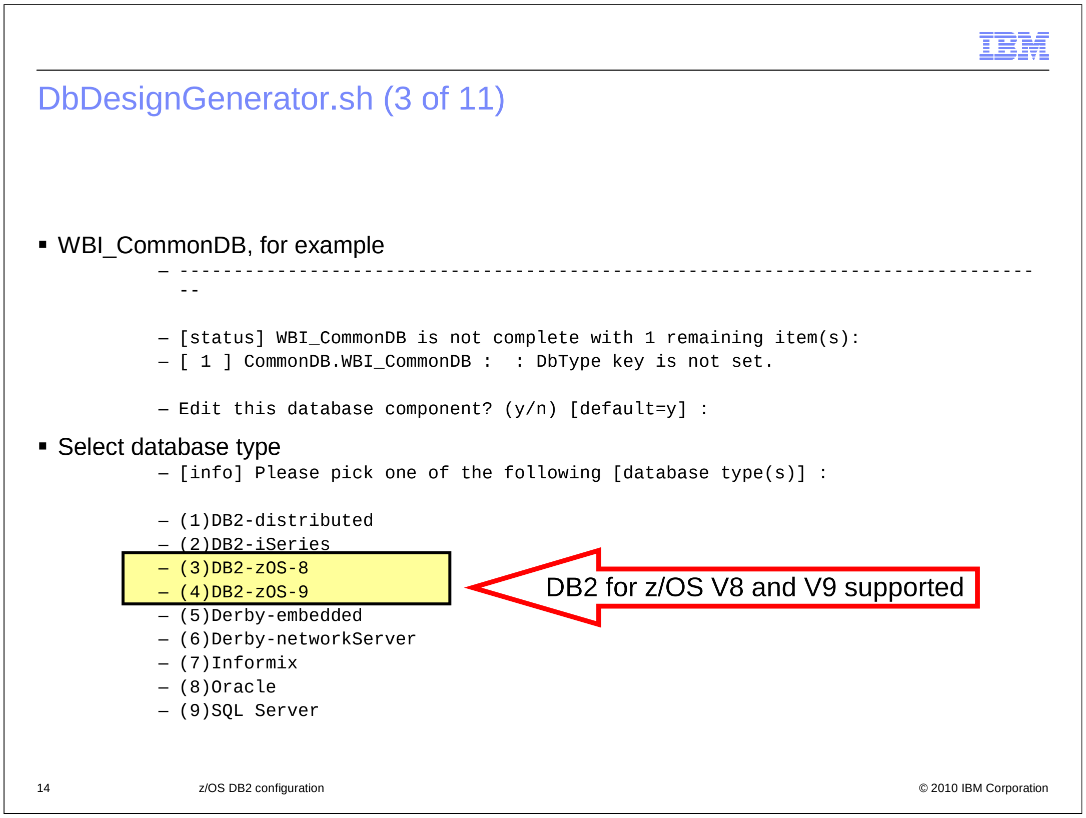
```
– [info] Please edit any database component with status of 'not complete' for required
  properties.
– [info] Completed database components can be edited to change existing or defaulted property
  values.
– [info] Design the 'master' component first, and then any parent components, since other
  components may inherit values from them.

– [info] Please pick one of the following [database component(s)] :

– (1)[CommonDB]   WBI_CommonDB : [master] [status = not complete]
– (2)[BPCReporting]       WBI_BPCEventCollector :  [status = not complete]
– (3)[BPC]        WBI_BPC :  [status = not complete]
– (4)[BSpace]     WBI_BSPACE :  [status = not complete]
– (5)[CEI]        WBI_CEI_EVENT :  [status = not complete]
– (6)[SibME]      WBI_SCA_SYS_ME :  [status = not complete]
– (7)[SibME]      WBI_BPC_ME : [parent = WBI_SCA_SYS_ME] [status = not complete]
– (8)[SibME]      WBI_CEI_ME : [parent = WBI_SCA_SYS_ME] [status = not complete]
– (9)[SibME]      WBI_SCA_APP_ME : [parent = WBI_SCA_SYS_ME] [status = not complete]
– (10)[save and exit]

– Please enter the number for the database component :1
```

Notice CommonDB is ‘master’

WBI_SCA_SYS_ME is ‘parent’

13 z/OS DB2 configuration © 2010 IBM Corporation

One of the first things the DDT asks about, as shown on the slide, is the topology that you are configuring. Once that has been selected, you can then go through each component that you plan to configure and input the needed information for your particular database system. The WBI_CommonDB configuration is required before doing any other. It is considered to be the ‘master’ component. In this example, the WBI_SCA_SYS_ME component is the ‘parent’ to each of the other SibME components and needs to be configured before any of the others. This slide shows the wps.nd.topology but the other topologies are similar with the ‘master’ component being the first component required and ‘parent’ components being pre-requisites for others. You’ll see the CommonDB configuration on the next few slides.



DbDesignGenerator.sh (3 of 11)

- WBI_CommonDB, for example

```
– -------------------------------------------------------------------------------
  --

– [status] WBI_CommonDB is not complete with 1 remaining item(s):
– [ 1 ] CommonDB.WBI_CommonDB :  : DbType key is not set.

– Edit this database component? (y/n) [default=y] :
```

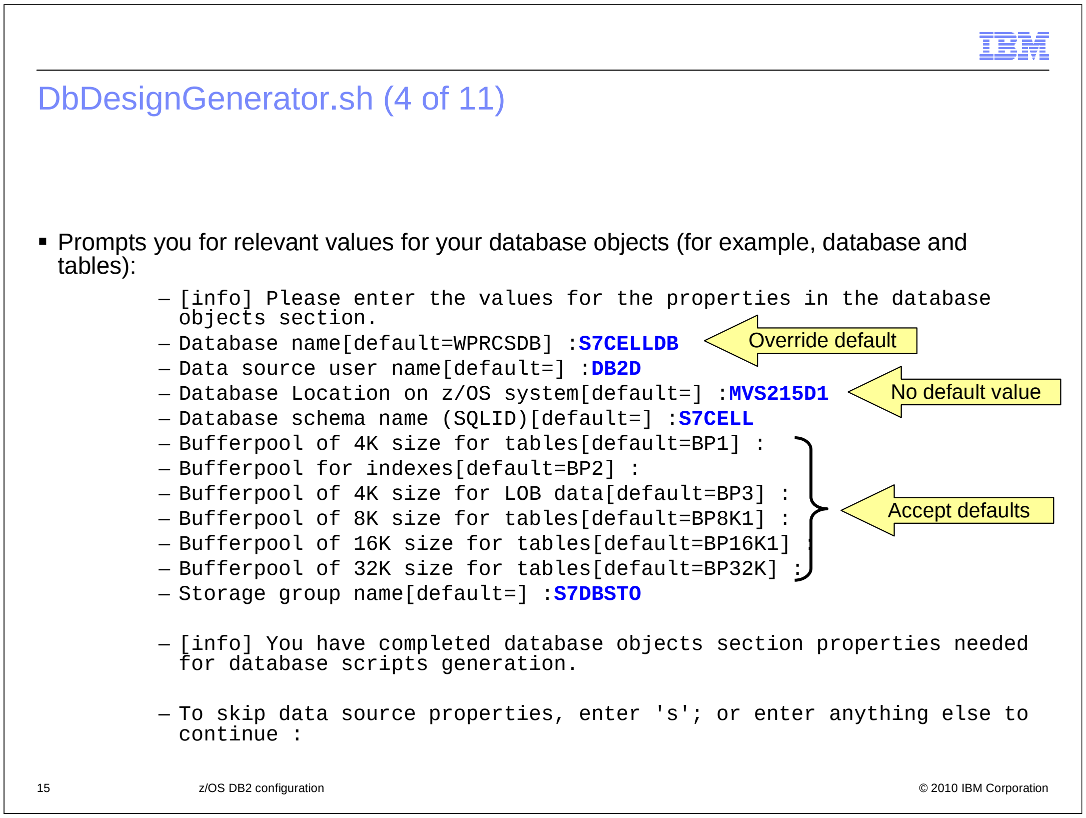
- Select database type

```
– [info] Please pick one of the following [database type(s)] :

– (1)DB2-distributed
– (2)DB2-iSeries
– (3)DB2-zOS-8
– (4)DB2-zOS-9
– (5)Derby-embedded
– (6)Derby-networkServer
– (7)Informix
– (8)Oracle
– (9)SQL Server
```

DB2 for z/OS V8 and V9 supported

14 z/OS DB2 configuration © 2010 IBM Corporation

Since you are creating a dbDesign document from scratch, WBI_CommonDB will not show as ‘complete’ when selected. You want to specify ‘y’ to edit the component. One of the next pieces of information that you are asked to provide is the database type. You’ll notice that DB2 for z/OS Versions 8 and 9 are highlighted here. These are the options you’ll want to select for your WebSphere Process Server for z/OS or WebSphere Enterprise Service Bus for z/OS configurations. While Derby is supported in the stand-alone case, you do not need to create a database design document to use that. That can more easily be configured automatically during augmentation.

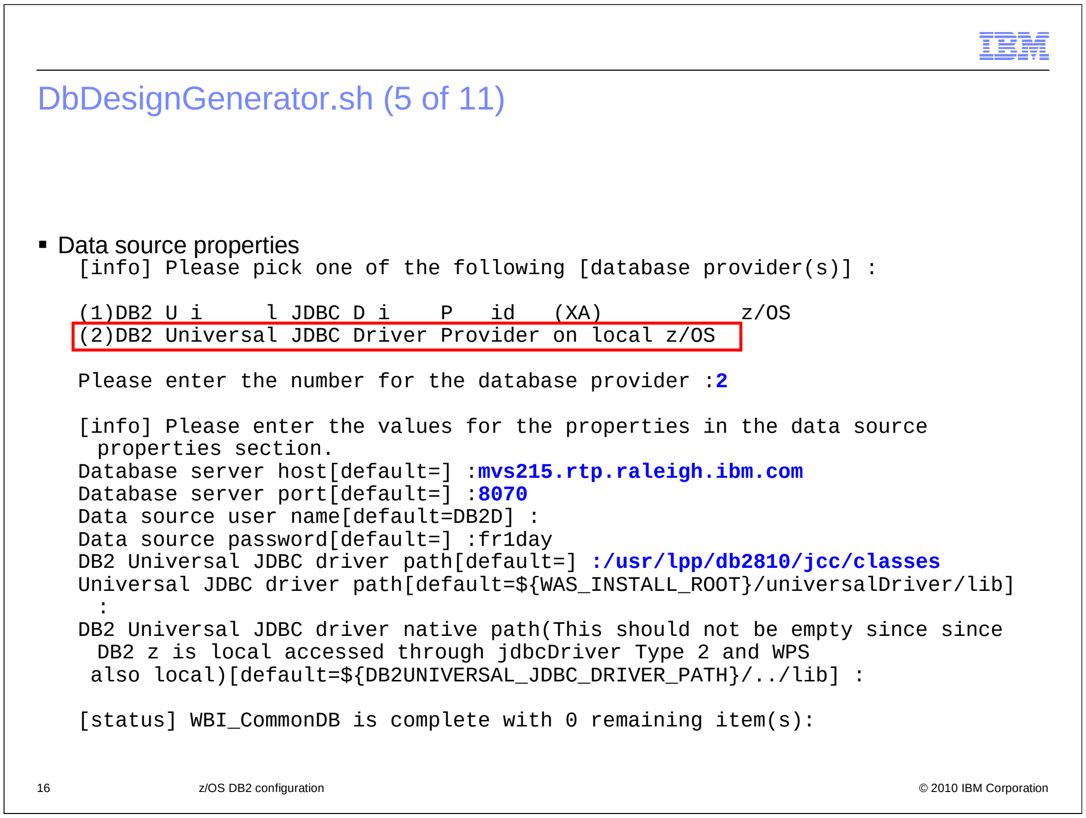


DbDesignGenerator.sh (4 of 11)

