System and SQL Performance Products for DB2
Version 10.1.00
November 15, 2012
Deploying System and SQL Performance Products

BMC is announcing changes in how users deploy the following System and SQL Performance for DB2 products and solutions:

- APPTUNE for DB2
- MainView for DB2 - Data Collector
- Pool Advisor for DB2
- SQL Performance for DB2
- System Performance for DB2

These changes accommodate installation enhancements that BMC introduced in version 2.3.40 of the Installation System. For these products and solutions only, use this bulletin as follows:

- As a replacement for the deployment information in the BMC Product and Solutions for DB2 Configuration Guide
- As a supplement to the Installation System User Guide

**Note**
If you have any questions, contact Customer Support at 1 800 537 1813 (United States or Canada) or call your local support center.

The following topics are discussed:

Deployment overview ...................................................................................................... 2
Preparing for installation .................................................................................................7
  Required minimum level of RTCS ............................................................................... 7
  RTCS system registry .................................................................................................. 8
  Cross-system coupling facility (XCF) groups ............................................................. 9
  DBC started task ........................................................................................................ 11
  DBC default groups ................................................................................................... 12
  $G65INIT and $U20INIT usage notes ...................................................................... 13
Deployment overview

Deploying version 10.1 of the System and SQL Performance for DB2 products to multiple environments requires careful planning. This overview introduces new infrastructure components and illustrates how you might combine these components into different configurations across LPARs.

Although each user's environment is unique, this document attempts to cover a variety of typical situations. For example, many users prefer to clone the setup from their original installation rather than execute the installation dialog again on other LPARs. The installation dialog can replicate the data sets from the original installation and transport them to LPARs on non-shared DASD, or copy the data sets to new names to be used in the same shared DASD sysplex.

In contrast, some users already have their own procedures; these users simply want to know what additional steps are required to set up the new components for different environments.

New infrastructure components

Version 10.1.00 of the BMC System and SQL Performance products for DB2 introduces the following new infrastructure components:

- Runtime Component System (RTCS)
- Next Generation Logger (NGL)
- DB2 Component Services (DBC)
- DB2 Product Management (LGC)

The following table compares the components for versions 6.2 and earlier to the components for versions 10.1 and later:

<table>
<thead>
<tr>
<th>Item</th>
<th>Versions 6.2 and earlier</th>
<th>Versions 10.1 and later</th>
</tr>
</thead>
<tbody>
<tr>
<td>Started task</td>
<td>DOMPROC started task</td>
<td>DBC$STC started task</td>
</tr>
<tr>
<td>Data Collector</td>
<td>DOM started task</td>
<td>DOM agent running under the DBC started task</td>
</tr>
<tr>
<td>Data Collector ID</td>
<td>DOMPROC SSID</td>
<td>DBC SSID</td>
</tr>
<tr>
<td>DOMPLEX name</td>
<td>DOMPLEX profile</td>
<td>DOMPLEX option set</td>
</tr>
<tr>
<td>APPTUNE filter</td>
<td>FILTER profile</td>
<td>FILTER option set</td>
</tr>
<tr>
<td>Output groups</td>
<td>Output groups</td>
<td>LOGSETS</td>
</tr>
<tr>
<td>Trace data sets</td>
<td>Trace data sets</td>
<td>LOGFILES</td>
</tr>
<tr>
<td>Archiving</td>
<td>DOMBCOPY batch job</td>
<td>NGLARCH started task</td>
</tr>
</tbody>
</table>

After installation of version 10.1 or later, each LPAR must have all of the following components:

- RTCS started task
- DBC started task
- *(optional)* NGLARCH started task
- LGC, NGL, and DOM agents defined and running under the DBC subsystem
- NGL private registry data set
- LGC private registry data set, which can be shared across multiple DBC subsystems
- DBC repository data set
- *(Pool Advisor only)* DCC$VARS and PMD$HIST data sets

You can combine these components into different configurations across LPARs, depending on whether you are sharing DBC groups, RTCS system registries, and LGC private registries.
If you have version 6.x of the System and SQL Performance products, you should replace each Data Collector with a DBC subsystem from version 10.1 or later. Also, you should convert each STATUS data set from the 6.x release to use a new LGC private registry from the 10.1 or later release.

Examples of possible deployment environments

Figure 1 on page 4 illustrates an environment in which different LPARs have different RTCS registries but share the LGC private registry and the DBC group name. In this example, both LPARs share the runtime data sets, LGC registry, DBC group, and DOMPLEX option set. The DOM and LGC agents and the DBC subsystem communicate across the coupling facility (XCF).

Figure 1: System with non-shared RTCS system registry, shared LGC private registry, and shared DBC group name
Figure 2 on page 5 illustrates an environment in which different LPARs share the RTCS registries, LGC private registry, and DBC group name. This example is similar to the previous one but demonstrates that the same setup can be achieved whether the RTCS system registry is shared or unshared.

**Figure 2: System with shared RTCS system registry, shared LGC test, and shared DBC group**

Figure 3 on page 6 illustrates an environment in which different LPARs have different RTCS registries, different LGC private registries, and different DBC group names. In this example, you want to keep TEST separate from PROD. They will use different runtime data sets, different LGC registries, different DBC groups, and different DOMPLEX option sets. You do not want LGC, DOM, and DBC to communicate across the coupling facility.
Choosing a configuration

The configuration that best suits your environment depends on:

- Whether RTCS is already configured (and if so, whether it is configured as shared or non-shared)
- How many DBC groups you want to have in a sysplex
- Whether you need to have more than one DBC started tasks running on the same LPAR
Note
If you decide to have just one DBC subsystem per LPAR, you might want to base your decision between implementing the configuration shown in Figure 1 on page 4 and Figure 2 on page 5 on whether RTCS has already been configured as non-shared or shared.

Carefully review the information in this document to help you plan how to install and deploy the performance products across your environment.

Preparing for installation

Reviewing this section before starting the installation will help you plan for deployment.

Note
This section assumes that you are using version 2.3.40 or later of the Installation System to install the products.

Required minimum level of RTCS

You must use RTCS BP00187 or later. Each LPAR will have only one RTCS started task running.

If you already have RTCS installed and running, issue the following command to determine which level of RTCS you are running:

```
F RTCS,PACKAGE MAINT KERNEL
```

The PACKAGE RMID for PROGRAM OBJECT=OSZKERNL must be BP00187 or higher. If it is not, obtain the current maintenance for RTCS.

Note
HOLD doc for RTCS PTFs might indicate that an IPL should occur after you apply maintenance.

If you are installing RTCS for the first time from Installation System version 2.3.40 or later, BP00187 will already be applied to RTCS. However, BMC recommends that you obtain all current maintenance for all FMIDs before deploying the product to other environments.
RTCS system registry

You need to have an RTCS started task on each LPAR where a DBC has a member. You can share the RTCS system registry among all or some of the LPARs in a sysplex.

Controlling whether a RTCS system registry is shared

If you are installing RTCS for the first time, review the ALOCRTCS step from the $GI0VSAM job and the OSZIN1nn member in your JCL data set (where nn is your SYSCLONE value). The ALOCRTCS step creates the RTCS system registry; this data set name is referenced on the SREGVLDS parameter in the OSZIN1nn member that RTCS uses at startup.

Depending on how the installation generates this data set name (as determined by specifying SYSTEM or SYSPLEX in the Specify runtime system values field), the name will include one of the following items. (For more information, see Table 1 on page 32.)

- The sysplex name as a node (if you indicated to generate at the sysplex level)
- SYSCLONE as part of a node in the name (if you indicated to generate at the system level)

The content of the OZIIN1nn member depends on whether you want to share the RTCS system registry, as follows:

- If you want to share the RTCS system registry across some or all of the LPARs in a sysplex, ensure that OZIIN1nn contains:
  - An SREGVLDS parameter that references the same system registry data set for those RTCS instances.
  - The NOPRIVATE-REGISTRY and SHARED-REGISTRY parameters
  - The same value for REGISTRY-XCF-GROUP to allow communication between these RTCS instances

- If you do not want to share the RTCS system registry with other RTCS instances in the sysplex, ensure that OZIIN1nn contains:
  - An SREGVLDS parameter that refers to a unique data set
  - The PRIVATE-REGISTRY and NOSHARED-REGISTRY parameters
  - No REGISTRY-XCF-GROUP value
Note
For more information about RTCS parameters, see the BMC Runtime Component System Configuration and Administration Guide.

The System and SQL Performance products can operate with a shared or non-shared RTCS system registry. However, in some situations, using a non-shared RTCS registry is better. For more information, see “Avoiding ZDBCssid” on page 54.

Determining whether an existing RTCS system registry is shared

If RTCS is already installed and running, you will need to review how the system registry is set up.

1 Issue the following console command to determine whether the RTCS system registry is shared across the LPARs in the sysplex:
   
   ```
   RO *ALL,F RTCS,REGISTRY STATUS
   ```

2 Look for message OSZ0240I in the output. If the same data set name is in the message from each LPAR, the registry is shared.

3 If different data set names are in the messages, the registry is not shared. Based on this finding, take one of the following actions:
   
   - If the registry is shared, run step LGCDFLT from the $G65INIT job from any LPAR in the sysplex to register all DBC SSIDs for the group.
   - If the RTCS system registry is not shared, run step LGCDFLT from the $G65INIT job on each LPAR where the DBC has a member.

Cross-system coupling facility (XCF) groups

The System and SQL performance products use the following cross-system coupling facility (XCF) groups:

- The DBC starts an XCF group based on the `<GROUP>` element in the $G65INIT and $U20INIT jobs.

You can specify a different name for the XCF group. To do so, use the `<XCFGROUP>myxcf</XCFGROUP>` element in the member referenced in the DBCPARNSS DD in the DBC$STC started task PROC. The variable `myxcf` represents the XCF group name to be used by the DBC.
The DOM agent starts an XCF group based on the DOMPLEX option set name specified in the <PARMS> element DOMPLEX= keyword in the DOM$STRT job. This XCF group should not be shared with other processes. If the DOMPLEX option set name is the same as the DBC GROUP name, the DOM code will add a $ sign to the end of the name (or change the last character to $) to make the name unique. Doing so ensures that the XCF group will be different from the one used by the DBC, even if the DBC group name matches the DOMPLEX option set name. The XCF group must be the same for all DOM agents in order to show data collected for remote DB2 subsystems.

The LGC agent starts an XCF group based on what you specify in the $G65INIT job (LGCDFLT step).

**WARNING**

Use caution with the @ symbol. This symbol can cause translation issues.

All three XCF groups must have different names.

**Example**

- Assume that the DBC group and DOMPLEX option set have the same name, DCPLEX. The group names could be as follows:
  DBC XCF group = DCPLEX
  DOM XCF group = DCPLEX$ (with DCPLEX automatically changed to DCPLEX$ by the DOM code)
  LGC XCF group = LGCGRP01

- Assume that the DBC group and DOMPLEX option set have different names. The group names could be as follows:
  DBC XCF group = DCPLEX
  DOM XCF group = TESTPLEX
  LGC XCF group = LGCTEST

To display XCF groups that are already in use, issue the following console command:

```
D XCF,GROUP
```

To determine the names of the XCF groups that are in use, review the DBC$STC started task log and search for XCF.

To enable communication across XCF groups for all three components from one LPAR to another, BMC recommends the following guidelines:
Both DBC subsystems should be defined to the same DBC group and should reside in the same sysplex.

