

13.1

*IBM Db2 Object Comparison Tool for  
z/OS  
User's Guide*



**2025-06-16 edition**

This edition applies to IBM® Db2® Object Comparison Tool for z/OS® 13.1 (product number 5655-CH1) and to any subsequent releases until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Specific changes are indicated by a vertical bar to the left of a change. A vertical bar to the left of a figure caption indicates that the figure has changed. Editorial changes that have no technical significance are not noted.

© **Copyright International Business Machines Corporation 2001, 2025.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>About this information.....</b>	<b>vii</b>
<b>Chapter 1. Overview.....</b>	<b>1</b>
What's New in Db2 Object Comparison Tool 13.1.....	2
GA enhancements.....	2
New-function APARs.....	3
The comparison process.....	15
Components of the comparison process.....	17
Terminology.....	18
Product documentation and updates.....	20
Accessibility features.....	21
<b>Chapter 2. Customization.....</b>	<b>23</b>
Customization checklist for Db2 Object Comparison Tool.....	23
Software requirements for Object Comparison Tool.....	24
Data sets used by Tools Customizer.....	25
Customizing Db2 Object Comparison Tool.....	25
Roadmap: Customizing Db2 Object Comparison Tool for the first time.....	25
Roadmap: Recustomizing Db2 Object Comparison Tool.....	28
Using Tools Customizer in a multiple-LPAR environment.....	29
Editing the GOCFB2VB job.....	30
Submitting the customization jobs.....	32
Allocating libraries for Db2 Object Comparison Tool.....	34
Customizing the JCL that Object Comparison Tool uses.....	34
Customizing data set names.....	35
Enabling product discovery.....	37
ADBL CLIST for invoking Db2 Object Comparison Tool.....	38
<b>Chapter 3. Getting started with Db2 Object Comparison Tool.....</b>	<b>39</b>
Opening Object Comparison Tool.....	39
Object Comparison Tool main menu.....	40
Db2 Object Comparison Tool scenarios.....	41
Scenario: Comparing a Db2 development catalog to a Db2 production catalog.....	41
Scenario: Undoing changes that were made in a catalog-to-catalog comparison.....	46
Scenario: Comparing DDL to a catalog.....	48
Scenario: Copying objects.....	50
Scenario: Converting partitioned table spaces to partition-by-range universal table spaces.....	53
<b>Chapter 4. Comparing Db2 objects.....</b>	<b>55</b>
1. Specifying source objects.....	56
Specifying a DDL file for the source or target definition.....	57
Specifying the Db2 catalog for the source or target definition.....	59
Specifying a SELECT statement for the source or target definition.....	64
Specifying a version file for the source or target definition.....	66
Specifying a version scope for the source or target definition.....	67
2. Specifying target objects.....	68
Specifying that the target definition is automatically selected based on the source.....	70
3. Specifying compare masks.....	71
Translation masks.....	73
Mask data set.....	76

4. Specifying ignores.....	80
Specifying ignore fields.....	81
Specifying ignore changes.....	83
5. Generating a compare job.....	84
Compare job options.....	87
Compare reporting options.....	110
Generating a compare batch job to make changes by using Change Management.....	111
Generating a compare batch job for a multi-target import.....	114
Parameters in the generated compare batch job.....	117
<b>Chapter 5. Checking the compare report.....</b>	<b>123</b>
Compare report format .....	124
Sample compare report 1.....	128
Sample compare report 2.....	130
Sample compare report 3.....	132
Sample compare report 4.....	135
Sample compare report 5.....	138
Sample summary conversion report.....	138
<b>Chapter 6. Excluding objects from the compare process.....</b>	<b>141</b>
Creating exclude specifications from saved compare results.....	142
Creating or editing exclude specifications during a comparison.....	144
<b>Chapter 7. Managing saved compare results.....</b>	<b>147</b>
Compare Results (ADBPCRS) panel.....	149
<b>Chapter 8. Ignoring changes.....</b>	<b>151</b>
Creating ignore changes specifications.....	151
Modifying ignore changes specifications.....	152
<b>Chapter 9. Applying changes to target objects.....</b>	<b>155</b>
Apply jobs.....	155
Running a work statement list to apply changes.....	157
<b>Chapter 10. Managing and restoring dialogs.....</b>	<b>159</b>
Saving dialogs.....	160
<b>Chapter 11. Converting version files to the latest Db2 version.....</b>	<b>163</b>
<b>Chapter 12. Comparing multiple sources and targets.....</b>	<b>165</b>
Manage Dialogs MultiCompare (GOCMC) panel.....	167
<b>Chapter 13. Batch DDL file extraction program .....</b>	<b>171</b>
Supported SQL statements for DDL file extraction.....	171
Batch DDL file extraction program report format .....	174
<b>Chapter 14. Batch Db2 catalog extraction program .....</b>	<b>175</b>
Batch Db2 catalog extraction program report.....	175
<b>Chapter 15. Batch compare program .....</b>	<b>179</b>
Compare version files .....	179
Special considerations for comparing Db2objects.....	180
Comparisons with LOBs.....	186
<b>Chapter 16. Running Compare by using a Change Management batch job.....</b>	<b>187</b>

Comparing table pairs.....	191
<b>Chapter 17. Specifying alternate input to the generate apply job program.....</b>	<b>195</b>
Alternate values for the generate apply program.....	196
Creating user-defined templates.....	212
<b>Chapter 18. Recommendations when comparing a large number of objects.....</b>	<b>217</b>
<b>Chapter 19. Troubleshooting and messages.....</b>	<b>219</b>
Db2 Object Comparison Tool condition codes.....	219
Troubleshooting: The Compare report shows changes to bind options for trigger packages.....	220
Performance considerations.....	221
Recommendations when comparing a large number of objects.....	221
<b>Notices.....</b>	<b>223</b>
<b>Index.....</b>	<b>227</b>



## About this information

---

This information describes how to use IBM Db2 Object Comparison Tool for z/OS.