- Prompts you for relevant values for your database objects (for example, database and tables):

```
– [info] Please enter the values for the properties in the database
  objects section.
– Database name[default=WPRCSDB] :S7CELLDB
– Data source user name[default=] :DB2D
– Database Location on z/OS system[default=] :MVS215D1
– Database schema name (SQLID)[default=] :S7CELL
– Bufferpool of 4K size for tables[default=BP1] :
– Bufferpool for indexes[default=BP2] :
– Bufferpool of 4K size for LOB data[default=BP3] :
– Bufferpool of 8K size for tables[default=BP8K1] :
– Bufferpool of 16K size for tables[default=BP16K1] :
– Bufferpool of 32K size for tables[default=BP32K] :
– Storage group name[default=] :S7DBSTO

– [info] You have completed database objects section properties needed
  for database scripts generation.

– To skip data source properties, enter 's'; or enter anything else to
  continue :
```

As you move to the next screen in the DDT, it will prompt you for values that will be used to configure the database objects that are needed. For many of the values, you are given the opportunity to accept the default value shown, by hitting enter. In this example, the default database name was overridden while all of the default bufferpools were used. If there is no default value, as seen for the ‘database location’, you need to provide a value before it will allow you to continue. As seen at the bottom of the slide, you need to specify data source properties before the component is considered complete.

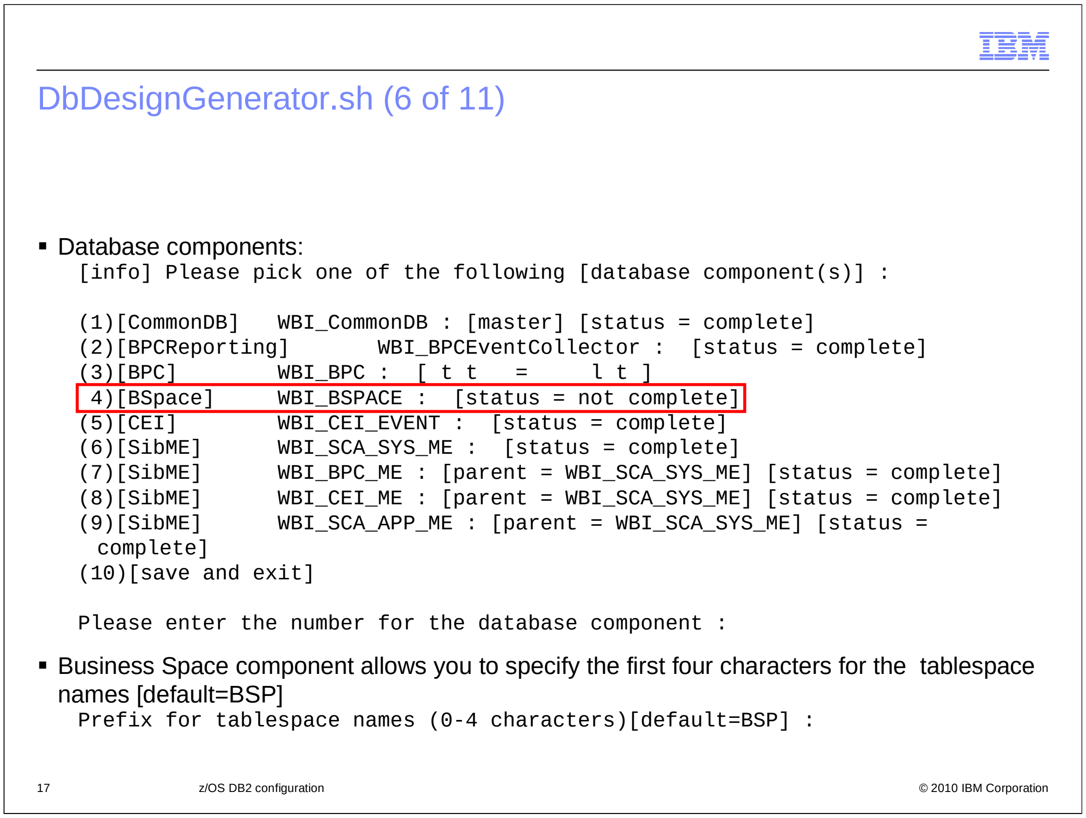


DbDesignGenerator.sh (5 of 11)

- Data source properties

```
[info] Please pick one of the following [database provider(s)] :

(1)DB2 U i      l JDBC D i    P   id   (XA)           z/OS
(2)DB2 Universal JDBC Driver Provider on local z/OS

Please enter the number for the database provider :2

[info] Please enter the values for the properties in the data source
  properties section.
Database server host[default=] :mvs215.rtp.raleigh.ibm.com
Database server port[default=] :8070
Data source user name[default=DB2D] :
Data source password[default=] :fr1day
DB2 Universal JDBC driver path[default=] :/usr/lpp/db2810/jcc/classes
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/universalDriver/lib]
  :
DB2 Universal JDBC driver native path(This should not be empty since since
  DB2 z is local accessed through jdbcDriver Type 2 and WPS
 also local)[default=${DB2UNIVERSAL_JDBC_DRIVER_PATH}/../lib] :

[status] WBI_CommonDB is complete with 0 remaining item(s):
```

For data source properties, you are prompted for the type of JDBC Driver provider you are configuring. This slide shows the local, type 2 driver being configured. The important information here is the JDBC driver path and the JDBC driver native path for your DB2 installation. Once you have provided this information, the WBI_CommonDB component is marked as complete.



DbDesignGenerator.sh (6 of 11)

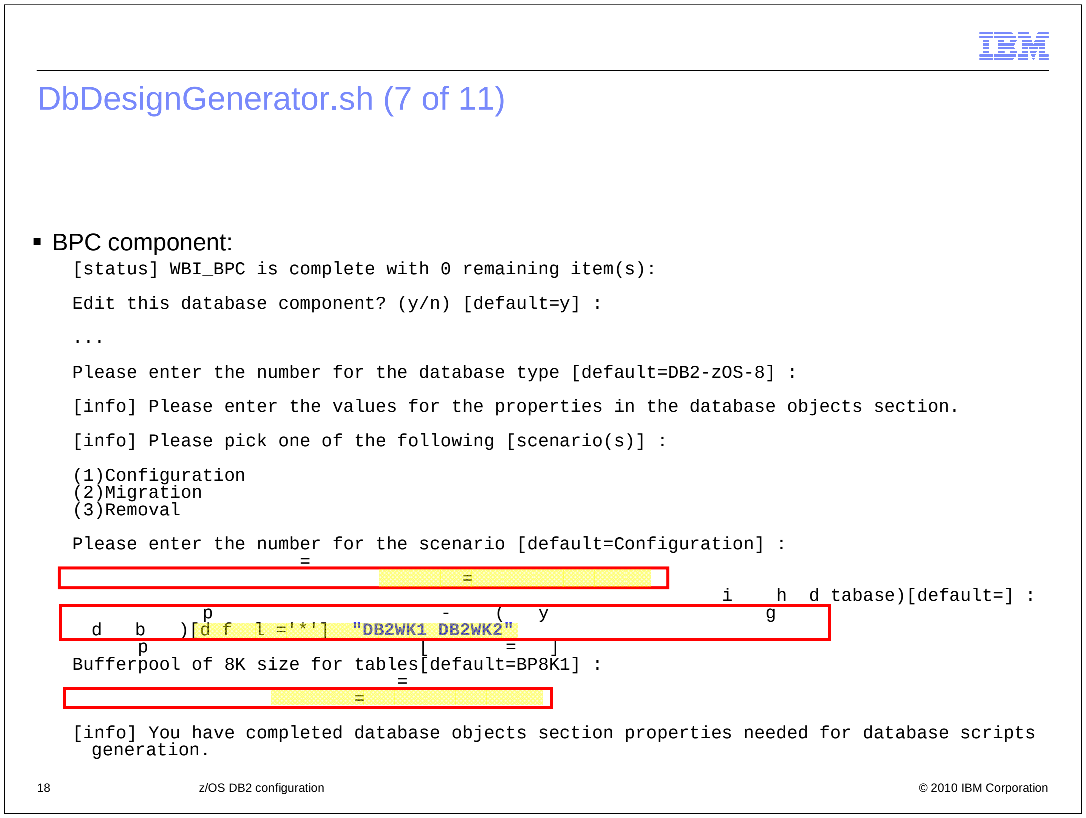
- Database components:

```
[info] Please pick one of the following [database component(s)] :

(1)[CommonDB]   WBI_CommonDB : [master] [status = complete]
(2)[BPCReporting]       WBI_BPCEventCollector :  [status = complete]
(3)[BPC]        WBI_BPC :  [ t t   =     l t ]
 4)[BSpace]     WBI_BSPACE :  [status = not complete]
(5)[CEI]        WBI_CEI_EVENT :  [status = complete]
(6)[SibME]      WBI_SCA_SYS_ME :  [status = complete]
(7)[SibME]      WBI_BPC_ME : [parent = WBI_SCA_SYS_ME] [status = complete]
(8)[SibME]      WBI_CEI_ME : [parent = WBI_SCA_SYS_ME] [status = complete]
(9)[SibME]      WBI_SCA_APP_ME : [parent = WBI_SCA_SYS_ME] [status =
  complete]
(10)[save and exit]

Please enter the number for the database component :
```

- Business Space component allows you to specify the first four characters for the tablespace names [default=BSP]

```
Prefix for tablespace names (0-4 characters)[default=BSP] :
```

Now back on the database component screen, you will see that many of the components are actually marked as ‘complete’ after finishing only the master component. All components need to be complete before having a valid dbDesign document. You see here that only the business space component is still incomplete. The business space component will prompt you for a tablespace name prefix as seen on the slide. This can be up to four characters in length but will default to BSP if not changed. Once the business space configuration is marked as complete, you can theoretically generate the dbDesign document however it’s a good practice to verify the values that are being used for the other components as well. For example, typically the default value for "Database schema name (SQLID)" is set by component. It is not inherited from the Common DB settings as you might expect, and so it is likely that you will want to customize that field.