Both DBC subsystems should join the same XCF group used by the DBC.

Both DOM agents should join the same XCF group as defined by using the same DOMPLEX option set.

Both LGC agents should join the same XCF group.

## DBC started task

Running the DBC$STC started task procedure starts the DBC subsystem.

The DBC SSID and NGL registry data set referenced in the started task procedure must match what you are using for this LPAR. The NGL registry is created in the ALOCNGL step in the $G10VSAM job. You can use the same started task member on each LPAR by specifying the SSID value in the startup command.

Figure 4 on page 11 illustrates that the DBC SSID is the same as the NGL PIID, so the &ssid variable is used for both.

### Figure 4: Example of DBC started task

```plaintext
//DBC    PROC  ACC=4111,         //SSID DC01,          //G=FDR1PLEX,         //T=NO,         //TIM=1440         //*
//DBC    EXEC PGM=DBCMAIN,REGION=0M,ACCT=&ACC,TIME=&TIM,          //PARM='SSID=&SSID,GROUP=&G,TRACE=&T'
//STEPLIB DD DSN=BMCPERF.TEST.BMCLINK,          //DISP=SHR
//DBCPRINT DD SYSOUT=* ,RECFM=VA     //SYSPRINT DD SYSOUT=* ,RECFM=VA     //BMCPSWD DD DISP=SHR,
//DSN=BMCPERF.TEST.BMCLINK
//SYSTSPRT DD SYSOUT=*
//SYSTEM DD SYSOUT=* ,RECFM=VA     //DBCPARMS DD DISP=SHR,
//DSN=BMCPERF.TEST.BMCSAMP($DBC&SSID)
//DBCSECUR DD DISP=SHR,
//DSN=BMCPERF.TEST.BMCSAMP($SEC&SSID)
//REGISTRY DD DISP=SHR,
//DSN=BMCPERF.TEST..&SSID.REGISTRY
//DOMTPLT DD DISP=SHR,
//DSN=BMCPERF.TEST.BMCTMPLT
//DOMPARMS DD DISP=SHR,
//DSN=BMCPERF.TEST.BMCPARM
```

Consider making the following changes:
Change the ACC= value at the top of the PROC (if necessary) for accounting information.

*(optional)* Add a step in the PROC to APF authorize anything in your STEPLIB, as shown in the following example:

```
//AUTH1    EXEC  PGM=JTHAPF00,ACCT=&ACC,PARM='DBC,STEPLIB'
```

The DBC$STC started task references two members for the DBC SSID:

- $DBC&ssid
- $SEC&ssid

For the initial installation, these members are customized into the UXXSAMP data set. If you chose runtime, the members are copied to the BMCSAMP data set, and the DBC started task will reference these runtime data sets. Replicate the members in the appropriate user or runtime data set to use the naming convention for the DBC SSID for this LPAR. Edit the $DBC&ssid member: either change the DBC repository name for this LPAR, or use the &SSID symbolic for the repository name in that member.

```
<NAME>BMCPERF.&SSID..DBCREPOS</NAME>
```

**Note**

The $SEC&ssid member contains only one parameter. If you want this parameter to be the same for all DBC subsystems in this DBC group, hard code the member and share it among all DBC subsystems.

---

**DBC default groups**

You can have only one default DBC group per RTCS system registry. After the default DBC group is established, changing it can have unexpected results.

If you are not certain which DBC group is the default, you can find the default by adding the following XML string as input to the LGCUTIL step:

```
<?xml version="1.0" standalone="yes"?>
<lgcutil version="101">
<list><dbc/></list>
</lgcutil>
```

In the LGCUTLPR output, the name of the default DBC group (DCPLEX in the following example) is marked with an asterisk.
If the DBC group name matches the group name of the DBC subsystem that is being registered, the default attribute should be set to yes; otherwise, it should be set to no. ALL DBC IDs in the same DBC group should have the same setting for the default attribute.

$G65INIT and $U20INIT usage notes

The $G65INIT and $U20INIT jobs define agents to the DBC subsystem and register products to the RTCS system registry. As generated by the installation, these jobs contain <LOADLIB> elements for each agent that starts under the DBC subsystem.

Depending on the option you chose for configuring user libraries and runtime data sets, multiple <LOADLIB> elements per agent might reference the specific data sets that the agent uses.

The DBC started task is meant to be a long-running address space service that would normally remain active for the life of an IPL, similar to the RTCS started task. The purpose of the <LOADLIB> elements is to enable bouncing an agent (not the DBC started task) when maintenance is applied. Version 10.1 of the products does not yet include this functionality; you must bounce the DBC subsystem when you apply maintenance. As more BMC products begin running as agents in the DBC subsystem, this feature will become more important. Then, the DBC started task should be bounced only when DBC maintenance is applied.

If the <LOADLIB> elements exist, those data sets will be first in the concatenation, ahead of the STEPLIB data sets when searching for the load module to load. The product will find its code in the <LOADLIB> element data sets. Otherwise, the code will be loaded from the data sets in the STEPLIB of the DBC$STC started task. If no <LOADLIB> elements or STEPLIB data sets are present, move the product data sets to LINKLIST. The order in which the product searches is as follows:

- <LOADLIB> elements, if present
- STEPLIB data sets, if present
- LINKLIST

In addition, the load library is referenced in the register step for APPTUNE for DB2 (step REGIOD in $U20INIT).

Determine which of the following maintenance strategies you want to use. BMC recommends Strategy 1.
Strategy 1

Run the product with an ALIAS on the LOADLIB. Maintenance is applied to a different load library, and the ALIAS name is switched to that library when maintenance is ready to be rolled out. Complete one of the following actions:

- Keep the <LOADLIB> elements but change the names to reflect the ALIAS names. Follow the instructions in “To change references to the load library” on page 15.

- Follow the instructions in “To remove references to the load library” on page 15 and complete one of the following actions:
  - Reference the ALIAS names in the STEPLIB of the DBC$STC started task procedure.
  - Remove the STEPLIB from the DBC$STC started task PROC, put the load modules in LINKLIST, and refresh LINKLIST.

Strategy 2

Run the product with LOADLIB1 and apply maintenance to LOADLIB2. To roll out maintenance in LOADLIB2, concatenate LOADLIB2 ahead of LOADLIB1. If problems arise, remove the LOADLIB2 data set to revert back to previous maintenance.

Follow the instructions in “To remove references to the load library” on page 15 and complete one of the following actions:

- Use the concatenation of data sets on the STEPLIB of the DBC$STC started task procedure.

- Remove the STEPLIB from the DBC$STC started task procedure, put the load modules in LINKLIST, and refresh LINKLIST.

Strategy 3

Run the product with the runtime data set LOADLIB1 and apply maintenance to the target libraries; then, copy maintenance into the runtime LOADLIB1 library. Complete one of the following actions:

- Ensure that the <LOADLIB> elements continue to reference the LOADLIB1 runtime data set.

- Follow the instructions in “To remove references to the load library” on page 15 and complete one of the following actions:
— Make no changes to the STEPLIB of the DBC$STC started task PROC. This
STEPLIB does not need to change because it will be pointing to LOADLIB1,
which will always be refreshed to point to the new maintenance.

— Remove the STEPLIB from the DBC$STC started task PROC, put the load
modules in LINKLIST, and refresh LINKLIST.

**To change references to the load library**

Use the following procedure if you want to keep the <LOADLIB> elements but
reference a different data set or change the LOADLIB to reference an ALIAS.

---

### Note

This procedure assumes that you have not yet run the $G65INIT or $U20INIT job. *If you have already run $G65INIT or $U20INIT, obtain the LOADCHG job from Customer Support and run it instead of using this procedure.*

---

1. Change the data set names in the <LOADLIB> elements.

2. *If you are installing APPTUNE or SQL Performance*, change the data set name in the <file id= "load"> element in the REGIOD step of the $U20INIT job.

   The referenced data set must be APF authorized.

**To remove references to the load library**

The $G65INIT and $U20INIT jobs reference the product LOADLIB in a few places.
Use the following procedure if you want to remove references to the LOADLIB data sets.

---

### Note

This procedure assumes that you have not yet run the $G65INIT or $U20INIT job. *If you have already run $G65INIT or $U20INIT, obtain the LOADDEL job from Customer Support and run it instead of using this procedure.*

---

1. Delete all <LOADLIB> lines in the job.

2. *If you are installing APPTUNE or SQL Performance*, delete the two <file id="value"> lines in the REGIOD step of the $U20INIT job.

   For example, change:

```xml
<files>
  <file id="load">BMCPERF.TEST.BMCLINK</file>
  <file id="iodexit" refid="load"/>
</files>
```
Communication between a DOM agent and an older release

As you phase in version 10.1 or later of the products and begin using the DBC, you can enable the product to report on data collected by earlier releases of the Data Collector. The following guidelines apply:

- The DBC group name must be different from the DOMPLEX profile name of the older release.

- The DOMPLEX option set name must be the same as the DOMPLEX profile name in the 6.x release.

Specify the DOMPLEX option set name in the DOM$STRT job by using the DOMPLEX= keyword. Ensure that you have a DOMPLEX option set with that name in the LGC private registry datastore before running the DOM$STRT job.

- If you want to have a version 10.1 or later DBC subsystem running on the same LPAR as a 6.x Data Collector, use a different ssid for the DBC than you used for that Data Collector.

For example, if you used DC01 for version 6.1, you could use DBC1 for version 10.1.

- Ensure that the output groups in the option set are correct for your DBC IDs. You can use the same high-level qualifier (HLQ) for the trace data sets because the low-level qualifiers (LLQs) in the versions differ.

Editing the DOMPLEX option set

You can edit the DOMPLEX option set either through an online interface or in batch mode.

Editing the option set for the first time can take a while because the NGL agent requires several values for each output group. BMC recommends editing the option set online (as instructed in Editing the DOMPLEX option set online on page 17) if either of the following situations applies to you:

- You are not migrating from a previous release.

- You are migrating from a previous release, but your DOMPLEX configuration contains a small number of output groups.
In contrast, consider editing in batch mode if you are migrating from a previous release and have a large number of output groups; for instructions, see Editing the DOMPLEX option set in batch mode on page 22.

**Editing the DOMPLEX option set online**

Use the following procedures to access the online interface and then use it to edit the DOMPLEX option set.

*Note*

BMC recommends editing the option set online if you are not migrating from an earlier release or you have a small number of output groups. Otherwise, you might consider editing in batch mode as instructed in Editing the DOMPLEX option set in batch mode on page 22.

**Before you begin**

Consider the following guidelines, and use them as a reference (if needed) while editing the DOMPLEX option set online.

- If you migrated from a previous release and navigate to a list of DOMPLEX option sets, the DOMPLEX name that you used in version 6.x will be listed in a panel similar to this sample:

<table>
<thead>
<tr>
<th>Command</th>
<th>File</th>
<th>Filter</th>
<th>Help</th>
<th>DOMPLEX Option Sets</th>
<th>Scroll =&gt;&gt; CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution/Product</td>
<td>Version Changed</td>
<td>More: +</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System and SQL Performance for DB2</td>
<td>V10.1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCPLEX</td>
<td>TEST ENVIRONMENT</td>
<td>2012/04/12 13:36:06 RDADAC</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Tip*

You can select a DOMPLEX option set for modification by typing `E` in the field beside that DOMPLEX and pressing Enter. Similarly, you can create a new option set by entering `I` next to the product or solution name. To create a new option set by copying an existing one, enter `C` in the field next to the existing option set.