These topics help database administrators, system programmers, and application programmers perform these tasks:

- Customize your Db2 Object Comparison Tool environment.
- Compare sets of IBM Db2 objects by using Db2 Object Comparison Tool
- Generate reports and jobs by using Db2 Object Comparison Tool

Always check IBM Documentation (IBM Docs) for the most current version of this publication:

<https://www.ibm.com/docs/en>



---

# Chapter 1. Overview of Db2 Object Comparison Tool

IBM Db2 Object Comparison Tool for z/OS (also referred to as Object Comparison Tool) compares the definitions of existing Db2 for z/OS objects from different sources and reports the differences. Object Comparison Tool can subsequently synchronize these sources by making the relevant changes to the objects. Additionally, Object Comparison Tool is a required prerequisite for using the Change Management function of IBM Db2 Administration Tool for z/OS (Db2 Admin Tool).

**Important:** Db2 Object Comparison Tool 13.1 (5655-CH1) is available only as part of IBM Db2 Change Management Solution Pack for z/OS 1.2 (5655-CH1) and IBM Db2 Administration Solution Pack for z/OS 3.3 (5697-ASP). Object Comparison Tool is no longer available as a stand-alone product.

Specifically, Db2 Object Comparison Tool can help you with the following goals:

- Keep your production system a mirror image of your test and development systems.

New applications, changes to existing applications, or mistakes can cause Db2 objects in one system to have different attributes from objects in other systems. Object Comparison Tool can find differences between objects (and dependent objects) in a Db2 catalog on one system and a Db2 catalog on a different system. Object Comparison Tool can then generate batch jobs to synchronize the catalogs.

- Compare objects with different names.

Often, production objects and test objects use different naming conventions. You can account for these naming differences by using a feature in Object Comparison Tool called *masks*. With masks, object names can be translated before a comparison. Therefore, a test object can be matched to the corresponding production object for comparison. For example, if you want to compare all tables that begin with TEST to all tables that begin with PROD, you can define a mask that tells Object Comparison Tool to translate table names TEST\* to PROD\* for the comparison. (In this example, the asterisk is a wildcard character.)

- Ignore specific properties when comparing objects.

You might not want your test objects to be exactly the same as your production objects. Object Comparison Tool can handle these intentional differences when comparing objects. To specify that the tool ignore certain attributes, such as the number of partitions in a table space or the storage group for a database, use the *ignore fields* feature.

- Produce reports about the object comparison.

Depending on the reporting options, Object Comparison Tool produces a variety of reports to show the differences between the objects.

- Apply any changes to the target objects.

Object Comparison Tool can generate jobs that apply any requested changes to the target objects. To request such jobs, use the *generate apply jobs* function. Alternatively, you can request that these changes be generated to a work statement list (WSL) that you can subsequently use to apply changes to the target objects. WSLs make it easy to propagate changes to remote sites.

- Track changes.

Changes can be imported into the Change Management Database to help you manage the process of recording and tracking the changes that you make to your objects.

- Undo implemented changes

If you made changes and need to revert to the original state of the objects, Object Comparison Tool can revert those changes for you. The *undo capability* of the tool can restore application objects to a previous version.

## Related information

[Video - IBM Db2 Object Comparison Tool for z/OS: An overview](#)  
[IBM Db2 Administration Tool for z/OS](#)

## What's new in Db2 Object Comparison Tool 13.1

IBM Db2 Object Comparison Tool for z/OS (Object Comparison Tool) 13.1 introduces new features to support Db2 13 for z/OS and other usability features. Some of these enhancements were delivered on the General Availability (GA) date. Other enhancements were delivered later in the service stream, as part of new-function APARs.

### Db2 13 function level support:

For information about any program temporary fixes (PTFs) that are required to support Db2 13 function levels, see [Db2 13 function level support \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

## GA enhancements in Db2 Object Comparison Tool 13.1

The following enhancements are available as of the General Availability (GA) date of Object Comparison Tool 13.1.

### Online conversion of partition-by-growth (PBG) table spaces to partition-by-range (PBR) table spaces

Db2 13 introduces the capability to convert a table with growth-based partitions (in a PBG table space) to use range-based partitions (in a PBR table space) with an online change that has minimal impact to your applications. This online conversion is accomplished by using an ALTER TABLE statement with the new ALTER PARTITIONING TO PARTITION BY RANGE clause.