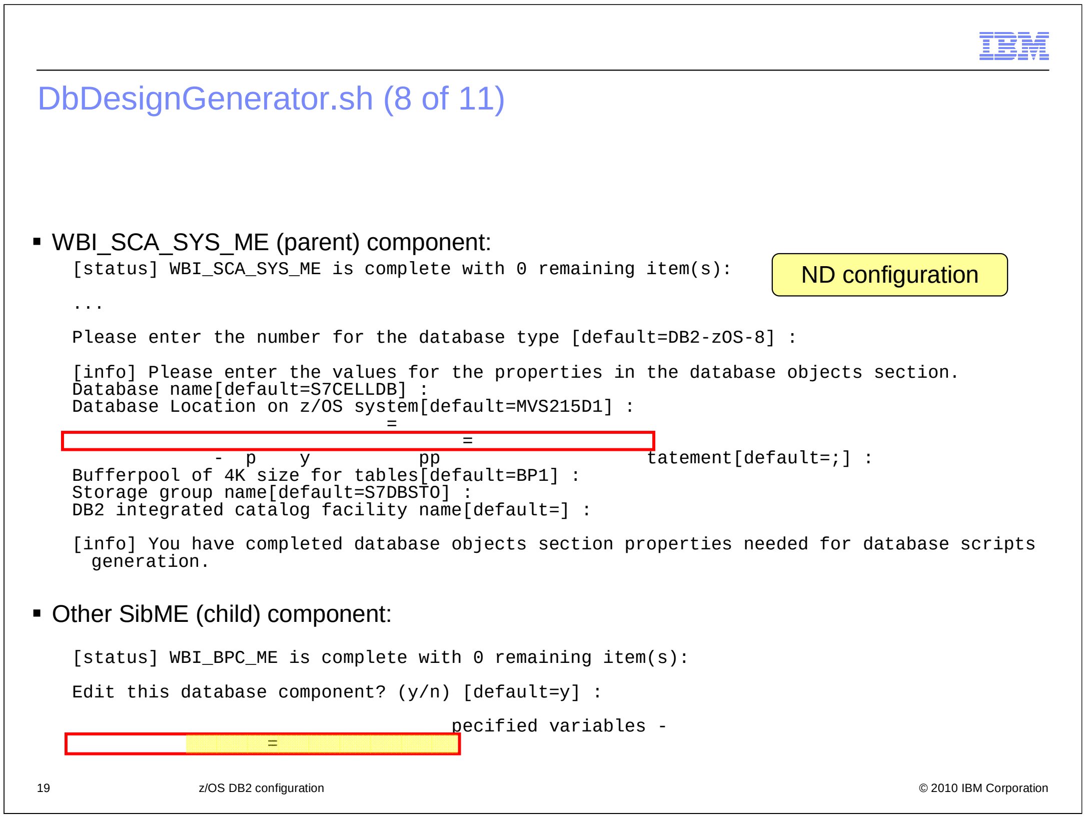
DbDesignGenerator.sh (7 of 11)

- BPC component:

```
[status] WBI_BPC is complete with 0 remaining item(s):

Edit this database component? (y/n) [default=y] :

...

Please enter the number for the database type [default=DB2-zOS-8] :

[info] Please enter the values for the properties in the database objects section.

[info] Please pick one of the following [scenario(s)] :

(1)Configuration
(2)Migration
(3)Removal

Please enter the number for the scenario [default=Configuration] :
                      =
                                        =
                                                                    i    h  d tabase)[default=] :
            p                        -    (   y                                g
  d   b    )[d f  l ='*']  "DB2WK1 DB2WK2"
      p                                [        =   ]
Bufferpool of 8K size for tables[default=BP8K1] :
                              =
                        =

[info] You have completed database objects section properties needed for database scripts
  generation.
```

18 z/OS DB2 configuration © 2010 IBM Corporation

On this slide you see the configuration for the BPC component. While it is complete, you should note that some of the default values that are used may not be what you had intended. For instance, the default database schema name is WPRBE00. You will see that was changed to S7CELL to match naming conventions. The DB2 volume-ids and storage group were also changed.



IBM

DbDesignGenerator.sh (8 of 11)

- WBI_SCA_SYS_ME (parent) component:

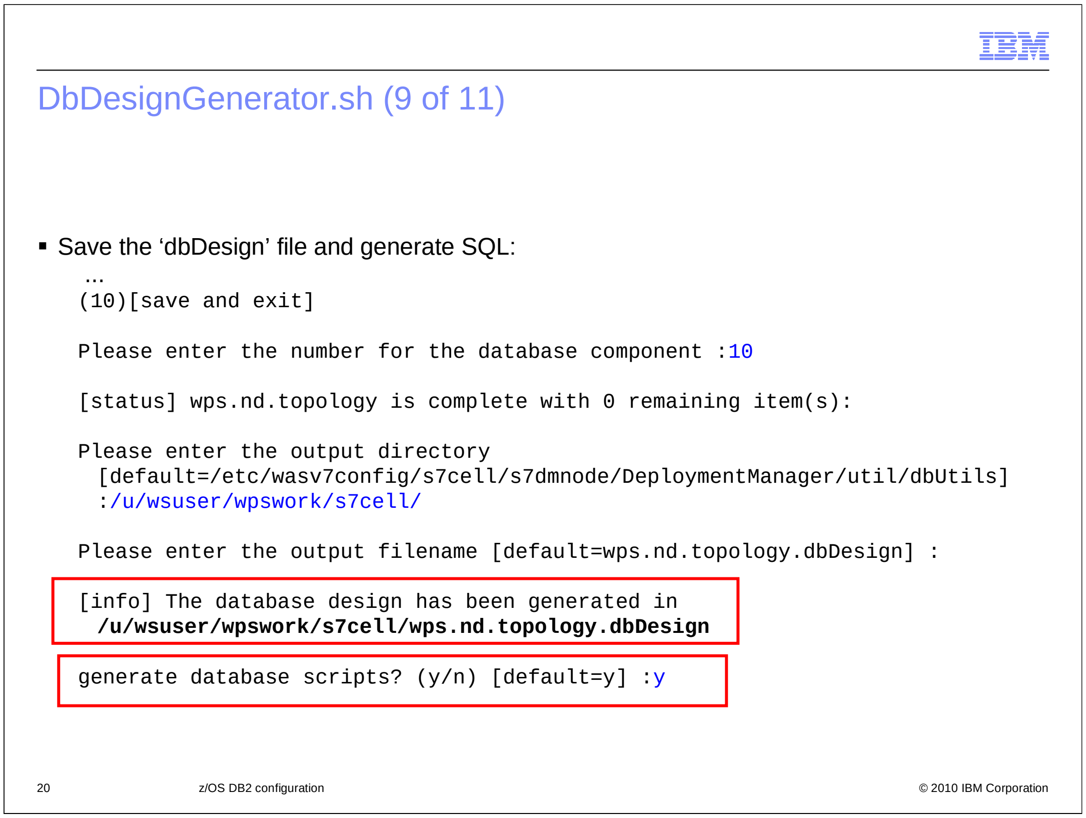
```
[status] WBI_SCA_SYS_ME is complete with 0 remaining item(s):

...

Please enter the number for the database type [default=DB2-zOS-8] :

[info] Please enter the values for the properties in the database objects section.
Database name[default=S7CELLDB] :
Database Location on z/OS system[default=MVS215D1] :
                             =
                                    =
             -  p    y          pp                     tatement[default=;] :
Bufferpool of 4K size for tables[default=BP1] :
Storage group name[default=S7DBSTO] :
DB2 integrated catalog facility name[default=] :

[info] You have completed database objects section properties needed for database scripts
  generation.
```

ND configuration

- Other SibME (child) component:

```
[status] WBI_BPC_ME is complete with 0 remaining item(s):

Edit this database component? (y/n) [default=y] :

                                   pecified variables -
                  =
```

Recall that the WBI_SCA_SYS_ME component was defined as a parent of all the other SibME components. For the stand-alone configuration, the SibME components are actually children of the CommonDB. This slide shows the definition of the parent in the network deployment case. You'll note that the only field needed for the other SibME components, or children, is the schemaName. This is shown on the bottom of the slide. You will most likely want to change the defaults for each one.

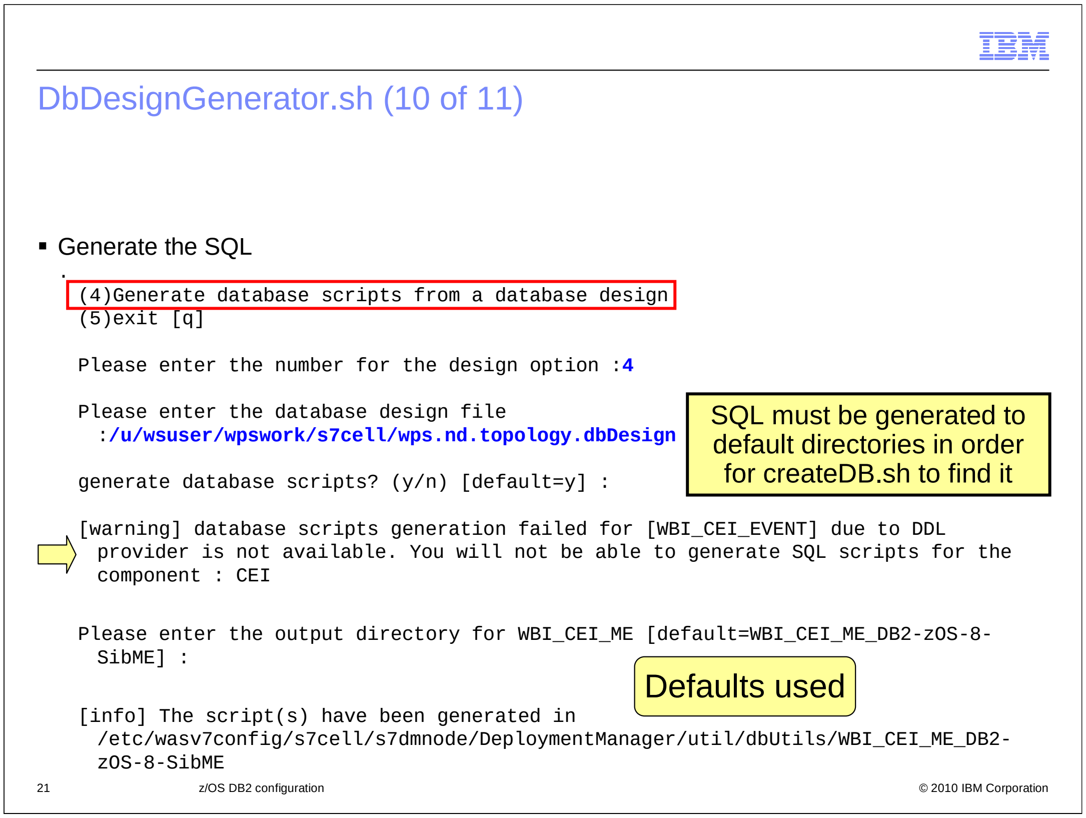


DbDesignGenerator.sh (9 of 11)

- Save the ‘dbDesign’ file and generate SQL:

```
 ...
(10)[save and exit]

Please enter the number for the database component :10

[status] wps.nd.topology is complete with 0 remaining item(s):

Please enter the output directory
  [default=/etc/wasv7config/s7cell/s7dmnode/DeploymentManager/util/dbUtils]
  :/u/wsuser/wpswork/s7cell/

Please enter the output filename [default=wps.nd.topology.dbDesign] :

[info] The database design has been generated in
  /u/wsuser/wpswork/s7cell/wps.nd.topology.dbDesign

generate database scripts? (y/n) [default=y] :y
```

After checking each component to make sure it matches the design you had settled on, you need to do a ‘save and exit’ in order to create a dbDesign file. You will use this file as input to the augment. Finally, you are then given the option of generating the SQL for your database objects. This is shown on the next slide.