Maintenance can affect layout and display of option sets. Displays may be different.
Figure 5 on page 18 shows the required fields in a DOMPLEX option set.

**Figure 5: DOMPLEX option set fields**

<table>
<thead>
<tr>
<th>Filter: Required</th>
<th>More:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td></td>
</tr>
<tr>
<td>- Security data set DSN.</td>
<td>(DOMPLEX) parameters apply to entire DOMPLEX</td>
</tr>
<tr>
<td>- Archive directory DSN.</td>
<td>(DOMPLEX) parameters apply to entire DOMPLEX</td>
</tr>
<tr>
<td>- Data Collector List</td>
<td>Data Collector(DBC) subsystems in DOMPLEX</td>
</tr>
<tr>
<td>- DC01</td>
<td>Data Collector SSID</td>
</tr>
<tr>
<td>- DB2 Monitor List</td>
<td>DB2 Sub-systems to be monitored</td>
</tr>
<tr>
<td>- DB2A</td>
<td>DB2 SSID</td>
</tr>
<tr>
<td>- SQL Performance/APPTUNE options</td>
<td></td>
</tr>
<tr>
<td>- APPTUNE Filter Name.</td>
<td>IBMCTEST (blank: test/prod deflts)</td>
</tr>
<tr>
<td>- OutGrp DCID DspSize</td>
<td>Output Groups - valid range: 001-256</td>
</tr>
<tr>
<td>- LDS Allocation type.</td>
<td>(SMS, VOL, NONE)</td>
</tr>
</tbody>
</table>

- If you access the interface from the installation instead of from a product CLIST, certain symbolic variables will get substituted automatically; otherwise, you might receive additional validation errors.

- While editing the option set, you might receive validation error messages if fields contain invalid entries. Validation errors change the filter setting to Filter:Invalid, and only fields that have validation errors are displayed. When the errors are corrected, the filter reverts to its previous setting (for example, Filter:Required or Filter:Off).

- You can stop an editing session before you are finished and save your changes, even if the option set still has validation errors. To do so, press **F3** and select **Save With Errors**.

- Option sets can also contain zoom fields that use the > (greater than) sign. You can zoom on the field by positioning the cursor on > and pressing **Enter**. All of the data for this field will then be shown.

- If you press **F1** while the cursor is positioned on an input or output field on a panel, specific information about that field is displayed. To view general information or information about a panel, use the Help menu at the top of the panel.

**To access the online interface**

Use this procedure to access the online interface for editing the DOMPLEX option set.

1. Invoke the appropriate CLIST, as follows:
   - Use BMCINSTL for initial installations or TDS installations on shared DASD.
   - Use TDSINSTL for TDS installations on non-shared DASD.
Use DOMCLIST or SPDCLIST (generated in the installation JCL data set) or hyperlink from MainView to launch the product.

2 If you invoked BMCINSTL or TDSINSTL CLIST, complete this step:
   a Select Product Customization.
   b Select whichever option you performed for runtime data sets, and specify values for Specify product customization.
   c From the Process Menu for Customizing Initial Runtime Instance, choose Customize product options under the DB2 product configuration heading.
   d If you are migrating from an earlier release, select the DOMPLEX option set name that matches the earlier release's DOMPLEX profile name. Otherwise, insert a new DOMPLEX with the same name that you specified in the installation dialog. (This name will match the DOMPLEX parameter in the DOM$STRT job.)

   Note
   When you invoke this dialog for the first time from the installation instead of from the product CLIST, certain symbolic variables will be substituted automatically; otherwise, you might get validation errors, requiring you to specify values that could have been substituted automatically. By default, the installation dialog shows only the required options. To see all options when editing the option set, type FILTOFF on the command line.

3 If you invoked DOMCLIST or SPDCLIST from your JCL or CLIB data set (or hyperlinked from MainView), complete this step:
   a Select Administration => DOMPLEX Option Sets.
   b Select the DOMPLEX option set name that matches your previous release DOMPLEX profile name.

   Tip
   For a new option set, you would insert a new DOMPLEX with the same name as the one that you specified in the installation dialog. (This name must match the DOMPLEX parameter in the DOM$STRT job.)

To edit the DOMPLEX option set online

Use this procedure to edit the DOMPLEX option set through the online interface.
Before you begin, access the interface as instructed in To access the online interface on page 18.

When working in the interface, you might need to type FILTOFF on the command line to see all of the fields.

1. In the interface, navigate to the DOMPLEX Options Set panel.

2. Select a DOMPLEX option set for modification by typing E in the field beside that DOMPLEX.

The panel displays sections in that option set:

<table>
<thead>
<tr>
<th>Filter: Required</th>
<th>More:</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ DOMPLEX Parameters</td>
<td>Parameters that apply to entire DOMPLEX</td>
</tr>
<tr>
<td>+ Data Collector List</td>
<td>Data Collector(DBC) subsystem SSIDs in DOMP</td>
</tr>
<tr>
<td>+ DB2 Monitor List</td>
<td>DB2 Sub-systems to be monitored</td>
</tr>
<tr>
<td>+ Output Groups</td>
<td>Historical data output group definitions</td>
</tr>
</tbody>
</table>

Tip

You can expand a section by selecting the plus sign and pressing Enter, or by typing S over the sign. (To collapse an expanded section, select the minus sign and press Enter, or type S over the minus sign.)

3. Expand the DOMPLEX Parameters section, and ensure that you have the correct SECURITY and COPYDIR data sets for version 10.1 or later listed.

This section contains values that apply to the entire DOMPLEX.

4. Expand the Data Collector List section, and ensure that the DBC ID is specified as the Data Collector.

This section lets you define the initialization parameters for each Data Collector (for example, the number of concurrent batch and online users allowed).

Tip

The Data Collector List, DB2 Monitor List, and Output Groups sections contain collections of repeating groups and support the following commands:

- I inserts a new instance in the repeating group.
- D deletes an instance.
- R replicates an instance.

5. If you have Pool Advisor, perform the following steps:
a Expand the **Data Collector List** section.

b Open each Data Collector ID listed, and zoom into both of the following field names by putting your cursor on the > sign and pressing **Enter**:

- **Data Collectors advisor variable repository**
- **Pool Advisor history repository**

**Note**

Ensure that these repositories exist before bringing up the DBC subsystem. Use the PMDHIST and PMDJINST members from the SAMP data set (BBSAMP or BMCSAMP) to create them.

6 Expand the **DB2 Monitor List** section, open each DB2 subsystem, and change the Dynamic Explain plan name to the correct plan name (DAA101D1) for 10.1.

This section lets you identify and define the DB2 subsystems that can be monitored by the Data Collectors in the DOMPLEX.

7 Expand the **Output Groups** section, open each output group, and check that the Data Collector SSID is correct and matches the DBC SSID that you plan to use on each LPAR in this sysplex.

This section lets you define the output groups that will be used to buffer trace records, and to define and allocate log files to which records will be written from the output groups.

**Note**

Do not use an asterisk (*) for the Data Collector ID. If you have a DB2 subsystem that comes up on different LPARs, define it to all of the Data Collectors that might monitor it.

8 Expand the **NGL LOGSET Parameters** section, define a value for the volume or SMS data, and zoom the DSN Prefix to ensure that a high-level qualifier (HLQ) is defined.

<table>
<thead>
<tr>
<th>NGL LOGSET Parameters</th>
<th>LOGSET attributes used by this group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logset time span. . . . 7D (nD,nH,nM)</td>
<td></td>
</tr>
<tr>
<td>Max log buffers. . . . 10 (2-20)</td>
<td></td>
</tr>
<tr>
<td>Max read buffers . . . 4 (2-99)</td>
<td></td>
</tr>
<tr>
<td>Deferred write time. . . 60 (1-999 sec.)</td>
<td></td>
</tr>
<tr>
<td>Minimum log file data sets (LDS) 2 (1-99)</td>
<td></td>
</tr>
<tr>
<td>Maximum log file data sets (LDS) 5 (1-99)</td>
<td></td>
</tr>
<tr>
<td>Space to allocate (per LDS). . . 100 (1-9999 MB) note: 1 CYL=720KB</td>
<td></td>
</tr>
<tr>
<td>LDS Allocation type. . . (SMS, VOL, NONE) (required for type=VOL)</td>
<td></td>
</tr>
<tr>
<td>LDS DFSMS Data class. . . (required for type=SMS)</td>
<td></td>
</tr>
<tr>
<td>LDS DFSMS Management class . . (required for type=SMS)</td>
<td></td>
</tr>
<tr>
<td>LDS DFSMS Storage class . . (required for type=SMS)</td>
<td></td>
</tr>
<tr>
<td>LDS DSN prefix. . . . . (required for type=SMS)</td>
<td></td>
</tr>
</tbody>
</table>
BMC recommends using SMS values that will have access to many volumes. If you specify a volume, BMC recommends that you specify a group name that would have access to a pack of volumes. If your SMS rules are set based on the HLQ in the DSN Prefix field, you should specify NONE for LDS Allocation Type and you do not need to fill in any of the LDS Volume or SMS fields. If validation errors indicate that a value for one of them is required, press F3 and choose Save With Errors.

9 If you plan to archive the trace data, review these fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Archiving</td>
<td>N (Y=Yes,N=No)</td>
</tr>
<tr>
<td>Archive post processing job</td>
<td>(optional) DOMPARM PDS member</td>
</tr>
<tr>
<td>Max days to keep archives</td>
<td>0 (1-999 Days,0=NoLimit)</td>
</tr>
<tr>
<td>Max number archives to keep</td>
<td>0 (1-999 Data sets,0=NoLimit)</td>
</tr>
<tr>
<td>Max combined size of archives</td>
<td>0 (1-999999 Mb,0=NoLimit)</td>
</tr>
<tr>
<td>Archive file Allocation type</td>
<td>(SMS, UNIT, VOL, NONE)</td>
</tr>
<tr>
<td>Archive Volume</td>
<td>(required for type=VOL)</td>
</tr>
<tr>
<td>Archive Unit</td>
<td>(required for type=UNIT)</td>
</tr>
<tr>
<td>Archive DFSMS Data class</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>Archive DFSMS Management class</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>Archive DFSMS Storage class</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>Archive DSN prefix</td>
<td>___________________</td>
</tr>
<tr>
<td>Alternate full archive DSN</td>
<td>___________________</td>
</tr>
</tbody>
</table>

10 Expand and review Subsystems supported by this group to ensure that the correct DB2s are listed for this data collector.

An asterisk (*) in this field indicate that all DB2 subsystems in the DB2 Monitor List that are active on the LPAR will be monitored.

11 Press F3 to save the option set.

12 To save a copy of your edited option set as a backup, use the export job cited in the “Editing the DOMPLEX option set in batch mode” on page 22 procedure.

Editing the DOMPLEX option set in batch mode

Use the following procedure to edit the DOMPLEX option set in batch mode. You will export the option set to a data set, edit the XML with a REXX exec, and import the edited option set back into the datastore.

Note

Editing in batch mode is recommended if you are migrating from an earlier release and have a large number of output groups. Otherwise, BMC recommends online editing for the option set as instructed in “Editing the DOMPLEX option set online” on page 17.