You can use Db2 Object Comparison Tool 13.1 to perform this online conversion. When APPLCOMPAT is set to V13R1M500 or higher and a target PBG table space needs to be changed to a PBR table space, Object Comparison Tool generates an ALTER statement when valid (according to any Db2 restrictions) and any necessary REORG statements to perform this conversion and thus minimize outages.

#### Related information:

[Overview of what's new in Db2 13 \(Db2 13 for z/OS documentation\)](#)

[What's new in Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

### Support for package owner type

To increase flexibility for package ownership, Db2 13 allows you to specify whether the owner of a package is a role or authorization ID with the following new syntax:

- For the Db2 commands BIND and REBIND, Db2 13 introduces the new keyword OWNERTYPE for the OWNER bind option.
- For the SQL CREATE and ALTER statements for compiled SQL scalar functions and native SQL procedures, Db2 13 introduces the new keywords AS ROLE and AS OWNER in the PACKAGE OWNER clause.

Object Comparison Tool 13.1 can compare the owner and owner type for these procedures and functions and generate changes as needed. For example, if the owner and owner type differ between the source object and the target object, the compare report includes a message similar to the following message:

```
Options
(A)Field PACKAGE OWNER changed from 'RL174061 AS ROLE' to 'TS5465 AS USER'
Native SQL Procedure options will be altered
```

#### Related information:

[Overview of what's new in Db2 13 \(Db2 13 for z/OS documentation\)](#)

[What's new in Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## Support for long column names

Db2 13 introduced support for long column names (up to 128 bytes) when the TABLE\_COL\_NAME\_EXPANSION subsystem parameter setting is ON. Previously, the limit was 30 bytes. Object Comparison Tool 13.1 can manage these longer column names. For example, you can compare objects with long column names and generate changes as needed.

### Related information:

[Overview of what's new in Db2 13 \(Db2 13 for z/OS documentation\)](#)

[Column names longer than 30 bytes \(Db2 13 for z/OS documentation\)](#)

[What's new in Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

### Related reference

“New-function APARs in Db2 Object Comparison Tool 13.1” on page 3

After GA, enhancements continue to be delivered later in the service stream, as part of new-function APARs.

### Related information

[Db2 13 function level support \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## New-function APARs in Db2 Object Comparison Tool 13.1

After GA, enhancements continue to be delivered later in the service stream, as part of new-function APARs.

The following table summarizes the APARs that introduce new function for Object Comparison Tool 13.1. It does not include problem fixes or other maintenance APARs.

Description	APAR	Date
<a href="#">“Support for REORG SCOPE PENDING” on page 4</a>	PH66759 PH66857	2025-06
<a href="#">“Improved processing when dropping columns” on page 5</a>	PH66867	2025-06
<a href="#">“New warning about tables with DATA CAPTURE CHANGES” on page 6</a>	PH65602	2025-03
<a href="#">“Db2 13 FL506: Support for adding implicitly hidden ROWID columns” on page 6</a>	PH63985 PH63986	2024-11
<a href="#">“Simplified compare job option selection” on page 6</a>	PH63721 PH63722	2024-10
<a href="#">“Ability to generate DDL for hidden ROWID columns” on page 8</a>	PH63073 PH63074	2024-09
<a href="#">“Ability to ignore column order when comparing tables” on page 9</a>	PH62527 PH62528	2024-08
<a href="#">“Non-disruptive ADD COLUMN NOT NULL with no default” on page 9</a>	PH60828	2024-04
<a href="#">“Support for CREATE TABLE LIKE statements when comparing DDL” on page 10</a>	PH60527	2024-04

<b>Description</b>	<b>APAR</b>	<b>Date</b>
<a href="#">“Ability to compare APPLCOMPAT for expression-based indexes” on page 10</a>	PH57692 PH57696 PH57698	2023-11
<a href="#">“Ability to use a SELECT statement to specify source and target objects” on page 10</a>	PH57198 PH57326	2023-10
<a href="#">“New CM batch options to include foreign keys changes when comparing objects” on page 11</a>	PH55583	2023-10
<a href="#">“Summary report can include original names of added objects” on page 11</a>	PH56749	2023-09
<a href="#">“Support for regenerating views” on page 12</a>	PH37650 PH55431	2023-07
<a href="#">“CM batch support for comparing DDL to DDL” on page 12</a>	PH54480	2023-05
<a href="#">“Masking support for removing a key label” on page 12</a>	PH54152	2023-05
<a href="#">“Improvements when transporting work statement lists to other systems” on page 12</a>	PH53482	2023-03
<a href="#">“Remove unnecessary changes when comparisons involve objects created before Db2 12” on page 13</a>	PH49601	2022-11
<a href="#">“Ability to specify REBIND options when altering objects” on page 14</a>	PH50333	2022-11
<a href="#">“REORG SHRLEVEL default change to avoid pending changes” on page 15</a>	PH49639	2022-09
<a href="#">“Improvements to inserting and adding partitions” on page 15</a>	PH48016	2022-08

## **Support for REORG SCOPE PENDING**

### **PH66759 (Db2 Admin Tool), PH66857 (Object Comparison Tool) - June 2025**

In Object Comparison Tool, you can now specify SCOPE PENDING when you run the REORG utility. You can also specify SCOPE PENDING when you run REORG in Db2 Admin Tool.