DbDesignGenerator.sh (10 of 11)

- Generate the SQL

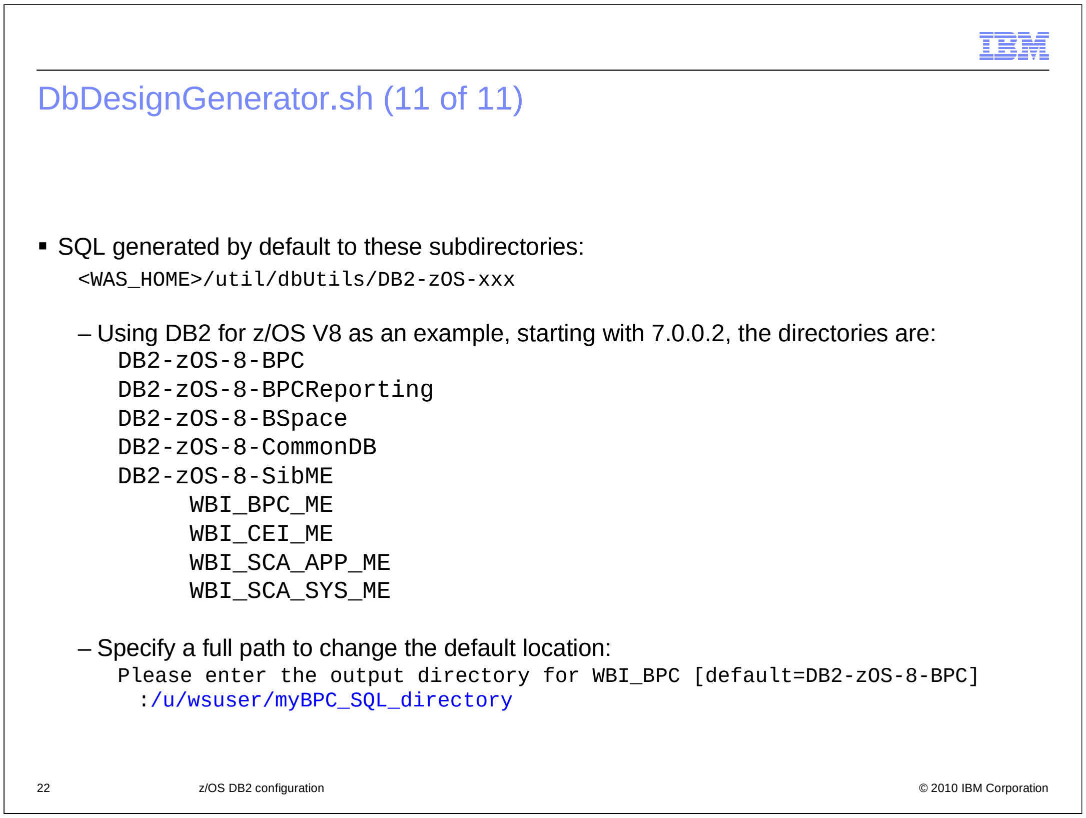
```
.
(4)Generate database scripts from a database design
(5)exit [q]

Please enter the number for the design option :4

Please enter the database design file
  :/u/wsuser/wpswork/s7cell/wps.nd.topology.dbDesign

generate database scripts? (y/n) [default=y] :

[warning] database scripts generation failed for [WBI_CEI_EVENT] due to DDL
  provider is not available. You will not be able to generate SQL scripts for the
  component : CEI


Please enter the output directory for WBI_CEI_ME [default=WBI_CEI_ME_DB2-zOS-8-
  SibME] :

[info] The script(s) have been generated in
  /etc/wasv7config/s7cell/s7dmnode/DeploymentManager/util/dbUtils/WBI_CEI_ME_DB2-
  zOS-8-SibME
```

SQL must be generated to default directories in order for createDB.sh to find it

Defaults used

21 z/OS DB2 configuration © 2010 IBM Corporation

After entering the location of the database design file that was specified on the previous slide, you are prompted for an output directory for each component to generate the required SQL. In order for the createDB shell script to find the scripts, you should keep the default directories. You'll notice that there is a warning about database script generation for the CEI component. The createDB shell script is used for this instead. Once you have provided output locations for the SQL for each component, you are done and ready to move on to running the SQL.



DbDesignGenerator.sh (11 of 11)

- SQL generated by default to these subdirectories:

```
<WAS_HOME>/util/dbUtils/DB2-zOS-xxx
```

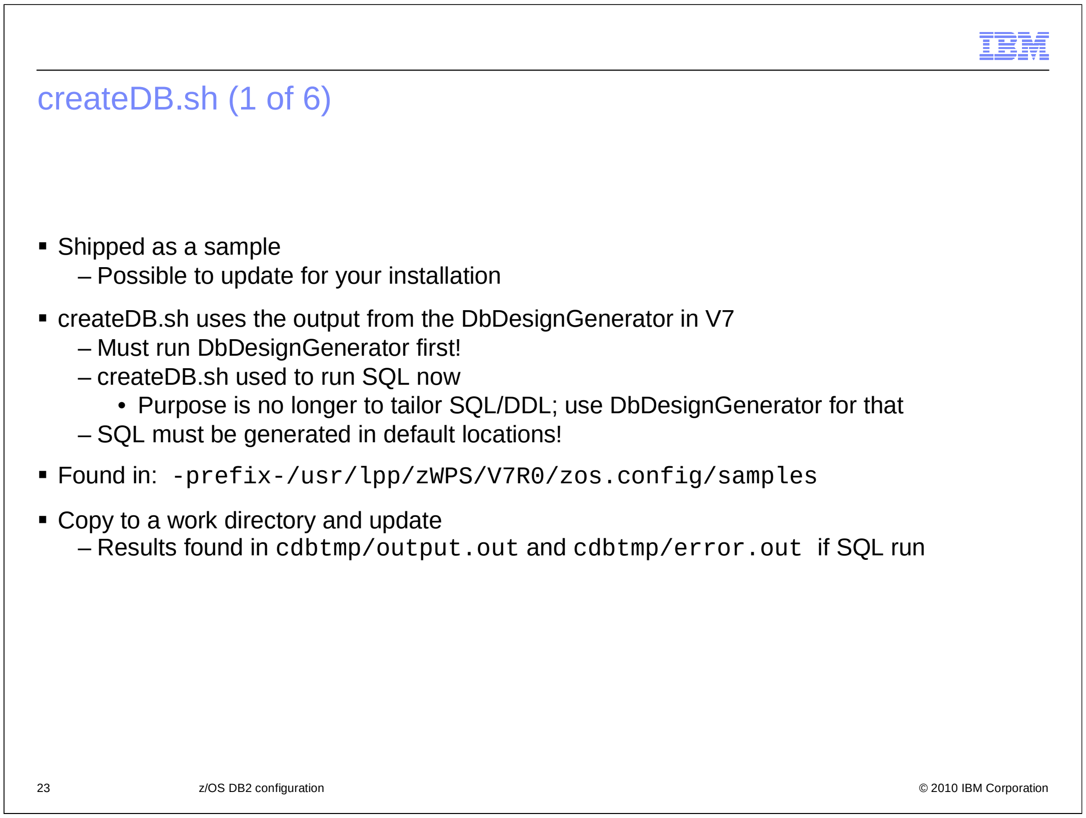
  – Using DB2 for z/OS V8 as an example, starting with 7.0.0.2, the directories are:

```
DB2-zOS-8-BPC
DB2-zOS-8-BPCReporting
DB2-zOS-8-BSpace
DB2-zOS-8-CommonDB
DB2-zOS-8-SibME
     WBI_BPC_ME
     WBI_CEI_ME
     WBI_SCA_APP_ME
     WBI_SCA_SYS_ME
```

  – Specify a full path to change the default location:

```
Please enter the output directory for WBI_BPC [default=DB2-zOS-8-BPC]
  :/u/wsuser/myBPC_SQL_directory
```

Here you see the default directories where SQL is generated. A directory for each component is created in the <WAS_HOME>/util/dbUtils directory by default. Here you see the default names for a DB2 for z/OS Version 8 installation. You can change where the SQL is generated by specifying a full path as seen on the slide.