Before you begin
Ensure that the RTCS started task is running before performing this task.

When editing JCL, ensure that your CAPS and NUM keys are off, and that numbers do not appear to the right. Do not change the case of the XML.

Make sure that the Pool Advisor PMD$HIST and DCC$VARS data sets exist before bringing up the DBC subsystem. Use the PMD$HIST and PMD$INST members from the SAMP data set to create them.

### To edit the DOMPLEX option set in batch mode

1. Make a backup copy of the option set.

2. To export the option set from the LGC private registry datastore to a sequential data set, edit and run the JCL shown in Figure 6 on page 23, as follows:

   **Figure 6: JCL for exporting an option set**

   ```
   //EXPORT   EXEC PGM=LGCUTIL
   //STEPLIB  DD DISP=SHR,DSN=<product_loadlib>
   //SYSPRINT DD SYSOUT=*  
   //LGCUPLTR DD SYSOUT=*  
   //LGCUPLER DD SYSOUT=*  
   //STDOUT DD SYSOUT=*    
   //STDERR DD SYSOUT=*    
   //SYSTEM DD SYSOUT=*    
   //SYSDUMP DD DUMMY SYSOUT=*  
   //EXPOPT DD DISP=(NEW,CATLG),UNIT=SYSDA,  
   // SPACE=(CYL,(10,10),RLSE),  
   // DCB=(RECFM=VB,LRECL=255,BLKSIZE=27998),  
   // DSN=<export_option_set_dsn>
   //STDIN *  
   <?xml version="1.0" standalone="yes"?>
   <lgcutil version="1010">  
   <export format="xml">   
   <optionset>   
   <prodcode>DOM</prodcode>   
   <prodvrsn>1010</prodvrsn>   
   <name>PLEXname</name>   
   </optionset>  
   </export>  
   </lgcutil>  
   
   a. Replace `product_loadlib` with the name of your product load library.

   b. Replace `export_option_set_dsn` with the name of the data set to which you want to export the option set.

   c. Replace `PLEXname` with the name of the option set that you want to export.

   This name should match the DOMPLEX option set name for this DBC. (This name is shown in the list when you execute the DOMCLIST or SPDCLIST—or hyperlink from Mainview—and select Administration => DOMPLEX Option sets.)
3. Run the JCL.

3. Edit the exported option set to make any necessary changes for version 10.1 of your product.

Observe the following guidelines:

- Some fields use substitution variables that populate a value when you view the option set through the installation dialog. If you view the option set outside the installation dialog, these fields might appear to be blank, to contain null, or to contain symbolic variables in the migrated option set. The following example shows a REXX Edit Macro that would change all of the occurrences of DAA620D1 to DAA101D1 for the product plan name, and set the SECURITY and COPYDIR data sets.

```
ISREDIT
MACRO
ISREDIT  C '<domauth>' '<domauth>BMCPERF.SECURITY' ALL
ISREDIT  C '<dombarc>' '<dombarc>BMCPERF.COPYDIR' ALL
ISREDIT  C 'DAA620D1' 'DAA101D1' ALL
ISREDIT
SAVE
ISREDIT
END
```

- Look at the exported option set to find out what the XML elements are. Some elements are in sections that can repeat.

- The following fields in the **NGL LOGSET Parameters** section are required and might not have defaulted to values from a migration:

<table>
<thead>
<tr>
<th>NGL LOGSET Parameters</th>
<th>LOGSET attributes used by this group</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDS Allocation type. . . . . . . . . . .</td>
<td>(SMS, VOL, NONE)</td>
</tr>
<tr>
<td>LDS Volume. . . . . . . . . . . . .</td>
<td>(required for type=VOL)</td>
</tr>
<tr>
<td>LDS DFSMS Data class. . . . . . . .</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>LDS DFSMS Management class. . . . . .</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>LDS DFSMS Storage class . . . . . .</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>LDS DSN prefix . . . . . . . . .</td>
<td></td>
</tr>
</tbody>
</table>

- If you plan to archive the trace data, specify values for the following fields in the **NGL** section:

<table>
<thead>
<tr>
<th>Enable Archiving . . . . . . . . . . . . Y</th>
<th>(Y=Yes,N=No)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Archive file Allocation type . . . . . .</td>
<td>(SMS, UNIT, VOL, NONE)</td>
</tr>
<tr>
<td>Archive Volume . . . . . . . . . . .</td>
<td>(required for type=VOL)</td>
</tr>
<tr>
<td>Archive Unit . . . . . . . . . . .</td>
<td>(required for type=UNIT)</td>
</tr>
<tr>
<td>Archive DFSMS Data class . . . . . .</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>Archive DFSMS Management class . . . .</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>Archive DFSMS Storage class . . . . .</td>
<td>(required for type=SMS)</td>
</tr>
<tr>
<td>Archive DSN prefix . . . . . . . .</td>
<td></td>
</tr>
<tr>
<td>Alternate full archive DSN &gt;</td>
<td></td>
</tr>
</tbody>
</table>

4. Review your edited option set to ensure that the REXX exec made the correct changes.
To import the edited option set, edit and run the JCL shown in Figure 7 on page 25, as follows:

**Figure 7: JCL for importing an option set**

```xml
<?xml version="1.0" standalone="yes"?>
<lgcutil version="1010">
  <import replace="yes">
    <ddname>IMPOPT</ddname>
  </import>
</lgcutil>
```

a Replace `product_loadlib` with the name of your product load library.

b Replace `exported_option_set` with the name of the data set that contains the option set that you want to import.

6 Using online navigation, select the option set after importing it to ensure that no validation errors occur. (If errors occur, correct them.)

7 To save a copy of your edited option set as a backup, export the option set as described in Step 2 on page 23.

### Switching to the DBC from the version 6.x Data Collector

Different options are available for switching from the previous release (version 6.1 or 6.2) of the Data Collector to version 10.x of the product that uses the DBC subsystem to monitor DB2 subsystems.

You can use either of the following methods:

- **Replace all Data Collectors.**
  BMC recommends that you totally replace the version 6.x Data Collector with the version 10.1 DBC when you roll out version 10.x of your products to each LPAR. For more information, see “Replacing all Data Collectors” on page 26.
Switch Data Collectors on specific DB2 subsystems to use DBC.

Switch specific DB2 subsystems from using the version 6.x Data Collectors to the DBC as the $C jobs are run for that subsystem. For more information, see “Switching the Data Collector on a specific DB2 subsystem” on page 27.

Replacing all Data Collectors

Use this procedure to replace all version 6.x Data Collectors with the version 10.x DBC when you roll out version 10.x of your products to each LPAR.

Before you begin

Review the following considerations:

- You must switch all DB2 subsystems that are monitored by a version 6.x Data Collector to being monitored by a DBC subsystem at the same time.
- The ID for the 6.x Data Collector becomes the DBC subsystem ID (ssid). The 6.x DOMPLEX profile name becomes the DBC DOMPLEX option set name.
- Before the DBC subsystem is started on this LPAR, you must execute all $C jobs on all of the DB2 subsystems that are being monitored by the DBC.

To replace all Data Collectors

1. Shut down the 6.x Data Collector.

2. If you plan to reuse the 6.x Data Collector ID from the older release as the DBC SSID, edit and run the #SSIDMIG job in the DOMSAMP or BBSAMP data set before starting the DBC subsystem:
   - Specify the parm value as PARM=’MIGRATE dataCollectorID’.
     For example, if your Data Collector ID is DC01, specify PARM=’MIGRATE DC01’ for the parm value.
   - Complete the STEPLIB and run the job.
     The job output indicates whether the request completed successfully.

3. Replicate the DBC$STC started task for the DBC subsystem and customize the task for this LPAR.
   For information about the DBC$STC started task, see “DBC started task” on page 11.
Switching the Data Collector on a specific DB2 subsystem

Use this procedure to switch specific DB2 subsystems from using the version 6.x Data Collectors to the DBC as the $C jobs are run for that subsystem.

Before you begin

Review the following considerations:

■ The $C jobs are run only for the specific DB2 subsystem that you are migrating to the new release.

■ The 6.x Data Collectors and the DBC subsystems run simultaneously; they can monitor the same DB2 subsystems until all of the subsystems are switched and the 6.x Data Collector can be shut down. No conflicts exist with trace data set names between releases because the DBC uses linear data sets, which have a different low-level qualifier (LLQ). Both releases can use the same high-level qualifier (HLQ).

■ You cannot use the version 6.x Data Collector ID for version 10.x or later of the product. You will need to:
  — Edit the option set for the new collector ID and reference it in the output groups
  — Change the SSID in the DBC$STC started task proc
  — Change the DBC SSID referenced in the LGCDFLT step in the $G65INIT job

To switch the Data Collectors on a specific DB2 subsystem

1 Replicate the runtime data sets as instructed in “Replicating runtime data sets for different environments” on page 29, but make the following changes when editing the DOMPLEX option set and before running the DOM$STRT job:
   a Type FILTOFF on the command line if the filter is not already set to OFF.
   b Expand the DB2 Monitor List section.
   c Expand each DB2 subsystem listed.
   d If you do not want the DBC to start monitoring and collecting data for this DB2 subsystem, set the monitor switch for that type to N for that DB2 subsystem.

The following lists shows the monitoring types for which you can set values:

■ Monitor with MainView for DB2 - DC
Monitor with Pool Advisor / System Performance

Monitor with APPTUNE

e Press F3 to save your changes.

Subsequently, when DOM$STRT is running and the DBC is cycled, data collection will occur only for DB2 subsystems with the monitor switch set to Y.

2 Turn off monitoring by the 6.x Data Collector:

a Run the $C jobs for this DB2 subsystem.

b Invoke the product's 6.x DOMCLIST or SPDCLIST (or hyperlink from MainView).

c Select Administration => DOMPLEX Profiles.

d Choose M to modify the DOMPLEX profile.

e Select DB2 Monitor List.

f Select the DB2 subsystem that you want to switch to using the DBC for data collection.

g Change the monitoring settings for that DB2 subsystem to N.

h Press F3 to save the changes.

This change prevents automatic collection from starting if the 6.x Data Collector is bounced.

3 Enable data collection by the DBC:

a Invoke the product's 10.x DOMCLIST or SPDCLIST (or hyperlink from MainView).

b Select Administration => DOMPLEX Option Sets.

c Select the DOMPLEX option set.

d Expand the DB2 Monitor List.

e Expand the DB2 subsystem for which you want to turn on DBC data collection.

f Set the monitor switches to Y.
Press **F3** to save the changes.

The next time the DOM agent for DBC started task is cycled, data collection will automatically start up for this DB2 subsystem.

4 Issue the following console command for the DBC to start data collection for this DB2 subsystem:

```
/dbcssid APPON db2ssid
```

The variable `dbcssid` represents the DBC SSID for the new DBC subsystem, and `db2ssid` represents the DB2 subsystem ID. An example follows:

```
/DBC1 APPON DEBF
```

**Note**

You should have previously defined output groups that included this DB2 subsystem.

5 Issue the following command for the 6.x Data Collector to stop data collection for this DB2 subsystem:

```
/dcid APPOFF db2ssid
```

The variable `dcid` represents the ID of the 6.x Data Collector, and `db2ssid` represents the DB2 subsystem ID. An example follows:

```
/DC01 APPOFF DEBF
```

6 Repeat these steps for each DB2 subsystem that should switch to using the DBC for data collection.

### Replicating runtime data sets for different environments

Several methods exist for replicating runtime data sets for different environments. For example, you can use the Target Destination System (TDS) method that the Installation System offers, or you can manually replicate the data sets.

TDS can package the runtime data sets and re-create them with different names (for shared DASD deployment) or transport them via FTP to a non-shared DASD environment. TDS also allows for additional customization by packaging the installation data sets when it is necessary to transport to a non-shared DASD environment.
Replicating runtime data sets by using the Target Destination System (TDS) installation

User libraries can be configured in different ways with runtime data sets. For the performance products, a TDS installation is allowed on all of the options except **Concatenate the user libraries ahead of SMP/E data sets**.

For the performance products, this topic provides an overview of what values to specify for the runtime enablement (RTE) and TDS systems to accommodate different scenarios. For specific procedures, see the *Installation System User Guide*.

Figure 8 on page 30 shows the Runtime Enablement (RTE) Process Menu. This menu is part of the Installation System and includes the options that you will use to install the System and SQL Performance products.

**Figure 8: Runtime Enablement (RTE) Process Menu**

<table>
<thead>
<tr>
<th>Command</th>
<th>BMIP11C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Runtime Enablement (RTE) Process Menu</td>
</tr>
<tr>
<td>Select from the following options:</td>
<td></td>
</tr>
<tr>
<td>Runtime environments definitions</td>
<td></td>
</tr>
<tr>
<td>- Create initial runtime instance data sets</td>
<td></td>
</tr>
<tr>
<td>- Specify global infrastructure values</td>
<td></td>
</tr>
<tr>
<td>Initial runtime instance</td>
<td></td>
</tr>
<tr>
<td>- Specify product customization values</td>
<td></td>
</tr>
<tr>
<td>Target destination systems (TDS)</td>
<td></td>
</tr>
<tr>
<td>- Transport runtime data sets</td>
<td></td>
</tr>
<tr>
<td>- Specify product customization values</td>
<td></td>
</tr>
<tr>
<td>Additional customization options</td>
<td></td>
</tr>
<tr>
<td>- DB2 subsystem options</td>
<td></td>
</tr>
</tbody>
</table>

This topic provides procedures for the following menu options:

“1. Specify runtime system values” on page 31

“2. Create initial runtime instance data sets” on page 32

“3. Specify global infrastructure values” on page 33

“4. Specify product customization values” on page 34

“5. DB2 subsystem options” on page 37

“6. Transport runtime data sets” on page 37
1. Specify runtime system values

1. After selecting the runtime environment for your site, select **Specify runtime system values** on the Runtime Enablement (RTE) Process Menu.

2. On the Runtime Customization Instances panel (BMIPRT1), specify the RTE001 row for your first instance, and replicate that row for each TDS instance.

   **Note**
   If your values for the RTE001 row and the TDS rows are similar, BMC recommends completing the RTE001 row, replicating it for each TDS row, and adjusting the values in the TDS rows as needed.

   BMC recommends that you use SMS-managed data sets. As referenced for maintenance “Strategy 1” on page 14, BMC recommends using aliases for LOADLIB data sets. However, exercise caution when choosing the **Create an alias relationship to the runtime data sets** option in the Installation System; the alias names will be dropped (if they exist) and redefined into the new runtime data sets when the runtime data sets are created. Either specify new alias names in the Installation System, or manually manage switching the aliases when you are ready.

The TDS rows might represent different scenarios that deploy the runtime environment to different sysplexes or LPARs. For example, consider the environment shown in **Figure 9 on page 31**.

**Figure 9: Scenario for TDS install**

Assume that the following conditions apply to this scenario:

- The sandbox area has two LPARs (S1 and S2) in sysplex X1, and you want these two LPARs to share the same RTE data sets.

- In the development environment, sysplexes X1 and X2 share DASD. You want development LPARs D1 and D2 to use different RTE data sets than the sandbox so that you can control when maintenance gets applied.
- The test and production environments have two LPARs each (T1 and T2 for test, and P1 and P2 for production) in sysplex X3. Because X3 does not share DASD with the sandbox, the RTE data sets will need to be transported to X3. The LPARs in X3 share DASD with each other (that is, T1, T2, P1, and P2 share DASD), but you want to use different RTE data sets between test and production so that you can control when maintenance is applied. Data sets for test and production will need to be transported separately and given different HLQs to keep them distinct.

For this environment, the values for the runtime customization instances could be as follows:

**Table 1: Example of runtime customization values**

<table>
<thead>
<tr>
<th>TDS</th>
<th>Shared DASD a</th>
<th>Sysplex</th>
<th>LPAR</th>
<th>Runtime data set HLQ</th>
<th>Sysplex or system</th>
<th>TDS installation data set</th>
<th>Product data set b</th>
<th>VSAM data set</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTE1</td>
<td>Y</td>
<td>X1</td>
<td>S1</td>
<td>hlq.SAND</td>
<td>Sysplex</td>
<td>N/A</td>
<td>SAND.PRD</td>
<td>SAND.VSAM</td>
</tr>
<tr>
<td>TDS1</td>
<td>Y</td>
<td>X1</td>
<td>S2</td>
<td>hlq.SAND</td>
<td>Sysplex</td>
<td>N/A</td>
<td>SAND.PRD</td>
<td>SAND.VSAM</td>
</tr>
<tr>
<td>TDS2</td>
<td>Y</td>
<td>X2</td>
<td>D1</td>
<td>hlq.DEV</td>
<td>Sysplex</td>
<td>N/A</td>
<td>DEV.PRD</td>
<td>DEV.VSAM</td>
</tr>
<tr>
<td>TDS3</td>
<td>Y</td>
<td>X2</td>
<td>D2</td>
<td>hlq.DEV</td>
<td>Sysplex</td>
<td>N/A</td>
<td>DEV.PRD</td>
<td>DEV.VSAM</td>
</tr>
<tr>
<td>TDS4</td>
<td>N</td>
<td>X3</td>
<td>T1</td>
<td>hlq.TEST</td>
<td>System</td>
<td>TEST.install</td>
<td>TEST.PRD</td>
<td>TEST.VSAM</td>
</tr>
<tr>
<td>TDS5</td>
<td>N</td>
<td>X3</td>
<td>T2</td>
<td>hlq.TEST</td>
<td>System</td>
<td>TEST.install</td>
<td>TEST.PRD</td>
<td>TEST.VSAM</td>
</tr>
<tr>
<td>TDS6</td>
<td>N</td>
<td>X3</td>
<td>P1</td>
<td>hlq.PROD</td>
<td>System</td>
<td>PROD.install</td>
<td>PROD.PRD</td>
<td>PROD.VSAM</td>
</tr>
<tr>
<td>TDS7</td>
<td>N</td>
<td>X3</td>
<td>P2</td>
<td>hlq.PROD</td>
<td>System</td>
<td>PROD.install</td>
<td>PROD.PRD</td>
<td>PROD.VSAM</td>
</tr>
</tbody>
</table>

If the TDS system shares DASD with the RTE1 row, the shared DASD should be set to Y. This setting indicates to the Installation System that the runtime data sets need to be transported.

These data sets are used with MainView products and Runtime Component System (RTCS).

---

**Note**

Not all fields from the installation dialog are shown in this table.

Because TESTPLEX and PRODPLEX (which you want to keep separate) are on the same sysplex (X3), you need to specify system instead of sysplex to duplicate this example. Doing so allows you to configure the RTCS system registry to be shared between T1 and T1 but separated from P1 and P2. Table 2 on page 33 illustrates the desired configuration.

### 2. Create initial runtime instance data sets

After you finish specifying runtime system values, return to the Runtime Enablement (RTE) Process Menu and select Create initial runtime instance data sets. This option allows you to change low-level qualifiers (up to four distinct values) if you need to separate some FMID product data sets from others.
3. Specify global infrastructure values

After running the $R05RTEC job to create the initial runtime data sets, return to the Runtime Enablement (RTE) Process Menu and select **Specify global infrastructure values**. Global infrastructure components include:

- RTCS
- DBC
- NGL
- LGC

**Note**

If MainView is also installed, additional infrastructure components for the MainView coordinating address space (CAS) and product address space (PAS) will also be present.

For the environment in the example shown in this section, the global infrastructure data sets should be set up as shown in Table 2 on page 33.

### Table 2: Infrastructure components

<table>
<thead>
<tr>
<th>Component</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTCS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LGC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NGL</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The RTCS system registry and LGC private registry will be shared per set up for the sandbox, development, test, and production environments. In other words, for each separate set of data sets, the RTCS and LGC components will be shared. The DBC and NGL components will have an instance on each LPAR and, thus, will have unique repository data sets. However, the runtime data sets for those components will be shared per sandbox, development, test, or production environment.

**Table 3 on page 34** shows an example of names of components and members of the global infrastructure components. The example shows unique DBC SSIDs, DBC repositories, NGL registries, NGL IDs, and shared LGC repositories; the example also shows the LGC XCF groups used per each RTE and TDS row.
### Table 3: Example of global infrastructure values

<table>
<thead>
<tr>
<th>Share RTCS system registry</th>
<th>DBC SSID</th>
<th>DBC group</th>
<th>DBC repository</th>
<th>NGL registry</th>
<th>NGL ID</th>
<th>LGC repository</th>
<th>LGC XCF group</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTE1 Y</td>
<td>SND1</td>
<td>SANDPLEX</td>
<td>hlq.SND1.DBCREPOS</td>
<td>hlq.SND1.NGLRGRY</td>
<td>SND1</td>
<td>hlq.SND1.LGCXCF</td>
<td>LGCSAND</td>
</tr>
<tr>
<td>TDS1 Y</td>
<td>SND2</td>
<td>SANDPLEX</td>
<td>hlq.SND2.DBCREPOS</td>
<td>hlq.SND2.NGLRGRY</td>
<td>SND2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TDS2 Y</td>
<td>DEV1</td>
<td>DEVPLEX</td>
<td>hlq.DEV1.DBCREPOS</td>
<td>hlq.DEV1.NGLRGRY</td>
<td>DEV1</td>
<td>hlq.DEV1.LGCXCF</td>
<td>LGCDV</td>
</tr>
<tr>
<td>TDS3 Y</td>
<td>DEV2</td>
<td>DEVPLEX</td>
<td>hlq.DEV2.DBCREPOS</td>
<td>hlq.DEV2.NGLRGRY</td>
<td>DEV2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TDS4 Y</td>
<td>TST1</td>
<td>TESTPLEX</td>
<td>hlq.TST1.DBCREPOS</td>
<td>hlq.TST1.NGLRGRY</td>
<td>TST1</td>
<td>hlq.TST1.LGCXCF</td>
<td>LGCTST</td>
</tr>
<tr>
<td>TDS5 Y</td>
<td>TST2</td>
<td>TESTPLEX</td>
<td>hlq.TST2.DBCREPOS</td>
<td>hlq.TST2.NGLRGRY</td>
<td>TST2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TDS6 Y</td>
<td>PRD1</td>
<td>PRODPLEX</td>
<td>hlq.PRD1.DBCREPOS</td>
<td>hlq.PRD1.NGLRGRY</td>
<td>PRD1</td>
<td>hlq.PRD1.LGCXCF</td>
<td>LGCPRD</td>
</tr>
<tr>
<td>TDS7 Y</td>
<td>PRD2</td>
<td>PRODPLEX</td>
<td>hlq.PRD2.DBCREPOS</td>
<td>hlq.PRD2.NGLRGRY</td>
<td>PRD2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note**

- Not all fields from the installation dialog are shown in this table.
- Specifying **Y** or **N** in the **Share RTCS system registry** field has no effect if you are reusing an existing RTCS instance.
- The **NGL ID** is shown to match the **DBC SSID**, but they can be different. BMC recommends that you use the **NGL ID** as a node in the **NGL registry data set**. The **DBC group name** is the same per sandbox, development, test, and production environment. This organization allows communication between the **DBC subsystems on LPARS** in the sandbox, development, test, and production environment, but not across **DBC groups**.
- The **LGC registry and LGC XCF group** are the same per sandbox, development, test, and production environment. This organization allows sharing option sets in each of those environments, but not between environments. For example, option sets for production will not be shared with the test, development, or sandbox environments.

BMC recommends not using a version-specific name for registries and repositories so that they can persist and be easily reused from release to release. Enter the global infrastructure values for the RTE and optionally the TDS rows. You can return to this option to complete the TDS rows later, if needed.

### 4. Specify product customization values

1. After specifying the global infrastructure values, return to the Runtime Enablement (RTE) Process Menu and select **Specify product customization values**.
2 Define the process values for the initial runtime instances information:

   a  From the Process Menu for Customizing Initial Runtime Instances panel, select **Process values for initial runtime instances** under **Global infrastructure** to process values for the initial runtime instance.

   b  Select the RTE001 row.

   c  On the Global Infrastructure configuration option panel (ZMCP100), specify a data set to which the RTCS PROCs (OS*) will be copied.

   **WARNING**
   
   If you already have RTCS installed and running, do not specify a system PROC library on this panel to avoid inadvertently replacing the existing PROC.

   d  Generate and run the $G jobs for RTE001.

   Be sure to read the instructions in the $G00DOC member and in each job; some started tasks might need to be running before you execute the job.

   **Note**
   
   If you already have RTCS installed and running, you do not need to run the ALOCRTCS step from the $G65INIT job or follow the RTCS steps in the $G24SESEC and $G30UPRM jobs.

3 Customize the initial runtime instance:

   a  From the Process Menu for Customizing Initial Runtime Instance panel, select **Customize the initial runtime instance** option under **Product Configuration**.

   b  Select the RTE001 row, and specify the values for the initial DB2 subsystem.

   c  Generate and run the $C jobs.

4 Initialize the DBC agents:

   a  On the Process Menu for Customizing Initial Runtime Instances panel, select **Initialize DB2 Component Services (DBC) Agents and Option Set Templates** under **DB2 product configuration**.

   b  Select the RTE001 row, and specify the DOMPLEX option set for that environment:

   - If you are not migrating from an earlier version of the performance products, specify a DOMPLEX option set name to use. The DOMPLEX option set contains configuration values for the DB2 subsystem to monitor and output groups for each Data Collector.
If you are migrating from version 6.x, use the DOMPLEX profile name from the previous release.

The $C68DOM job will extract all of the DOMPLEX configuration information from the version 6.x STATUS data set. The job will then create an XML document that will be imported as a DOMPLEX option set into the LGC registry. You will need to edit this option set for new required values before starting the DOM agent. If you want to use a different name than the previous release, specify it on the BMC System and SQL Performance Options (DAAPU10) panel. Later, you can copy the migrated option set to the new name.

The DOMPLEX option set name should be shared per sandbox, development, test, and production LPARS.

<table>
<thead>
<tr>
<th>Row</th>
<th>Option set name</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTE1</td>
<td>SANDOPT</td>
</tr>
<tr>
<td>TDS1</td>
<td></td>
</tr>
<tr>
<td>TDS2</td>
<td>DEVOPT</td>
</tr>
<tr>
<td>TDS3</td>
<td></td>
</tr>
<tr>
<td>TDS4</td>
<td>TESTOPT</td>
</tr>
<tr>
<td>TDS5</td>
<td></td>
</tr>
<tr>
<td>TDS6</td>
<td>PRODOPT</td>
</tr>
<tr>
<td>TDS7</td>
<td></td>
</tr>
</tbody>
</table>

The DOMPLEX option set name can be the same as the DBC group name, but using the same name might prevent a version 10.1 DBC subsystem from communicating with a version 6.x Data Collector. Review “Communication between a DOM agent and an older release” on page 16 to determine whether you want to allow the communication during deployment.

c Generate and run the $U jobs for the RTE001 row.

5 Select Customize product options under DB2 product configuration on the Process Menu for Customizing Initial Runtime Instances (BMIP11R) panel, and complete the information.

Note
You must be on the LPAR where the DBC subsystem is running and the LGC agent is active within that DBC subsystem. For information about the option set, see “Editing the DOMPLEX option set” on page 16.
6 After customizing the product options, select **Start DBC Product Agents** under **DB2 product configuration** on the Process Menu for Customizing Initial Runtime Instances panel, and complete the information.

**Note**
For the performance products, the DOM$STRT job will start the DOM agent in the DBC subsystem, using the DOMPLEX option set name specified in the DOMPLEX= keyword parameter in this job. The DOMPLEX option set must exist in the LGC registry before you run this job.

7 Return to the Runtime Enablement (RTE) Process Menu.
After the DOM$STRT job runs, the performance products for the initial RTE001 system (sandbox) will be running, and installation for the first DB2 subsystem will be complete.