You can specify this new option on the **Specify Utility Options - REORG TABLESPACE (ADB2USO1)** panel:

```
ADB2USO n ----- DD1A Specify Utility Options - REORG ----- 19:35
Command ==>
```

```
Execute utility on alter object
using the following options:
```

```
More: +
LISTPARTS . . . . . (Positive integer)
CLONE . . . . . (Yes/No)
REUSE . . . . . (Yes/No)
SCOPE . . . . . (A - ALL, P - PENDING)
REBALANCE . . . . . (Yes/No)
  SORTCLUSTER . . . . . (Yes/No)
LOG . . . . . (Yes/No)
DROP_PART . . . . . (Yes/No)
SORTDATA . . . . . YES (Yes/No)
  RECLUSTER . . . . . (Yes/No)
NOSYSREC . . . . . (Yes/No)
SORTKEYS . . . . . (Yes)
SHRLEVEL . . . . . R (N - None, R - Reference, C - Change)
FASTSWITCH . . . . . (Yes/No)
FORCE . . . . . (A - All, R - Readers, N - None)
SORTNPSI . . . . . (A - Auto, Y - Yes, N - No)
OFFPOSLIMIT . . . . . (integer)
INDREFLIMIT . . . . . (integer)
  REPORTONLY . . . . . (Yes/No)
UNLOAD . . . . . E (C - Continue, P - Pause, O - Only, E - External)
  NOPAD . . . . . (Yes/No)
KEEPDICTIONARY . . . . . (Yes/No)
STATISTICS . . . . . (Yes/No)
  TABLE schema . . . . . >
    name . . . . . > (ALL)
```

You can also specify this option in Change Management batch by using the new parameter UTIL\_REORG\_SCOPE.

#### Related information:

[New-function APARs for Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[PH66759](#)

[PH66857](#)

## Improved processing when dropping columns

### PH66867- June 2025

Previously, when Object Comparison Tool needed to drop a column, some dependent objects prevented this change from being made with an ALTER operation. Instead, the table had to be dropped and recreated and informational message ADB7191I or ADB7192I was issued.

With this APAR, Object Comparison Tool can now drop columns with dependent objects in a less disruptive operation by using an ALTER TABLE DROP COLUMN statement, if Db2 allows the operation. [Db2 restricts certain DROP COLUMN operations; see ALTER TABLE (Db2 13 for z/OS).] As part of this change, Object Comparison Tool drops and recreates the dependent objects.

Specifically, this APAR removes the following restrictions for using an ALTER operation to drop a column:

- The column is referenced by a unique constraint.
- The column is referenced by a referential constraint.
- The column is referenced by views or triggers.
- The table has check constraints.
- The table has a validation exit procedure.
- The table is referenced by row permissions or column masks.
- INSTEAD OF triggers are defined on a view that is dependent on the table.
- The table is referenced by extended indexes or inline SQL table functions.
- The table is referenced by materialized query tables.

#### Related information:

[ADB7191I \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[ADB7192I \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[PH66867](#)

## **New warning about tables with DATA CAPTURE CHANGES**

### **PH65602- March 2025**

When Object Comparison Tool needs to drop and re-create a table that has the DATA CAPTURE CHANGES attribute and is involved in replication, additional actions are required to ensure that replication continues. Replication must be stopped before the table is dropped and restarted after the table is re-created. To help you remember to take these actions, Object Comparison Tool now returns a new warning, ADB7415W, for any tables with DATA CAPTURE CHANGES that will be dropped and re-created. You can then determine whether the table is involved in replication and whether additional actions are necessary.

#### **Related information:**

[ADB7415W \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[PH65602](#)

## **Db2 13 FL506: Support for adding implicitly hidden ROWID columns**

### **PH63985, PH63986 - November 2024**

Db2 13 FL 506 Starting in Db2 13 function level 506, you can add an implicitly hidden ROWID column to a table in a UTS table space. Specifically, you can issue an ALTER TABLE ADD COLUMN statement with ROWID IMPLICITLY HIDDEN. This ALTER operation is supported in Db2 Object Comparison Tool. If an implicitly hidden ROWID column needs to be added to a target table, this change can now be done by using an ALTER statement rather than dropping and recreating the object. Making this change by using an ALTER operation is less disruptive to your subsystems than dropping and recreating the table.

#### **Related information:**

[Db2 13 function level 506](#)

[PH63985](#)

[PH63986](#)