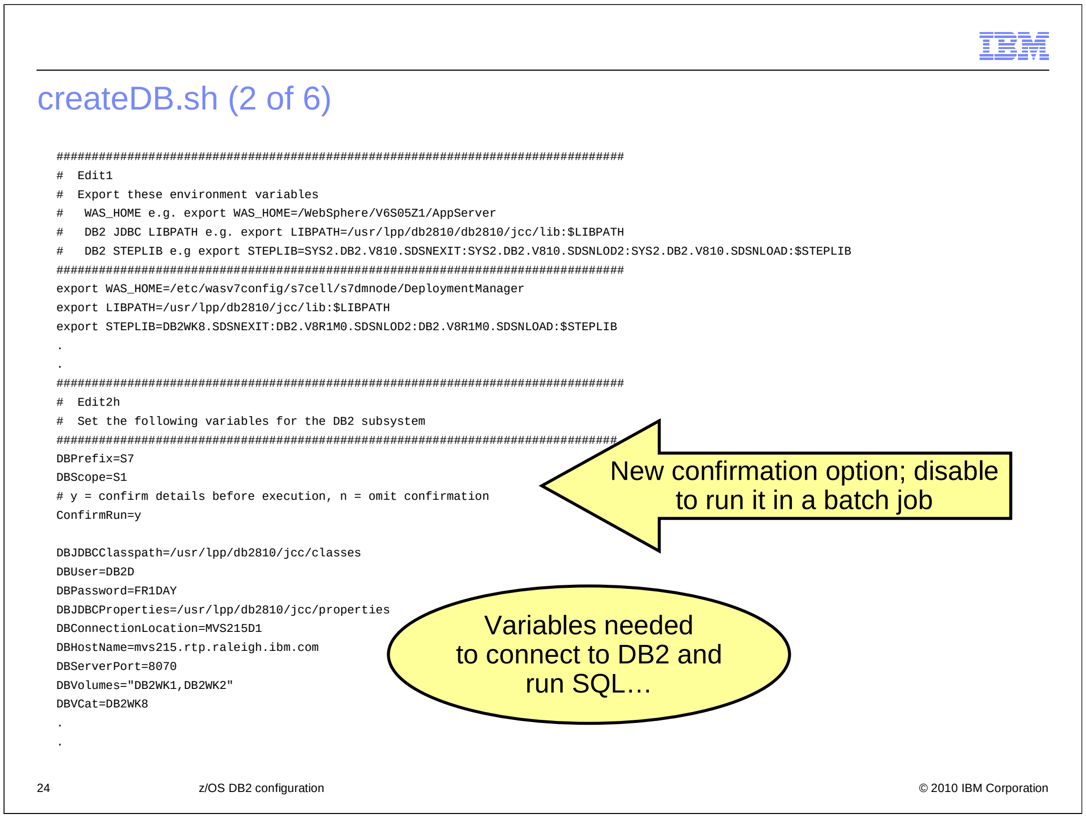


createDB.sh (1 of 6)

- Shipped as a sample
  - – Possible to update for your installation
- createDB.sh uses the output from the DbDesignGenerator in V7
  - – Must run DbDesignGenerator first!
  - – createDB.sh used to run SQL now
    - Purpose is no longer to tailor SQL/DDL; use DbDesignGenerator for that
  - – SQL must be generated in default locations!
- Found in: `-prefix-/usr/lpp/zWPS/V7R0/zos.config/samples`
- Copy to a work directory and update
  - – Results found in `cdbtmp/output.out` and `cdbtmp/error.out` if SQL run

The next tool, createDB, is shipped as a sample and you can update it as needed to conform to your installation's requirements. createDB.sh was used in prior versions of the product to actually create the tailored SQL and DDL and then optionally run it. In version 7, it can still optionally be used to run the SQL but you must use the DbDesignGenerator tool to tailor the SQL first. It is no longer responsible for the creation of the tailored SQL and DDL files, with the exception of the CEI component. You will see that on a later slide. In order for the createDB shell script to find the SQL generated by DbDesignGenerator, you must have saved it to the default directories unless you have updated the createDB sample to find it elsewhere.

The tool is found in the zos.config/sample directory in your SMP/E directory. In order to use it, you need to copy it to a work directory and update it for your installation. You'll see this on the next slide. After it has run, output and errors are found in the cdbtmp directory in the output.out and error.out files.



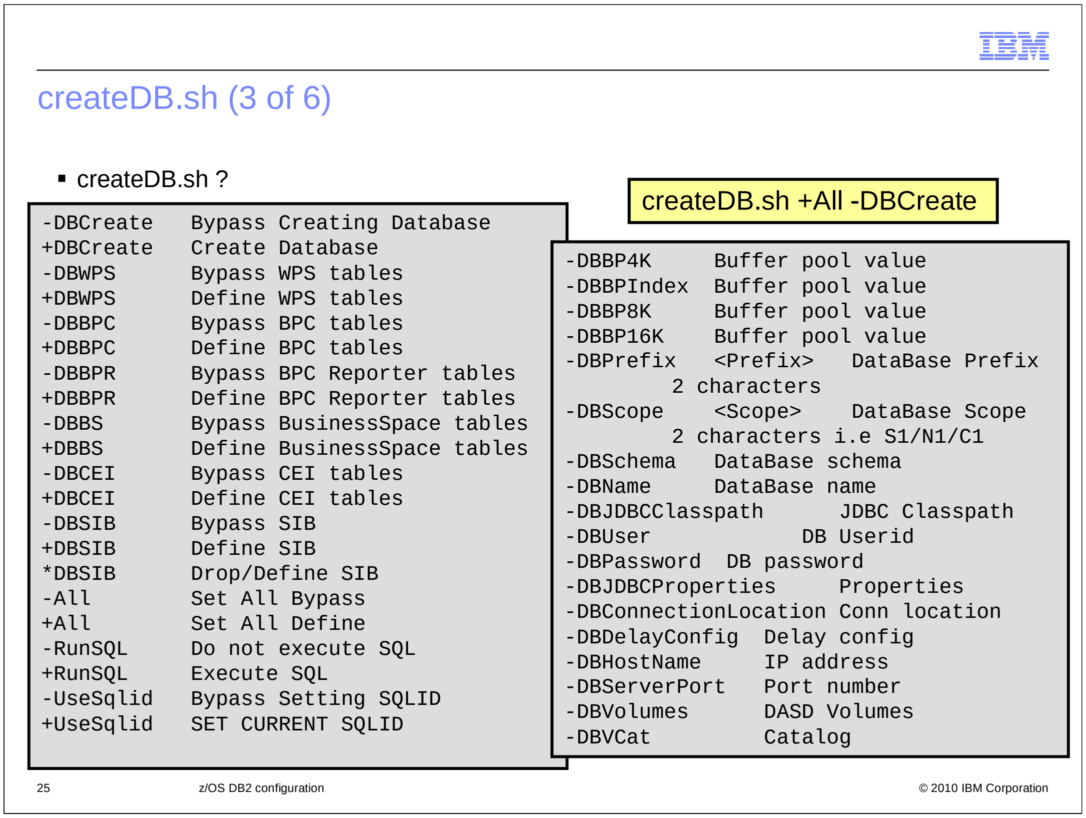
createDB.sh (2 of 6)

```
################################################################################
#  Edit1
#  Export these environment variables
#   WAS_HOME e.g. export WAS_HOME=/WebSphere/V6S05Z1/AppServer
#   DB2 JDBC LIBPATH e.g. export LIBPATH=/usr/lpp/db2810/db2810/jcc/lib:$LIBPATH
#   DB2 STEPLIB e.g export STEPLIB=SYS2.DB2.V810.SDSNEXIT:SYS2.DB2.V810.SDSNLOD2:SYS2.DB2.V810.SDSNLOAD:$STEPLIB
################################################################################
export WAS_HOME=/etc/wasv7config/s7cell/s7dmnode/DeploymentManager
export LIBPATH=/usr/lpp/db2810/jcc/lib:$LIBPATH
export STEPLIB=DB2WK8.SDSNEXIT:DB2.V8R1M0.SDSNLOD2:DB2.V8R1M0.SDSNLOAD:$STEPLIB
.
.
################################################################################
#  Edit2h
#  Set the following variables for the DB2 subsystem
###############################################################################
DBPrefix=S7
DBScope=S1
# y = confirm details before execution, n = omit confirmation
ConfirmRun=y

DBJDBCClasspath=/usr/lpp/db2810/jcc/classes
DBUser=DB2D
DBPassword=FR1DAY
DBJDBCProperties=/usr/lpp/db2810/jcc/properties
DBConnectionLocation=MVS215D1
DBHostName=mvs215.rtp.raleigh.ibm.com
DBServerPort=8070
DBVolumes="DB2WK1,DB2WK2"
DBVCat=DB2WK8
.
.
```

New confirmation option; disable to run it in a batch job

Variables needed to connect to DB2 and run SQL…

Here you see the portions of the createDB shell script that need to be updated. The first set of edits has to do with environment variables that are needed in the shell environment such as the location of your WAS_HOME. The second set of edits is used to define the DB2 environment in your installation. A new variable in version 7, ConfirmRun, allows you to disable the confirmation so that you can easily run the script from a batch job.



createDB.sh (3 of 6)

- createDB.sh ?

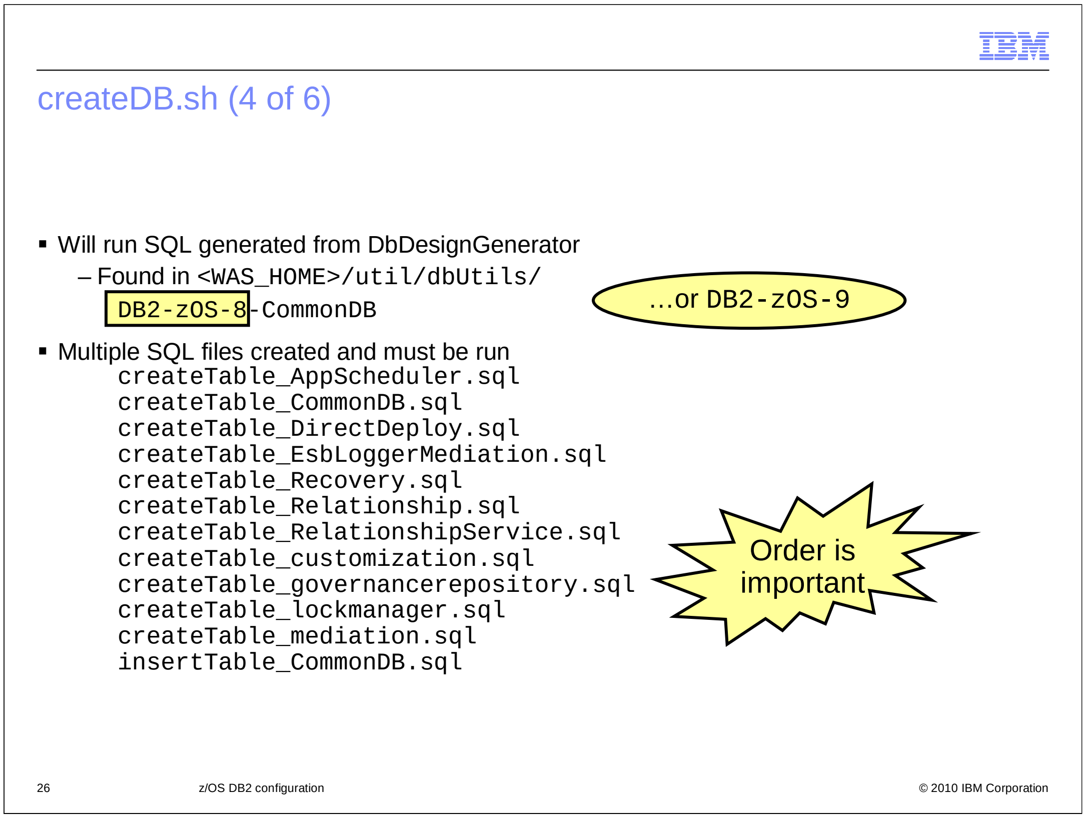
```
-DBCreate   Bypass Creating Database
+DBCreate   Create Database
-DBWPS      Bypass WPS tables
+DBWPS      Define WPS tables
-DBBPC      Bypass BPC tables
+DBBPC      Define BPC tables
-DBBPR      Bypass BPC Reporter tables
+DBBPR      Define BPC Reporter tables
-DBBS       Bypass BusinessSpace tables
+DBBS       Define BusinessSpace tables
-DBCEI      Bypass CEI tables
+DBCEI      Define CEI tables
-DBSIB      Bypass SIB
+DBSIB      Define SIB
*DBSIB      Drop/Define SIB
-All        Set All Bypass
+All        Set All Define
-RunSQL     Do not execute SQL
+RunSQL     Execute SQL
-UseSqlid   Bypass Setting SQLID
+UseSqlid   SET CURRENT SQLID
```