8 Verify that the product is working correctly:

- Check the DBC log for messages to ensure that the product agents are monitoring the correct set of DB2 subsystems, as defined in the associated DOMPLEX option set.

- Invoke the DOMCLIST or SPDCLIST (or hyperlink from MainView) and navigate to reports to verify that the product is working correctly.

**Note**
For the performance products, if other DB2 subsystems were specified in the DOMPLEX option set, the Data Collector might begin successfully monitoring them. However, any Explain command might fail until the $C jobs have run for that DB2 subsystem.

5. **DB2 subsystem options**

For RTE001, if additional DB2 subsystems should be monitored by the DBC subsystem that is associated with RTE001, select **DB2 subsystem options** under **Additional customization options** on the Runtime Enablement (RTE) Process Menu.

For more information, see “Replicating the installation to other DB2 subsystems” on page 44.

6. **Transport runtime data sets**

After the RTE001 runtime data sets are created and you have verified product installation, use the following procedure to transport the TDS instances.

1 On the Runtime Enablement (RTE) Process Menu, select **Transport runtime data sets** under **Target Destination systems (TDS).**
2 On the Generate TDS Transport JCL panel, select to generate transport JCL for all TDS instances. Depending on whether the TDS was defined to share DASD with the RTE instance, the transport might include a step to FTP the runtime data sets to the TDS system. The HLQs that you specified for the TDS rows will be used when the TDS runtime data sets are created.

3 Before executing the $TRA0010 job, consider generating other jobs ($G, $C, and $U) for the TDS environment first. Doing so is especially important when DASD is not shared between the RTE and TDS installations; after the $TRA0010 job runs, the JCL for the TDS will exist on another LPAR, and any further customization and generation of jobs will need to be done on each TDS. BMC recommends generating all jobs for TDS before executing the $T jobs. To generate the rest of the jobs needed for the TDS installation, use the Specify product customization information for TDS instance option from the Runtime Enablement (RTE) Process Menu.

   **Note**

   If you have already run the $TRA0010 and DASD is not shared between the RTE and TDS environment, you will need to invoke the TDSINSTL CLIST (instead of the BMCINSTL CLIST) on that TDS system. On the main menu, select Product Customization and then select Specify product customization information for the TDS instances.

When going through the installation panels for the TDS instances, consider changing the following items to values that are specific to this LPAR/DB2 subsystem:

- VSAM data sets from which to migrate the DOMPLEX configuration
- DB2 libraries
- APF data set name, if used
- DB2 subsystem and data sharing members

4 Review the comments in the $TRA0010 job before running it. If the FTP step was included, edit the job and specify logon information. If data sets were transported to an LPAR that did not share DASD with the original install, you will need to run the RECEIVE job.

5 Run the $TRA0010 job on the subsystem of origin where the RTE was configured.

6 If $TRA0010 transported data sets to another system, run the RECEIVE job for the TDS system.
Note
Locate the comment in the $TRA0010 job.

.SetFloat (S233.RJCL)

This job indicates the name of the RECEIVE job that will be created on the TDS LPAR.

7 On each TDS, execute the jobs on the appropriate LPAR, in the following order:

- $G

Note
If you already have RTCS installed and running, you do not need to run the ALOCRTCS step from the $G65INIT job or follow the RTCS steps in the $G24SESEC and $G30UPRM jobs.

- $C

- $U

- DOM$STRT

Read the comments in the DOC members and in each job; for some jobs, the started task must be running before you run the job.

Manually replicating your runtime data sets

BMC recommends using the TDS method provided in the Installation System to define, transport, and replicate runtime data sets because it includes specifying values for other LPARs in the customization steps. However, if you choose to transport or replicate the runtime data sets yourself, use the following procedure to edit the jobs manually.

In this procedure, you will use your own transportation method to:

- Copy the JCL data set and the runtime data sets from the original installation to the new sysplex

- Manually edit the JCL for each subsystem
Note

You do not need to run any of the $B jobs because these jobs created your SMP/E zones and target data sets.

You do not need to run the $R jobs if you are manually transporting the runtime data sets. The original installation uses the $R job only to copy members from the target data sets to the runtime data sets. To copy maintenance to another system, you will need to use your own method.

The DOMPLEX option set must exist in the LGC registry before running the DOM $STRT job.

You will need to edit all jobs to use the correct high-level qualifier (HLQ) for this environment.

1. Update and execute the $G05ALOC job.
   If you chose an option in the installation that concatenated user libraries ahead of runtime data sets, edit this job to change the HLQs (if necessary) before executing the job.

2. Update and execute the $G10VSAM job:
   a. Change the data set name in the ALOCLGC step, if necessary.
      This step allocates the LGC registry.
      
      Note
      Because this data set can be shared across LPARs, you might not need to run this step on every LPAR, depending on your configuration.

   b. For the DBC and NGL on each LPAR, change the ALOCNGL step to have the correct name for the DBC subsystem.
      This step allocates the NGL registry.
      For example, you might want to use the NGL PIID, LPAR name, or DBC ID in the node of the data set name to identify the NGL for this environment. If your site, by default, uses the SHAREOPTIONS based on data set names or SMS classes, ensure that the option is set to SHAREOPTIONS(1,3). The name of the NGL registry data set must match the data set that is specified in the REGISTRY DD in the DBC$STC PROC for the DBC subsystem on this LPAR.

   c. Change the data set name in the ALOCRTCS step, if necessary.
      This step allocates the RTCS system registry.
Note

Because this data set can be shared across LPARs, you might not need to run this step on every LPAR, depending on your configuration.
If RTCS is already running on this LPAR, skip this step and the RTCS instructions in $G24ESEC and $G30UPRM.

3 Review the $G45COPY job and make changes, if needed.
This job copies members from the JCL data set to the runtime data sets. Either edit this job to reference the correct data sets, or manually copy the members.
The DBC will use the following members on this LPAR:

- $DBCssid, where ssid represents the DBC subsystem ID
  Rename the member to specify the DBC SSID for this LPAR. Edit this member to change the DBC repository name that will be allocated by the DBC at start up for this LPAR. This member name should be referenced on the DBCPARMS DD in the DBC$STC PROC.

- $SECssid, where ssid represents the DBC subsystem ID.
  This member typically does not contain any LPAR-specific information, so it can have a generic name if you want to share it across all DBC subsystems. This member name should be referenced in the DBCSECUR DD statement in the DBC$STC PROC.

4 Ensure that the DBC$STC started task is running before running the rest of the jobs.
For more information about the DBC started task, see “DBC started task” on page 11.

5 Update and execute the $G65INIT job:

a Edit the values used within the following elements to match your naming conventions (if necessary):

- <GROUP>
  This element should specify the DBC group name.

- <PIID>
  This element contains the NGL ID for the NGL agent on this LPAR. This element might appear more than once. The NGL PIID is also specified in the NGLID= keyword parameter.

WARNING

Do not change the PIID for the NGR agent. The NGR agent must use NREG.
b In the LGCDFLT step, change the values for the following elements:

- dbcssid =
  This value contains the DBC SSID that is used for the LGC on this LPAR.

- dbcgroup =
  This value contains the DBC group name and should match the <GROUP> element.

- default=
  This value indicates whether the specified DBC group should be registered as the default DBC. For more information, see “DBC default groups” on page 12 and “Replicating DBCs on shared DASD in the same sysplex” on page 46.

- rtsreg xcfgroup
  This value indicates the XCF group that the LGC agent will use. See “Cross-system coupling facility (XCF) groups” on page 9.

- regdsn
  This value indicates the LGC registry data set that the LGC agent will use on this LPAR.
  To determine whether other DBC groups will share this registry, see the figures in “Deployment overview” on page 2 and “Replicating runtime data sets by using the Target Destination System (TDS) installation” on page 30.

c In the REGLGC step, change the name of the XML data sets in the <dsn> elements, if needed.

6 Run the $G jobs.
Read the comments in the $G00DOC member and in each job; for some jobs, the started task must be running before you run the job.

7 Update the $C jobs:

   **Note**
   For the $C05ALOC job, if you chose an option in the installation that concatenated user libraries ahead of runtime data sets, edit this job to change the HLQ (if necessary) and run the job.

   a For each subsystem, edit all $C jobs to have the correct DB2 subsystem and DB2 libraries.
b For each DOMPLEX or DBC group, edit the $C10VSAM job and change the names of the VSAM data sets in the DOM* steps (if necessary) to create new VSAM data sets for this environment.

If you are migrating from a previous release, the $C68DOM job can then migrate the data to these new data sets.

c For each subsystem, before running the $C45CNTL job, create the correct PSS2ssid members in your JCL data set (where ssid is the DB2 subsystem ID).

Ensure that you use the correct subsystem and DB2 libraries. For example, you would use PSS2DB2A for DB2 subsystem DB2A. You should have a PSS2ssid member for each DB2 subsystem or member of a data sharing group, and a member for the group attach name.

d For each DOMPLEX or DBC group, edit the $C68DOM job to reference the older release data sets pertinent to this environment.

Doing so ensures that you are migrating the data for this environment and converting it to 10.1 data.

e After you finish updating the $C jobs, run the jobs.

8 Update and run the $U20INIT job.

You might need to edit the values used in the following elements, depending on your naming convention. For more information about the $U20INIT job, see “$G65INIT and $U20INIT usage notes” on page 13.

- <GROUP>
  This element contains the DBC group name.

- <LOADLIB>
  Review all LOADLIBs referenced in this job.

- <dsn>
  This element indicates the product XML data sets (also referenced in the IODFILTI step if you are installing APPTUNE).

- <file id = "load">
  The value indicates the load library that APPTUNE uses.

9 If necessary, create a DOMPLEX option set as instructed in “Editing the DOMPLEX option set” on page 16.

--- Note

After running the $U20INIT job, if you migrated from a 6.x release, your DOMPLEX option set will have been imported into the LGC registry. In that case, you can skip this step.
10 After saving any changes that you made to the DOMPLEX option set, update and run the DOM$STRT job.

You might need to edit the values used in the following elements, depending on your naming conventions:

- `<GROUP>`
  This value represents the DBC group name.

- `DOMPLEX =`
  This value represents the name of the DOMPLEX option set that you specified.

- `NID`
  This value represents the NGL PIID for the NGL agent that is specified in the NGLINIT step from the $G65INIT job.

After you run the DOM$STRT job, the performance products for this LPAR will be running, and installation will be complete.

**Note**

For the performance products, the DOM$STRT job will start the DOM agent in the DBC subsystem, using the DOMPLEX option set name specified in the DOMPLEX= keyword parameter in this job.

11 To perform the installation for other DB2 subsystems to be monitored by this DBC, use the instructions in “Replicating the installation to other DB2 subsystems” on page 44.

**Replicating the installation to other DB2 subsystems**

The procedures in this section offer two methods for replicating DB2 objects to other DB2 subsystems. You can use the Multiple Subsystem Install (MSSID) method through the Installation System, or you can manually replicate the original JCL.
The performance products for DB2 use product-specific DB2 objects for some features. The same product code base can be used with many DB2 subsystems at differing DB2 versions and modes (DB2 V8, V9, or V10). The procedures in this section are based on the following assumptions:

- Only the DB2 objects need to be created.
- The product suite of data sets from the original installation will be used to run the relevant jobs.

Use the procedure that best suits your situation:

- The Installation System's MSSID method replicates the DB2 objects to other DB2 subsystems with just a few changes to the original DB2 subsystem values. This method generates a JCL data set for each subsequent subsystem.
- If you choose to replicate the original JCL data set manually, you will have to copy the JCL data set and then edit the $C jobs for each subsystem.

**To replicate DB2 objects by using MSSID**

1. Invoke the install CLIST and select Product Customization.
2. Select whichever option you performed for runtime data sets, and then select DB2 subsystem options under Additional Customization Options.
3. On the Additional Customization Options menu, select Customize for multiple similar DB2 subsystems (MSSID).
4. Select the RTE or TDS instance to process.
5. Complete the MSSID options for the DB2 subsystems for which you want to replicate the installation.

**Note**

Use the same suite of product data sets, even if that suite is on a different LPAR. You can change the job card for each subsystem, if necessary.

When you finish specifying options, the Installation System generates the $S00JCL job.

6. Run the generated $S00JCL job.

This job replicates a JCL data set for each DB2 subsystem. Each data set contains jobs to run for that subsystem, customized with the correct DB2 libraries.
7 Run the $C jobs for each DB2 subsystem.

To manually replicate the original JCL

1 Copy the original JCL data set.

2 Edit all $C jobs to have the correct DB2 subsystem and libraries.

3 Before running $C45CNTL, create the correct PSS2 ssid members in your JCL data set (where ssid is the DB2 subsystem ID), with the correct subsystem and DB2 libraries in it.

   For example, PSS2DB2A would be used for DB2 subsystem DB2A. You should have a PSS2 ssid member for each DB2 subsystem or member of a data sharing group, and a member for the group attach name.

4 Run all of the $C jobs except:
   - $C05ALOC
   - $C10VSAM
   - $C15PSWD
   - $C20APF
   - $C68DOM

Replicating DBCs on shared DASD in the same sysplex

The procedures in this section offer two methods for replicating the DBC in a shared-DASD environment in the same sysplex as the original installation. You can replicate the DBC in the same DBC group as the original installation, or in a different DBC group.

Replicating the DBC (using the same DBC group and DOMPLEX option set)

Use this procedure if you have one DBC up in the DOMPLEX, and you want to deploy the installation to another DBC that will join the same DBC group and use the same DOMPLEX option set.
1 Replicate the installation to other DB2 subsystems.

See “Replicating the installation to other DB2 subsystems” on page 44.

2 Run the ALOCNGL step of $G10VSAM on each LPAR.

Because the resulting NGL registry is used by each DBC started task on each LPAR, the registry name needs to be unique. For example, you can use the LPAR name or DBC ID in the data set name to identify the data set to this LPAR.

**Note**

You do not need to run the ALOCLGC step in $G10VSAM because the LGC from the first LPAR will be shared. Similarly, you do not need to run the ALOCRTCS step because the RTCS system registry from the first installation will be shared.

3 To determine what SSID you will use, review “Switching to the DBC from the version 6.x Data Collector” on page 25.

4 Bring up the DBC$STC started task on this LPAR.

For information about the started task, see “DBC started task” on page 11.

5 Run all of the steps in the $G65INIT job except REGLGC, and ensure that the job runs on the correct LPAR.

For the LGCDFLT step, all of the information in the `<register>` element should be the same as in the original installation with one exception: change the DBC ssid value to the new one for this subsequent LPAR.