## **Simplified compare job option selection**

### **PH63721 (Db2 Admin Tool), PH63722 (Object Comparison Tool) - October 2024**

When you use Object Comparison Tool to compare objects, you can specify compare job options on the **Generate Compare Jobs (GOC5)** panel:

```

GOC5 ----- Generate Compare Jobs -----
Command ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
  Worklist name . . . . . PQ76055N (also used as middle qualifier in DSNs)

Compare options:
  Suppress DROP of objects . NO          (Yes/No)
  Drop FKs not in source . . NO          (Yes/No)
  Suppress DROP of columns . NO          (Yes/No)
  Suppress adding columns . NO           (Yes/No)
  Run SQLID . . . . . (Blank, an SQLID, or <NONE>)
  Object Grantor . . . . . (Blank or an SQLID)
  Run Validate. . . . . V (Validate, None)
  Allow implicit drop of
    excluded objects . . . . NO          (Yes/No)
  Enable auth-switching . . . YES        (Yes/No)
  Disable REORG optimization YES        (Yes/No)
  Scope Warning Messages . . YES        (Yes/No)

Change reporting options . . YES         (Yes/No)
Save compare results . . . . YES        (Yes/No)

Data set information:
  PDS for batch jobs . . . . CMP.PQ76055N
  Prefix for data sets . . . NBRON
  Changes file data set name.
    Member name . . . . . (if Changes file is an existing PDS)

Options:
  Generate online . . . . . YES          (Yes/No)
  Single compare job . . . . NO          (Yes/No)
    Member name . . . . . COMPARE        (default COMPARE)
  Allow deferred restart . . NO          (Yes/No)
  Generate apply jobs . . . . YES        (Yes, No, or (Delta) Change)
    Generate one job. . . . . YES        (Yes, No, or (Per) Process)
    Member prefix . . . . . APPLY        (default APPLY)
  As work statement list . YES           (Yes/No to append to work stmt list)
    Embed IFF into WSL . . NO           (Yes/No)
  Use customized util opts. YES          (Yes/No)
  Content of apply job(s) . ALL          (All, DDL)
  Unload method . . . . . P (Unload, Parallel unload, HPU)
  Generate templates. . . . . NO        (Yes/No)
  Stop on conversion error. NO           (Yes/No)
  Use DEFER YES . . . . . YES           (Yes/No)
  Allow rotate parts . . . . YES         (Yes/No)
  Retain GENERATED ALWAYS:
    For ROWID . . . . . YES             (Yes/No)
    For ROW CHANGE TIMESTAMP. YES        (Yes/No)
  Retain START and RESTART values:
    For sequence object . . . . (Yes/No)
  IDENTITY START value . . . ORIGINAL    (Original, Computed)
  Mask ignored fields . . . . NO         (Yes/No)

  Optional jobs after Reload or Alter:
    Run CHECK DATA . . . . YES          (Yes/No)
    Take an image copy . . . R           (after: Reload/Alter/Both/None)
    Run REORG/REBUILD . . . M           (Mandatory, All relevant, None)
    Run RUNSTATS . . . . . R            (after: Reload/Alter/Both/Min/None)
    Run REBIND . . . . . M              (Mandatory, All relevant, None)
    REBIND options . . . . YES          (Yes/No)

BP - Change batch job parameters
TU - Specify TEMPLATE usage
UO - Customize utility options
CO - Change options common to change functions

```

To simplify the process of specifying these options, this panel now has a simple display mode, which displays only the essential options that must be specified for the comparison:

```

GOC5 re ----- Generate Compare Jobs -----
Command ==>

Specify the following for DB2 Object Comparison Tool:
                                         More:   +
Worklist information:
  Worklist name . . . . . GOC1      (also used as middle qualifier in DSNs)

Compare options:
  Run SQLID . . . . .              (Blank, an SQLID, or <NONE>)

Change reporting options . . YES      (Yes/No)

Data set information:
  PDS for jobs . . . . . TS6462.GOC.JCL
  Prefix for data sets . . . TS6462.GOC
  Changes file data set name. TS6462.GOC.CHANGE1
  Member name . . . . .           (if Changes file is an existing PDS)

Options:
  Compare job member name . . COMP1   (default COMPARE )
  Generate apply jobs . . . . NO      (Yes, No, or (Delta) Change)
  Generate one job . . . . YES        (Yes, No, or (Per) Process)
  Member prefix . . . . . APPL1      (default APPLY )
  Use customized util opts NO        (Yes/No)
  Generate templates. . . . YES       (Yes/No)

BP - Change batch job parameters
TU - Specify TEMPLATE usage
UO - Customize utility options
CO - Change options common to change functions
ADV - Advanced options

```

You can still view or change other options by switching to the advanced display mode. The advanced display mode includes all compare job options. To toggle between the simple and advanced display modes, use the new SIM and ADV commands. You can also change the display mode in the Db2 Admin Tool display options on the **Panel Display Options (ADBPPDO)** panel

When you first install Db2 Admin Tool, as a new user, the **Generate Compare Jobs (GOC5)** panel is initially displayed in simple display mode. When you upgrade Db2 Admin Tool, this panel will be initially displayed in advanced display mode.

#### Related information:

[“Simple and advanced display modes” on page 89](#)

[“Compare job options” on page 87](#)

[“5. Generating a compare job” on page 84](#)

[PH63721](#)

[PH63722](#)

## Ability to generate DDL for hidden ROWID columns

### PH63073, PH63074 - September 2024

Previously, when Object Comparison Tool generated DDL for a table with an implicitly generated ROWID column, the ROWID column was not included in the generated CREATE TABLE statement. With this APAR, you can now specify that you want the generated CREATE TABLE statement to include the ROWID column with the IMPLICITLY HIDDEN attribute.

In Object Comparison Tool, you can specify that you want these ROWID columns included in the DDL by using the new **Generate hidden ROWID columns** option on the **Options for Change Functions (ADB2PCO)** panel. You can specify that you want the implicitly generated ROWID column to always be included in the CREATE TABLE statement (ALWAYS), to be included only if it is not the last column in the table (ONLY), or to not be included (NO). By default, the ROWID column is not included in the CREATE TABLE statement.