createDB.sh +All -DBCreate

```
-DBBP4K     Buffer pool value
-DBBPIndex  Buffer pool value
-DBBP8K     Buffer pool value
-DBBP16K    Buffer pool value
-DBPrefix   <Prefix>   DataBase Prefix
        2 characters
-DBScope    <Scope>    DataBase Scope
        2 characters i.e S1/N1/C1
-DBSchema   DataBase schema
-DBName     DataBase name
-DBJDBCClasspath      JDBC Classpath
-DBUser            DB Userid
-DBPassword  DB password
-DBJDBCProperties     Properties
-DBConnectionLocation Conn location
-DBDelayConfig  Delay config
-DBHostName     IP address
-DBServerPort   Port number
-DBVolumes      DASD Volumes
-DBVCat         Catalog
```

25 z/OS DB2 configuration © 2010 IBM Corporation

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, all SQL is run except the SQL to actually create the database and storage groups; the database and storage groups would already need to exist. The default behavior can be updated in the script itself by setting variables to 'true' or 'false'. For instance, the default behavior of the shipped sample is to define the WPS tables. In order to change that default behavior, you can update the DBWPSCreate variable in the script to be 'false'. As you recall, the job of createDB isn't actually to create the tailored SQL anymore so what does something like +DBWPS mean when you are not actually running the SQL? You will see that on the next slide.



createDB.sh (4 of 6)

- Will run SQL generated from DbDesignGenerator
  - Found in `<WAS_HOME>/util/dbUtils/DB2-zOS-8-CommonDB`

…or `DB2-zOS-9`

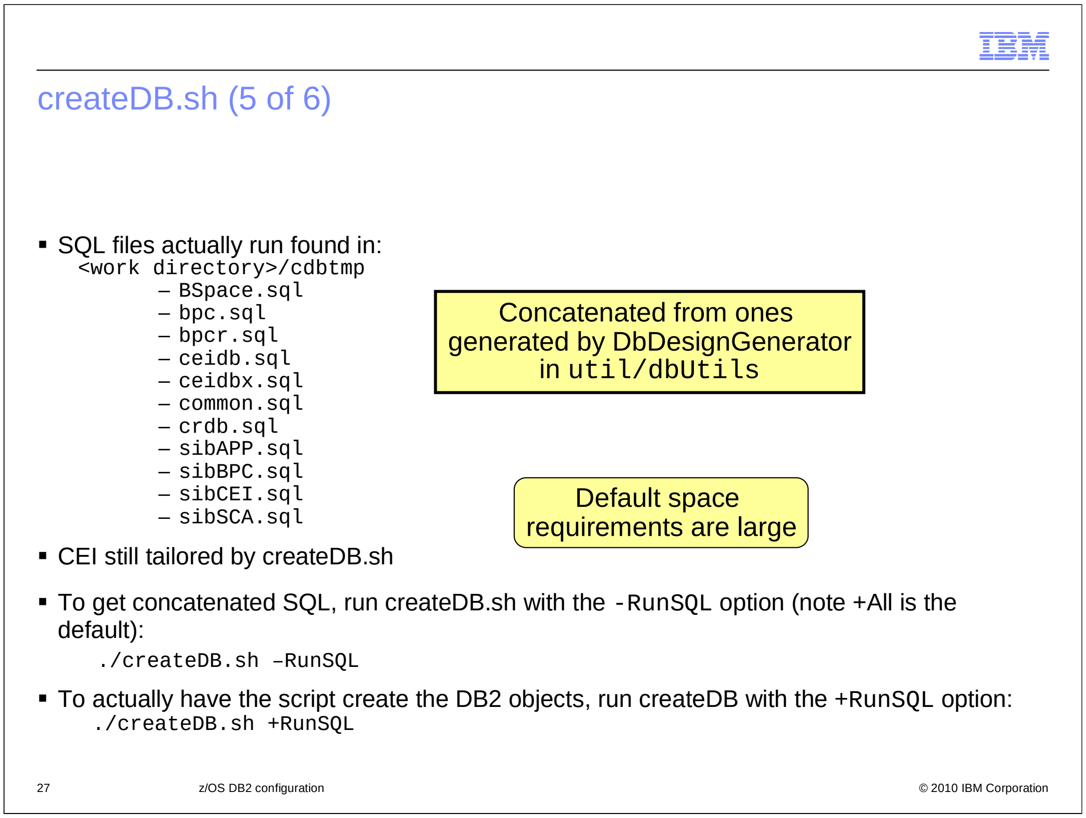
- Multiple SQL files created and must be run

```
createTable_AppScheduler.sql
createTable_CommonDB.sql
createTable_DirectDeploy.sql
createTable_EsbLoggerMediation.sql
createTable_Recovery.sql
createTable_Relationship.sql
createTable_RelationshipService.sql
createTable_customization.sql
createTable_governancerepository.sql
createTable_lockmanager.sql
createTable_mediation.sql
insertTable_CommonDB.sql
```

Order is important

26 z/OS DB2 configuration © 2010 IBM Corporation

The default location that the createDB shell script will look for the SQL to run is in component directories found in the util/dbUtils directory. The directory for the CommonDB is shown here. You will see that there are actually many SQL files that are created in that directory and need to be run. The order in which they are run can be important too. This is where the createDB helps out again.



createDB.sh (5 of 6)

- SQL files actually run found in:
  `<work directory>/cdbtmp`
  - `BSpace.sql`
  - `bpc.sql`
  - `bpcr.sql`
  - `ceidb.sql`
  - `ceidbx.sql`
  - `common.sql`
  - `crdb.sql`
  - `sibAPP.sql`
  - `sibBPC.sql`
  - `sibCEI.sql`
  - `sibSCA.sql`
- CEI still tailored by createDB.sh
- To get concatenated SQL, run createDB.sh with the `-RunSQL` option (note +All is the default):

```
./createDB.sh –RunSQL
```

- To actually have the script create the DB2 objects, run createDB with the `+RunSQL` option:

```
./createDB.sh +RunSQL
```

Concatenated from ones generated by DbDesignGenerator in `util/dbUtils`

Default space requirements are large

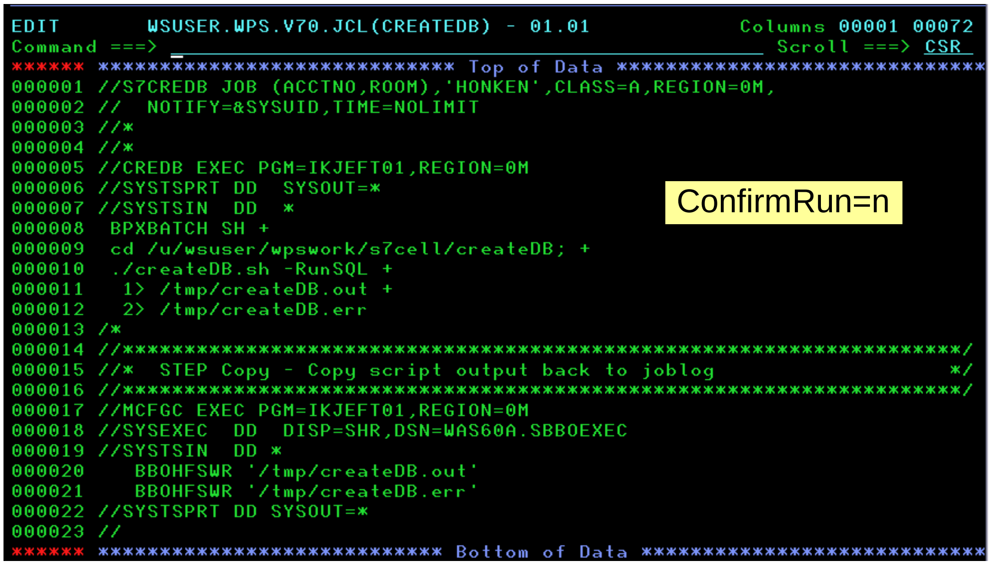
27 z/OS DB2 configuration © 2010 IBM Corporation

If you allow the createDB shell script to run the SQL, it actually runs the SQL found in the cdbtmp directory. It does a concatenation of all the SQL files created in each of the component directories created by dbDesignGenerator before actually running the SQL. You can use this to your advantage by running the concatenated SQL rather than having to run each of the SQL files created by the dbDesignGenerator script. The SQL scripts are ordered correctly and you are less likely to miss one.

The CEI component was not tailored while running the dbDesignGenerator script. That is created for you in the cdbtmp directory by the createDB shell script as ceidb.sql and ceidbx.sql. So, in order to get the tailored SQL concatenated by the createDB script without actually running it, you can use the command with the minus RunSQL parameter. That gives you the SQL files shown on the slide that can then be handed over to your DB2 administrator to run. Note that the default database uses many cylinders so you might want to adjust space requirements for non-production cells.

Finally, if you have the authority to run the SQL, you can specify plus RunSQL and it will tailor the CEI SQL for you, concatenate the SQL into the files shown and actually run it automatically.