```xml
<register>
  <dbc ssid="DC01" dbcgroup="DCPLEX" default="yes">
    <rtcsreg xcfgroup="DCPLEX1">
      <regdsn>BMCPERF.TEST.LGCRGRY</regdsn>
    </rtcsreg>
  </dbc>
</register>

<register>
  <dbc ssid="DC02" dbcgroup="DCPLEX" default="yes">
    <rtcsreg xcfgroup="DCPLEX1">
      <regdsn>BMCPERF.TEST.LGCRGRY</regdsn>
    </rtcsreg>
  </dbc>
</register>
```

**Note**

In the `<register>` step that you run, *only* the DBC SSID should be different from what you ran for the original installation for this DBC group. The DBC group name, the default, the XCF group name, and the `<regdsn>` names should all be the same as they were in the original installation. This approach enables both DBC subsystems to share the LGC private registry and option sets.
6 Modify and run only the DOMINIT and DOMDEFN steps in the $U20INIT job to define the DOM agent.

   Review the DOMUTIL1 member referenced on the first data set in the DOMDEFN step. This member should be customized for this LPAR: the <PIID> must match the <PIID> and NGLID= element and statement.

   For more information, see “$G65INIT and $U20INIT usage notes” on page 13. You do not need to execute any of the steps that call PGM=LGCUTIL from the $U20INIT job.

7 Ensure that you have the second DBC subsystem defined to the DOMPLEX option set:

   a Invoke the DOMCLIST or SPDLIST (or hyperlink from MainView) and select Administration => DOMPLEX Option sets.

   b Select the option set that the first DBC subsystem is using.

   c Expand the Data Collector List section and ensure that the DBC ID for this LPAR is defined as a data collector.

   d (Pool Advisor only) Expand the section for the DBC ID for this LPAR, and confirm that the following fields reference data sets:

      ■ Data Collectors advisor variable repository

      ■ Pool Advisor history repository

   Note

   Ensure that these data sets exist before continuing. Use sample members PMDHIST and PMDJINST in SAMP data set (BBSAMP or BMCSAMP) to create the data sets, if needed.

   e Expand the DB2 Monitor List section and ensure that your DB2 subsystems for this LPAR are defined.

      You might also want to expand the related sections on the panel and review their settings.

   f Expand the Output Group section and ensure that you have output groups defined and log set definitions for this DBC ID.

8 In the DOM$STRT JCL, edit the <PARMS> value to ensure that:

      ■ The NID is the same NGLID as used in $G65INIT for this LPAR
The DOMPLEX name is the option set name that the original installation used

9 Run the job (ensuring that it runs on the correct LPAR), and check for new messages in the DBC log.

You can check the DBC log messages to ensure that the product agents are monitoring the correct set of DB2 subsystems, as defined in the associated DOMPLEX option set. Also, you can invoke the DOMCLIST or SPDCLIST (or hyperlink from MainView), and navigate to reports to verify that the product is working correctly.

**Replicating the DBC (using a different DBC group)**

Use this procedure if you need to change the setup of the DBC to accommodate more than one DBC group in the same sysplex or on the same LPAR.

Using this procedure is recommended in the following situations:

- Multiple DBC subsystems are running on the same LPAR but are in different DBC groups, like in the following example:

```
<table>
<thead>
<tr>
<th>LPAR 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBC TESTPLEX</td>
</tr>
<tr>
<td>DBC PRODPLEX</td>
</tr>
</tbody>
</table>
```

- Multiple DBC subsystems are running in the same sysplex, they are in different DBC groups, *and* the RTCS system registry is shared:
For example, assume that you have DB2 subsystems that represent the test and production environments on the same LPAR (or in the same sysplex where the RTCS system registry is shared). You should separate the DBC subsystems that are monitoring those DB2 subsystems so that you can roll maintenance into production in a controlled manner. The test and production DBC subsystems will use separate runtime data sets, option sets, and a separate LGC private registry datastore. Because the installation is set up to use the RTCS system registry to find the LGC private registry datastore, you need to use special DD statements in the installation jobs to direct TEST or PROD to its own datastore.

**Overview of using ZBCCssid to designate non-default DBC groups**

The DOM agent (that is, the Data Collector) looks for the DOMPLEX option set for its DBC subsystem by finding the LGC private registry associated with the DBC group that was registered as `default=yes`. Only one DBC group can be registered as the default per RTCS system registry. In the situations described earlier in this topic, the DOM agent might not find the correct DOMPLEX option set; in that case, the DOM agent receives an LGC-CONNECT or LGC-GETOPTION error.

You can avoid this issue if the DBC groups that are *not* designated as the default group use a special ZDBCssid designation in:

- The DBC$STC started task procedure
- The product CLISTs (DOMCLIST, SPDCLIST, or DOMC)

You can designate only one DBC group as `default=yes` per RTCS system registry. In other words, each RTCS system registry can have only one default DBC group.
designated (meaning a one-to-one correlation). All other DBC groups will be registered as `default=no` and need to use the ZDBCssid designation.

- If you have more than one DBC subsystem running on the same LPAR and they are in different DBC groups, one group must be designated `default=yes`; the others must be `default=no` and must use the ZDBCssid designation.

- If you have more than one DBC up in the same sysplex, they are in different DBC groups, and your RTCS system registry is shared across these LPARs, one group will be designated `default=yes`; the others will be `default=no` and must use the ZDBCssid designation.

**Replicating the DBC (for a different DBC group) by using a ZDBCssid designation**

This procedure can help you determine when to assign a ZDBCssid designation for the non-default DBC group. For example, you could have your test environment use the ZDBCssid designation and your production environment use the `default=yes` designation. Alternatively, you could designate ZDBCssid on all DBC subsystems if you wanted to keep them similar. You must also allocate ZDBCssid when invoking the product CLIST (DOMCLIST, SPDCLIST, or DOMC) in order to access the correct LGC private registry datastore for any non-default DBC subsystem.

1. Review “Replicating runtime data sets for different environments” on page 29 and follow the instructions for the method that you select.

   Whether you used TDS processing or a manual process for a subsequent LPAR, review the steps in this section to see if any changes need to be made to the jobs before you run them.

   **Note**

   This section documents exceptions to the normal process. To set up the ZDBCssid designations, follow your normal method for replicating the runtime data sets, but observe the exceptions noted in this procedure when running the jobs.

2. Review the “RTCS system registry” on page 8 and “DBC default groups” on page 12 sections.

   Assume that you have the following environment, and that you want to set up TEST to specify ZDBCssid and let PROD use the default DBC group:

<table>
<thead>
<tr>
<th>Environment</th>
<th>DBC SSID</th>
<th>DBC GROUP</th>
<th>Default</th>
<th>XCF Group</th>
<th>REGDSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEST</td>
<td>TST1</td>
<td>TESTPLEX</td>
<td>No</td>
<td>TSTPLEX1</td>
<td>BMCPERF.TEST.LGCRGRY</td>
</tr>
<tr>
<td>PROD</td>
<td>PRD1</td>
<td>PRODPLEX</td>
<td>Yes</td>
<td>PRDPLEX1</td>
<td>BMCPERF.PROD.LGCRGRY</td>
</tr>
</tbody>
</table>
For PROD, the LGCDFLT step from $G65INIT will look as follows:

```xml
<register>
  <dbc ssid="PRD1" dbcgroup="PRODPLEX" default="yes">
    <rtcsreg xcfgroup="PRDPLEX1">
      <regdsn>BMCPerfil.PROD.LGCRGRY</regdsn>
    </rtcsreg>
  </dbc>
</register>
```

For TEST, the LGCDFLT will appear as follows:

```xml
<register>
  <dbc ssid="TST1" dbcgroup="TESTPLEX" default="no">
    <rtcsreg xcfgroup="TSTPLEX1">
      <regdsn>BMCPerfil.TEST.LGCRGRY</regdsn>
    </rtcsreg>
  </dbc>
</register>
```

Make sure that the $G65iNIT job runs on the correct LPAR.

3 Edit DBC$STC (the customized DBC started task in your JCL data set) to add the following DD statement to the PROC, where ssid is your DBC subsystem ID:

```bash
//ZDBCssid DD DUMMY
```

For example, in the TST1 installation on TEST, you would use ZDBCTST1, where TST1 is the DBC SSID. On other LPARS in this group, specify their DBC ID for ssid.

**Note**

This step is required for any DBC that was registered with a DBC GROUP that is default=no.

4 Perform a FIND command on LGCUTIL in the $G65INIT and $U20INIT jobs, and (for each step calling that program) add the following DD statement to the step, where ssid is your DBC subsystem ID:

```bash
//ZDBCssid DD DUMMY
```

5 Determine which method you prefer for editing the DOMPLEX option set for the first time (as explained in “Editing the DOMPLEX option set” on page 16), and use that method to edit the option set, as follows:

- **If you prefer to use the Installation System dialogs**, complete these steps:

  1 Run BMCINSTL or TDSINST.

  2 Select **Product Customization**, and then select whichever option you performed for the runtime data sets.

  3 Select **Specify product authorization values** under the **Initial runtime instances** section.
4 *(required for any DBC that was registered with a DBC GROUP that is default=no)* Type the following command (where ssid is your DBC subsystem ID) on the command line in the installation dialog:

```
TSO ALLOC FI(ZDBCssid) DUMMY REU
```

5 From the Runtime Enablement (RTE) Process Menu, select **Customize product options** under **DB2 Product Configuration**, and then select the appropriate TDS row to update your DOMPLEX option set, using the correct LGC private registry for your installation.

- *If you prefer to use the DOMCLIST or SPDCLIST job in the installation JCL OR CLIB data set*, complete these steps:

1 Edit the DOMCLIST or SPDCLIST to change the top of the PROC statement to automatically connect to your DBC subsystem:

   Having separate CLISTs for the test and production environments forces TEST to connect with TEST and PROD to connect with PROD. Use the following statement, where domplex is your DOMPLEX name:

   ```
   PROC 0 P('PRD=,DP=domplex') SSID()
   ```

   For example, you could use:

   ```
   PROC 0 P('PRD=,DP=TESTPLEX') SSID().
   ```

   **Note**

   The DOMPLEX name should match the DOMPLEX = keyword value in the DOM$STRRT started task for TEST.

2 *(required for all DBC subsystems that were registered with a DBC GROUP that is default=no)* Add the following allocation to your product CLISTs (DOMCLIST, SPDCLIST, or DOMC) to connect to your LGC component:

   ```
   ALLOC F(ZDBCssid) DUMMY REUSE
   ```

   And a FREE statement, where ssid is your DBC SSID:

   ```
   FREE F(ZDBCssid)
   ```

   For example, TEST would use the following statements:

   ```
   ALLOC DD(ZDBCTST1) DUMMY REUSE
   FREE F(ZDBCTST1)
   ```

3 If you want to use the same CLIST for multiple LPARS, add code similar to the following example, and then use similar IF statements at the end of the CLIST to free the allocation:

   ```
   IF &SYSNAME=DB2A THEN +
   DO
   ALLOC F(ZBCLGA2)DUMMY REU
   END
   IF &SYSNAME=DB2B THEN +
   DO
   ALLOC F(ZBCLGA3)DUMMY REU
   END
   ```
4 Execute the DOMCLIST or SPDCLIST (or hyperlink from MainView).

5 Navigate to Administration and then DOMPLEX Option Sets to edit and customize your DOMPLEX option set.

- If you prefer to export the option set, use TSO Edit or a REXX exec to make changes, and then import the option set, as follows:

1. Review “Editing the DOMPLEX option set in batch mode” on page 22.

2. On the Export and Import JCL, add the following DD statement to the job, where ssid is your DBC ID:

```
//ZDBCssid DD DUMMY
```

**Avoiding ZDBCssid**

If necessary, you could avoid needing the ZDBCssid designation by using a different configuration. As stated in the previous section, use of the ZDBCssid is indicated in the following situations:

- **Multiple DBC subsystems are running on the same LPAR that are in different DBC groups.**

  To avoid use of ZDBCssid in this situation, use either of the following options:

  — Require the DBC subsystem registered as default=yes to come up first.

  — Set up a dedicated DBC group to be registered as default=yes, include only the LGC agent in it, and require it to come up before the DBC subsystems.

- **Multiple DBC subsystems are running in the same sysplex, they are in different DBC groups, and the RTCS system registry is shared.**

  To avoid use of ZDBCssid in this situation, you could set up the RTCS system registry to be shared per DBC group.

  For more information, contact BMC Support.

---

**Replicating the installation to another sysplex on non-shared DASD**

Use this procedure to replicate the installation to another sysplex on non-shared DASD.
1. Determine which method you prefer for transporting the runtime data sets to the non-shared DASD LPAR (as explained in “Replicating runtime data sets for different environments” on page 29), and use that method to run the necessary jobs.

2. Replicate the installation to other DB2 subsystems that you want to monitor using this DBC.

   For more information, see “Replicating the installation to other DB2 subsystems” on page 44.

3. Replicate this DBC subsystem to other LPARs in the same group.

   For more information, see “Replicating the DBC (using the same DBC group and DOMPLEX option set)” on page 46.

Where to get the latest product information

To view the latest BMC documents, see the Support Central website at http://www.bmc.com/support.

Notices such as flashes, technical bulletins, and release notes are available on the website. You can subscribe to proactive alerts to receive e-mail messages when notices are issued or updated. For more information about proactive alerts, see the Support Central website.

© Copyright 2012 BMC Software, Inc.

BMC, BMC Software, and the BMC Software logo are the exclusive properties of BMC Software, Inc., are registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

DB2 is a trademark or registered trademark of International Business Machines Corporation in the United States, other countries, or both.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

BMC SOFTWARE INC
2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA
Telephone 1 713 918 8800 (or 1 800 841 2031 United States and Canada)

Customer Support: 1 800 537 1813 (United States and Canada) or contact your local support center