#### Related information:

[“CO - Change options common to change functions” on page 109](#)

[GENROWID](#)

[PH63073](#)

[PH63074](#)

## Ability to ignore column order when comparing tables

### PH62527, PH62528 - August 2024

When you use Object Comparison Tool to compare tables, you can now specify that you want column order to be ignored. When column order is not relevant, ignoring it can help minimize any resulting changes. In some cases, where column order is the only difference, no changes are needed. In other cases, column changes can be made by using ALTER operations rather than dropping and re-creating the table.

For example, consider the case where the source table has one more column than the target table and the source table columns are in a different order. If column order is not ignored, the target table is dropped and re-created to achieve the changes. If column order is ignored, the change can be achieved by an ALTER TABLE statement.

To specify that you want to ignore column order, set the new Change Management batch option **IGNORE\_COLUMN\_ORDER** to YES or set the new panel option **Ignore source column order** to YES on the **Options for Change Functions (ADB2PCO)** panel:

```
ADB2PCO n                Options for Change Functions                14:19
Command ==>

                                DB2 System: DD1A

Recreate accelerated tables . . . . . YES (Yes/No. Default is Yes)
Restore replication of tables . . . . . YES (Yes/No. Default is Yes)
Reload accelerated tables . . . . . YES (Yes/No. Default is Yes)
Restore acceleration of tables . . . . . YES (Yes/No. Default is Yes)
Remove deleted accelerated tables . . YES (Yes/No. Default is Yes)

Load accelerated tables LOCKMODE . . . . NONE (Default is TABLESET)
Load accelerated tables DETECTCHANGES DATA (Default is DATA)
Unload altered tables . . . . . NO (Yes/No/Des. Default is YES)
Preserve all data . . . . . YES (Yes/No. Default is YES)

Enable WSL authorization switching . . NO (Yes/No. Default is No)
Object processing order . . . . . H (T - Object type, H - DB hierarchy.
                                     Default is H)
Statement validation exit name . . . . (Name of EXEC used to validate
                                     statements in WSL Validate)

Allow PBR2 to PBR changes . . . . . NO (Yes/No. Default is No)
DB2 release number . . . . . 1215 (Use VVRM format)
DB2 function level . . . . . 504 (E.g. 100, 500, 501, 5nn)
GRANT processing order . . . . . C (C - CREATE prefix for GRANT
                                     P - POSTUTIL prefix for GRANT
                                     Default is C )

Auto view regenerate . . . . . NO (Yes/No. Default is No)
Ignore source column order . . . . . NO (Yes/No. Default is No)
```

### Related information:

[“Compare job options” on page 87](#)

[PH62527](#)

[PH62528](#)

## Non-disruptive ADD COLUMN NOT NULL with no default

### PH60828 - April 2024

When you compare objects with Object Comparison Tool, and the resulting changes to the target include appending a column that is NOT NULL with no default, that change is no longer disruptive. Previously,

such a change was made by dropping and re-creating the table. Now, this change is accomplished by altering the table as follows:

```
ALTER TABLE table ADD COL NOT NULL WITH DEFAULT
REORG TABLESPACE table_space
ALTER TABLE table ALTER COL DROP DEFAULT
```

A REORG operation is needed in this case to clear the restrictive state. Therefore, set the **Run REORG/REBUILD** option on the **Generate Compare Jobs (GOC5)** panel to M or A. Or, if you are using CM batch, set the RUN\_REORG\_REBUILD parameter to M or A.

**Restriction:** This ALTER operation is not used in the following situations:

- For tables with DATA CAPTURE CHANGES
- For base tables with materialized query tables
- If the appended column is part of a check constraint or a business period

**Related information:**

[PH60828](#)

## Support for CREATE TABLE LIKE statements when comparing DDL

**PH60527 - April 2024**

Db2 Object Comparison Tool now supports CREATE TABLE LIKE statements in DDL that is used for comparisons.

When you specify source and target objects for a comparison, one way to specify the object definitions is to use DDL. Previously, CREATE TABLE LIKE statements were not supported in this DDL. This restriction is now removed. The DDL that is used for source or target object definitions can now contain CREATE TABLE LIKE statements. However, the table in the LIKE clause must be defined in the same DDL.

**Related information:**

[“Specifying a DDL file for the source or target definition” on page 57](#)

[“Supported SQL statements for DDL file extraction” on page 171](#)

[PH60527](#)

## Ability to compare APPLCOMPAT for expression-based indexes

**PH57692 (Db2 Admin Tool), PH57696 (Object Comparison Tool), PH57698 - November 2023**

When you compare expression-based indexes, the APPLCOMPAT values are compared. If you do not want to compare these values, you can now ignore them by using the new ignore SYSENVIRONMENT APPLCOMPAT.