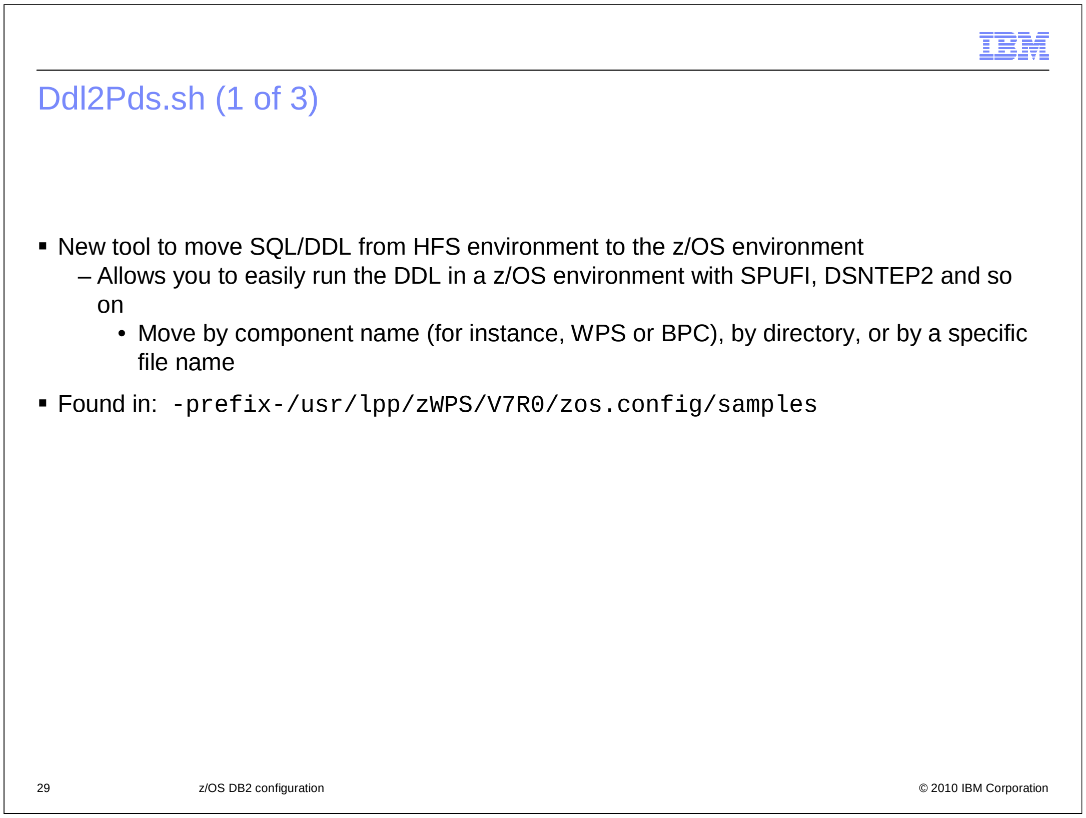
createDB.sh (6 of 6)



```
EDIT       WSUSER.WPS.V70.JCL(CREATEDB) - 01.01            Columns 00001 00072
Command ===> _                                                 Scroll ===> CSR
****** ***************************** Top of Data ******************************
//S7CREDB JOB (ACCTNO,ROOM),'HONKEN',CLASS=A,REGION=0M,
//   NOTIFY=&SYSUID,TIME=NOLIMIT
//*
//*
//CREDB EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD   SYSOUT=*
//SYSTSIN   DD   *
 BPXBATCH SH +
 cd /u/wsuser/wpswork/s7cell/createDB; +
 ./createDB.sh -RunSQL +
  1> /tmp/createDB.out +
  2> /tmp/createDB.err
/*
//***********************************************************************/
//*  STEP Copy - Copy script output back to joblog                      */
//***********************************************************************/
//MCFGC EXEC PGM=IKJEFT01,REGION=0M
//SYSEXEC   DD   DISP=SHR,DSN=WAS60A.SBBOEXEC
//SYSTSIN   DD *
   BBOHFSWR '/tmp/createDB.out'
   BBOHFSWR '/tmp/createDB.err'
//SYSTSPRT DD SYSOUT=*
//
****** **************************** Bottom of Data ****************************
```

ConfirmRun=n

Finally, here is an example of running the createDB shell script as a batch job, specifying that it should do everything but run the SQL for you. Remember, that the ConfirmRun parameter must be set to ‘n’ in the createDB script in order to run it in batch mode!

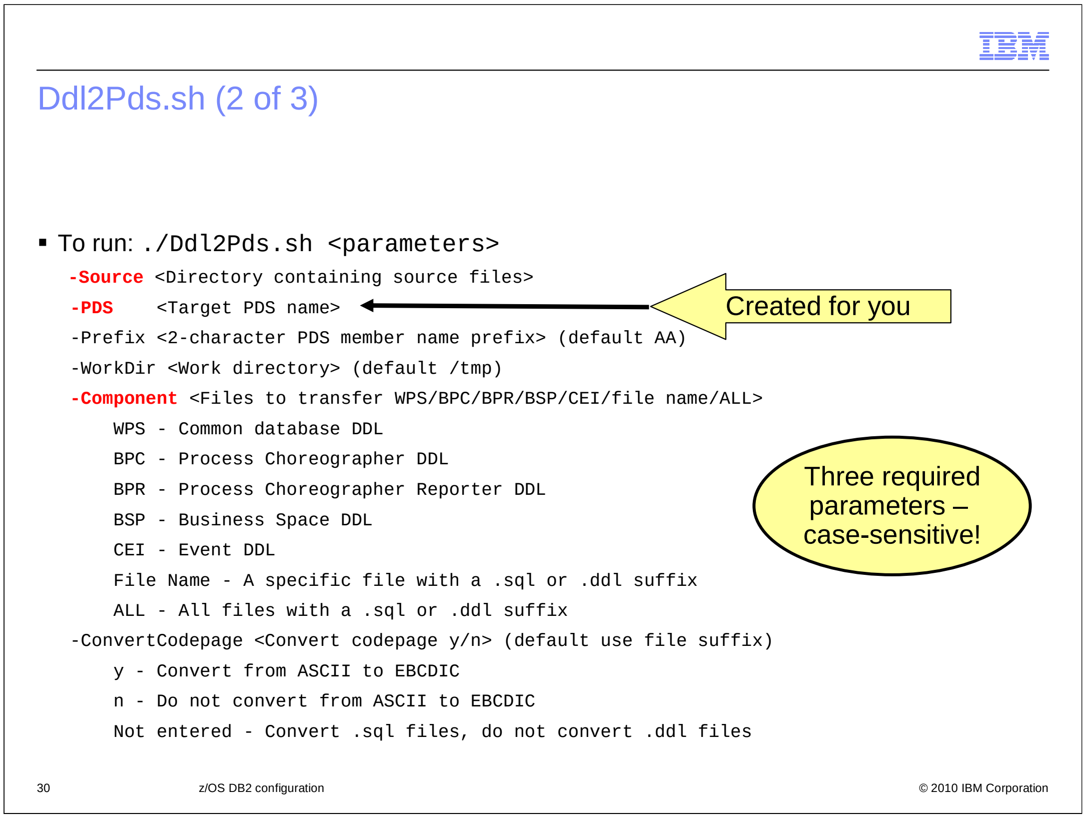


Ddl2Pds.sh (1 of 3)

- New tool to move SQL/DDL from HFS environment to the z/OS environment
  - Allows you to easily run the DDL in a z/OS environment with SPUFI, DSNTEP2 and so on
    - Move by component name (for instance, WPS or BPC), by directory, or by a specific file name
- Found in: `-prefix-/usr/lpp/zWPS/V7R0/zos.config/samples`

29 z/OS DB2 configuration © 2010 IBM Corporation

The last tool at your disposal that this presentation will talk about is the Ddl2Pds shell script. It allows you to easily move the SQL created by the DbDesignGenerator and createDB tools from the HFS environment over to the z/OS environment. Moving it to a PDS where DB2 administrators can use their favorite tools, such as SPUFI and DSNTEP2, to run the SQL can be tedious and error prone. Ddl2Pds worries about the codepage translation from ASCII to EBCDIC for you. The DB2 administrator can then concentrate on the creation of the WebSphere Process Server and WebSphere Enterprise Service Bus database objects for you. It is found in the zos.config/sample directory of your SMP/E directory.



Ddl2Pds.sh (2 of 3)

- To run: `./Ddl2Pds.sh <parameters>`

```
-Source <Directory containing source files>
-PDS    <Target PDS name>
-Prefix <2-character PDS member name prefix> (default AA)
-WorkDir <Work directory> (default /tmp)
-Component <Files to transfer WPS/BPC/BPR/BSP/CEI/file name/ALL>
    WPS - Common database DDL
    BPC - Process Choreographer DDL
    BPR - Process Choreographer Reporter DDL
    BSP - Business Space DDL
    CEI - Event DDL
    File Name - A specific file with a .sql or .ddl suffix
    ALL - All files with a .sql or .ddl suffix
-ConvertCodepage <Convert codepage y/n> (default use file suffix)
    y - Convert from ASCII to EBCDIC
    n - Do not convert from ASCII to EBCDIC
    Not entered - Convert .sql files, do not convert .ddl files
```

Created for you

Three required parameters – case-sensitive!

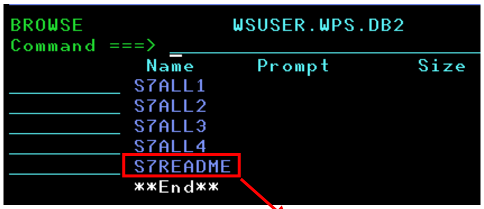
30 z/OS DB2 configuration © 2010 IBM Corporation

To run the tool, three parameters are required. The first one is ‘Source’ and for that you need to specify where your SQL or DDL files are found in the HFS. The second required parameter is ‘PDS’ which tells the tool what PDS to copy the DDL or SQL to in the z/OS environment. It is created for you if it does not already exist. The third required parameter is ‘Component’. This tells Ddl2Pds what files to transfer over to the z/OS environment. You can select a particular component or you can specify that you want to transfer all of the SQL and DDL files in the given directory over to the z/OS environment. They must have a DDL or SQL file extension. The ‘Prefix’ parameter is a prefix for the PDS members. You’ll see how this is used on the next slide. The tool needs a work directory for its processing so if you want to use something other than /tmp, you can specify the ‘WorkDir’ parameter. Finally, if you need to specify that you want to have Ddl2Pds convert from ASCII to EBCDIC you can specify the ConvertCodePage parameter. By default, the tool will convert SQL files but assumes that DDL files are already EBCDIC.