Db2 Admin Tool also now provides the ability to manage and change APPLCOMPAT values for expression-based indexes. For more information about these enhancements, see [2023 new-function APARs for Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

**Related information:**

[“4. Specifying ignores” on page 80](#)

[PH57692](#)

[PH57696](#)

## Ability to use a SELECT statement to specify source and target objects

**PH57198, PH57326 - October 2023**

In Db2 Object Comparison Tool, you specify the objects to be compared by selecting either DDL, objects from the Db2 catalog, or a compare version file. With this enhancement, you now have another option: you can identify the source and target objects to be compared by using an SQL SELECT statement against

the Db2 catalog. Object Comparison Tool uses all objects that are returned by the query for the source or target definition.

Using a SELECT statement to identify the source and target objects can be more efficient than individually specifying objects from the Db2 catalog. Additionally, when using a SELECT statement, you can use clauses to filter the object list.

To specify a SELECT statement, you must first select the Db2 catalog as the source of the object definitions [option 2 on the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel]. Then you can select the new option: **5 - Source is the result of an SQL SELECT statement**. The SQL statement that you specify must return certain columns and can optionally return other columns. For detailed instructions and requirements, see [“Specifying a SELECT statement for the source or target definition” on page 64](#).

You can also specify a SELECT statement when using Change Management (CM) batch to run a comparison. In this case, set the SOURCE\_TYPE parameter, TARGET\_TYPE parameter, or both to USER and use a quick scope to specify the SELECT statement. For detailed instructions and examples, see Chapter 16, [“Running Compare by using a Change Management batch job,” on page 187](#).

**Related information:**

[PH57198](#)  
[PH57326](#)

## **New CM batch options to include foreign keys changes when comparing objects**

### **PH55583 - October 2023**

When running the CM batch interface JCL procedure (GOCCM) to compare objects, you can now specify whether foreign key changes should be included in the generated DDL. To do so, use the following new CM batch parameters:

- SOURCE\_GEN\_FOREIGN\_KEYS
- TARGET\_GEN\_FOREIGN\_KEYS

This functionality is similar to the GENRELS parameter in the JCL that is generated from the **Generate Compare Jobs (GOC5)** panel.

**Related information:**

[CM batch parameter definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)  
[PH55583](#)

## **Summary report can include original names of added objects**

### **PH56749 - September 2023**

When you compare objects with Object Comparison Tool, the resulting changes might include adding objects to the target. In this case, those new object names might be masked and therefore different than the original object names in the source. To help you determine which source object was added, you can now request that Object Comparison Tool report the original object name in addition to the new name when running change management (CM) batch. To do so, set the new CM batch parameter REPORT\_ORIGINAL\_NAMES\_ADDED\_OBJECTS to YES. When this parameter is set to YES, the summary report includes the original name (under Source Object) and the new masked name (under Target Object), as shown in the following example:

```
COMPARISON SUMMARY REPORT
=====
```

```
Obtyp Source Object                Target Object                Result
Object type
-----
```

### Related information:

[CM batch parameter definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)  
[PH56749](#)

## Support for regenerating views

### PH37650, PH55431 - July 2023

Object Compare now supports regenerating views if needed when the source and target views have different APPLCOMPAT values. Specifically, if the source and target views are the same, but the APPLCOMPAT values are different, and the source view APPLCOMPAT value is lower than current APPLCOMPAT value, Object Compare generates the following statement:

```
ALTER VIEW view_name REGENERATE USING APPLICATION COMPATIBILITY source_applcompat
```

### Related information:

[PH37650](#)  
[PH55431](#)

## CM batch support for comparing DDL to DDL

### PH54480 - May 2023

You can now compare DDL to DDL by using Change Management (CM) batch. Previously, this type of comparison was allowed only by using the Object Comparison Tool panels. To do this comparison, use the new DDL value for the TARGET\_TYPE CM batch parameter. For details, see [2023 new-function APARs for Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

### Related information:

[Examples: Invoking the Change Management \(CM\) batch interface for various actions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)  
[PH54480](#)

## Masking support for removing a key label

### PH54152 - May 2023

When comparing objects, you can overwrite a key label value for a storage group by using the existing mask SGKEYLABL or a table by using the existing mask TBKEYLABL. This APAR adds support to these masks for removing a key label. You can now specify the value NOKEYLABEL (or NO) for these masks to remove a key label. For example:

```
TBKEYLABL : TBCRE .MYTB ,NOKEYLABEL**  
TBKEYLABL : TBCRE .MYTB ,NO**
```

### Related information:

[“Mask data set” on page 76](#)  
[PH54152](#)

## Improvements when transporting work statement lists to other systems

### PH53235 (Db2 Admin Tool), PH53482 (Object Comparison Tool) - March 2023

When you compare objects by using Db2 Object Comparison Tool, you can store the resulting changes in a work statement list (WSL). If those changes require an unload operation, Object Comparison Tool generates an IFF file. If you then transport the WSL to another system, the IFF file must also be

transported independently. To simplify this process of transporting WSLs, a new option is now available to embed the IFF file directly in the WSL. When you specify this option, all the information in the IFF file is embedded in the WSL in an encoded format. You can then transport the WSL to another system without needing a separate IFF file.

This new **Embed IFF into WSL** option is available on the **Generate Compare Jobs (GOC5)** panel when comparing objects. The default value for this new option is NO.