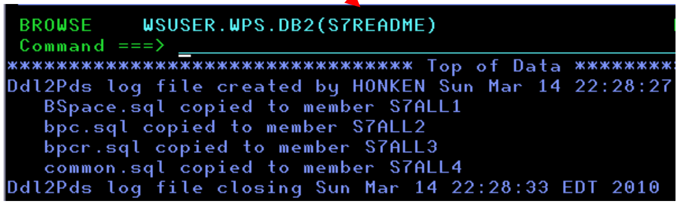
Ddl2Pds.sh (3 of 3)

- Example:

```
./Ddl2Pds.sh -Source /u/wsuser/wpswork/s7cell/cdbtmp -PDS 'wsuser.wps.db2' -Component ALL –Prefix S7

Ddl2Pds script executing ... Sun Mar 14 22:28:24 EDT 2010

Source  = /u/wsuser/wpswork/s7cell/cdbtmp
PDS Name = wsuser.wps.db2
PDS Member Prefix = S7
Work Directory = /tmp
Component = ALL
Codepage conversion = OnSuffix

Enter y to proceed, anything else to cancel
y

Creating wsuser.wps.db2

   Processing SQL file BSpace.sql

   Processing SQL file bpc.sql

   Processing SQL file bpcr.sql

   Processing SQL file common.sql

Ddl2Pds processing complete
```

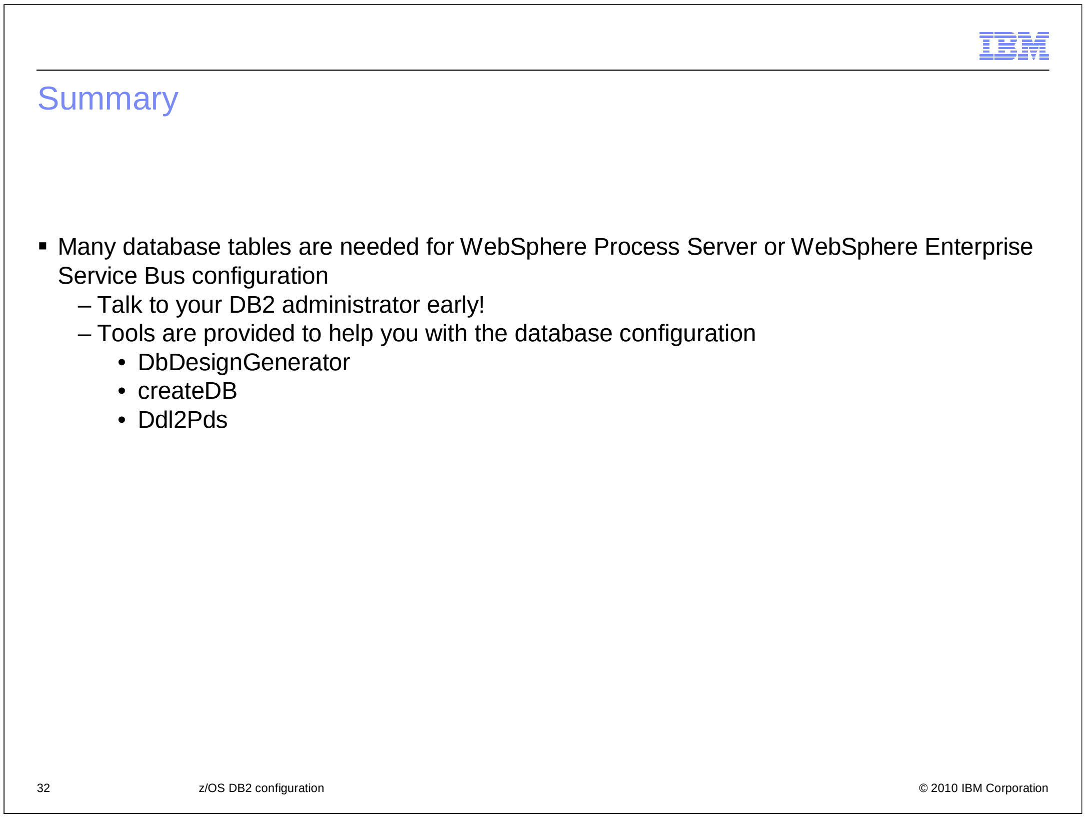


```
BROWSE                WSUSER.WPS.DB2
Command ===> _
                 Name         Prompt         Size
_____________ S7ALL1
_____________ S7ALL2
_____________ S7ALL3
_____________ S7ALL4
_____________ S7README
              **End**
```



```
 BROWSE     WSUSER.WPS.DB2(S7README)
 Command ===> _
********************************* Top of Data *********
Ddl2Pds log file created by HONKEN Sun Mar 14 22:28:27
    BSpace.sql copied to member S7ALL1
    bpc.sql copied to member S7ALL2
    bpcr.sql copied to member S7ALL3
    common.sql copied to member S7ALL4
Ddl2Pds log file closing Sun Mar 14 22:28:33 EDT 2010
```

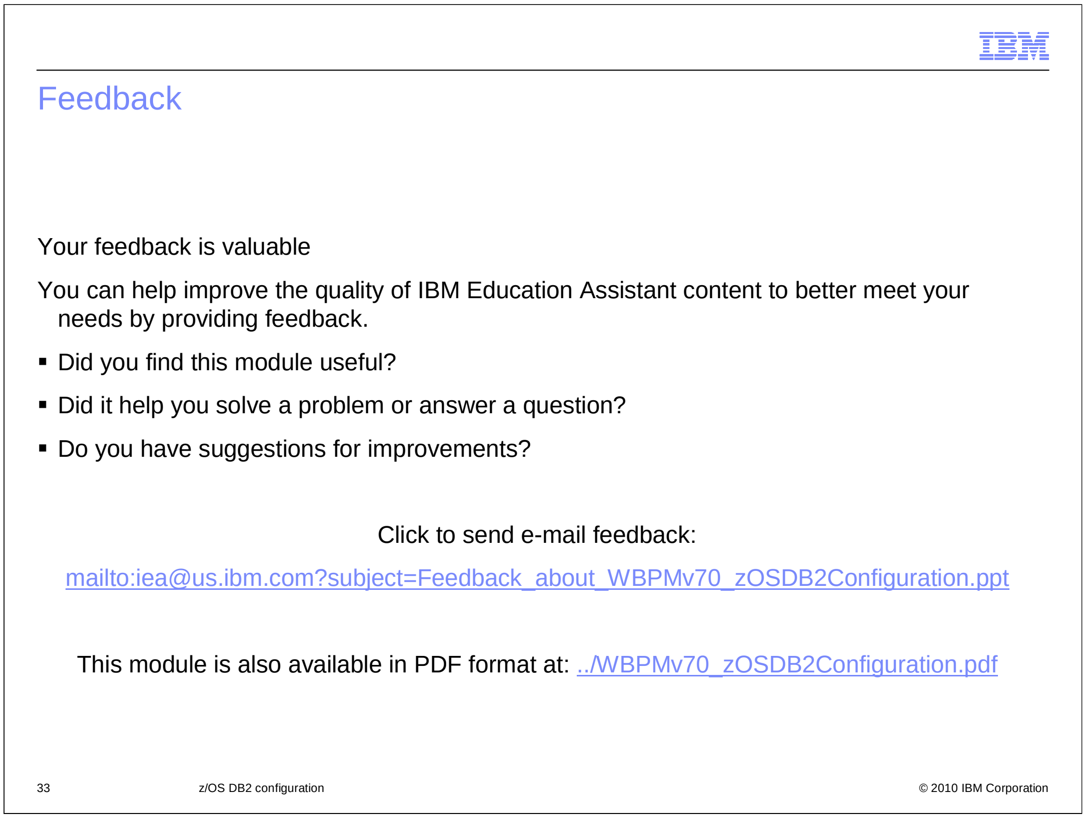
Shown here is an example of running the Ddl2Pds shell script. Notice that ‘ALL’ is specified for the component parameter so all DDL or SQL files in the specified source directory is copied to the specified PDS. Each file that matches the component you specify is translated to a member whose name starts with the prefix you have specified, followed by the component name you specified and ending in a number. To translate from the file name in the HFS to the PDS member name, you can look in the <prefix>README member in the PDS. Note that if you specify one of the components such as BPC or CEI, the tool expects the file names that are created, by default, by the DbDesignGenerator script.



Summary

- Many database tables are needed for WebSphere Process Server or WebSphere Enterprise Service Bus configuration
  - Talk to your DB2 administrator early!
  - Tools are provided to help you with the database configuration
    - DbDesignGenerator
    - createDB
    - Ddl2Pds

In summary, there are many tables that you need to have defined in DB2 to run WebSphere Process Server or WebSphere Enterprise Service Bus. Talk to your DB2 administrator early and decide on naming conventions. There are tools to help you with the database configuration. They are DbDesignGenerator, createDB and Ddl2Pds as seen in this presentation.



Feedback

Your feedback is valuable

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

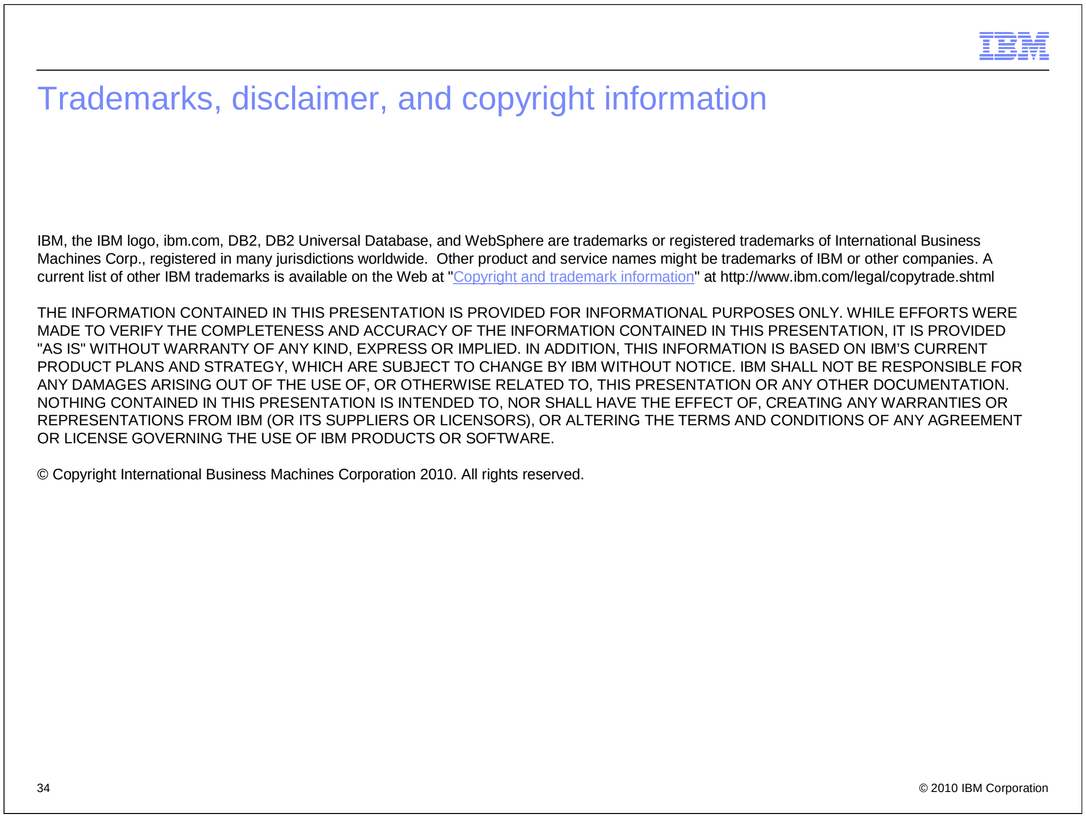
- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WBPMv70_zOSDB2Configuration.ppt

This module is also available in PDF format at: ../WBPMv70_zOSDB2Configuration.pdf

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



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, DB2, DB2 Universal Database, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.