This option is also available on several Db2 Admin Tool panels. See [2023 new-function APARs for Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

**Related information:**

[“Compare job options” on page 87](#)

[PH53235](#)

[PH53482](#)

## **Remove unnecessary changes when comparisons involve objects created before Db2 12**

### **PH49601 - November 2022**

If a table space was created before Db2 12, certain table space attributes that are new in version 12 might still be set to NULL in the Db2 catalog for that table space. If you use Db2 Object Comparison Tool to compare one of these table spaces in the catalog (the target) with DDL that was created in Db2 12 or later (the source), and the first partition values on the target are the same as the source table-space-level attributes, no ALTER statements should be generated for these attributes.

This APAR ensures that Object Comparison Tool does not generate unnecessary changes for the following attributes when comparing table spaces and the target is a table space in the catalog that was created before Db2 12:

- PQTY
- SECQTYI
- STORTYPE
- STORNAME
- VCATNAME
- PCTFREE
- PCTFREE\_UPD
- TRACKMOD
- COMPRESS
- FREEPAGE
- GBPCACHE

A similar situation exists for indexes that were created before Db2 12. This APAR also ensures that Object Comparison Tool does not generate unnecessary changes for the following index attributes when comparing indexes and the target is an index in the catalog that was created before Db2 12:

- PQTY
- SECQTYI
- STORTYPE
- STORNAME
- VCATNAME
- FREEPAGE
- PCTFREE
- GBPCCAHE

**Related information:**

[PH49601](#)

**Ability to specify REBIND options when altering objects**

**PH50333 - November 2022**

When a comparison results in a change that requires an object to be altered, you can choose whether to rebind any dependent packages. Before this APAR, these packages were rebound with their existing BIND options (the options that were used during the previous bind or rebind operation). With this APAR, you can now specify different BIND options. For example, you can specify APREUSE(ERROR) to help retain existing access paths.

To specify REBIND options, use the new **REBIND options** field on the **Generate Compare Jobs (GOC5)** panel:

```
GOC5 re ----- Generate Compare Jobs -----
Command ==>

Specify the following for DB2 Object Comparison Tool:
....

  For ROWID . . . . . NO          (Yes/No)
  For ROW CHANGE TIMESTAMP. NO    (Yes/No)
  Retain START and RESTART values:
  For sequence object . . . . . (Yes/No)
  IDENTITY START value . . . ORIGINAL (Original, Computed)
  Mask ignored fields . . . . NO    (Yes/No)

  Optional jobs after Reload or Alter:
  Run CHECK DATA . . . . NO       (Yes/No)
  Take an image copy . . . N       (after: Reload/Alter/Both/None)
  Run REORG/REBUILD . . . M       (Mandatory, All relevant, None)
  Run RUNSTATS . . . . N          (after: Reload/Alter/Both/Min/None)
  Run REBIND . . . . . A          (Mandatory, All relevant, None)
  REBIND options . . . . YES     (Yes/No)

BP - Change batch job parameters
TU - Specify TEMPLATE usage
UO - Customize utility options
CO - Change options common to change functions
```

When you specify Yes in this new field (and either M or A in the **Run REBIND** field), the **REBIND options (ADBPREBO)** panel is displayed where you can specify the following options:

```
ADBPREBO ----- REBIND options -----
Command ==>

Specify additional REBIND parameters to generate rebinds
for dependent packages.

APREUSE . . . . . (None, Warn, Error)
EXPLAIN . . . . . (Yes, No, Only)
OWNER . . . . . > (Owner of package)
OWNERTYPE . . . . (Role, User)

Additional options:
-----
-----
-----
-----
-----
```

Any BIND options that you specify in the **Additional options:** field are added as is; they are not validated.

**Related information:**

[“5. Generating a compare job” on page 84](#)  
[PH50333](#)

**REORG SHRLEVEL default change to avoid pending changes****PH49639 – September 2022**

When a comparison results in a change that requires a REORG utility operation and no value is specified for the REORG SHRLEVEL option, Object Comparison Tool generates a REORG statement with a default value for SHRLEVEL. With this APAR applied, SHRLEVEL NONE will no longer be generated for this situation, because it prevents pending definition changes from being materialized and can leave objects in a pending state. Instead, to ensure that any pending changes are materialized successfully, either SHRLEVEL CHANGE or SHRLEVEL REFERENCE is used; Object Comparison Tool determines the best value (CHANGE or REFERENCE) depending on the circumstance.

As usual, you can override this behavior by specifying a value for SHRLEVEL and setting **Use customized util opts** to YES on the **Generate Compare Jobs (GOC5)** panel. If you specify SHRLEVEL NONE and a pending change exists, a warning is issued.

This change also applies to Change Management and the ALT command in Db2 Admin Tool. For more details, see [New-function APARs for Db2 Admin Tool 13.1 \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

**Related information:**

[PH49639](#)

**Improvements to inserting and adding partitions****PH48016 – August 2022**

Db2 Object Comparison Tool is enhanced to improve how partitions are added and inserted. In some cases, these changes reduce unnecessary and potentially costly REORG utility operations.

Prior to this APAR, when comparing partition-by-range (PBR) table spaces where the source has more partitions than the target, Object Comparison Tool generated the following statements for each partition to be added:

1. ALTER TABLE statement with the ADD PARTITION (MAXVALUE) clause.
2. REORG utility statement.
3. ALTER TABLE statement with the ALTER PARTITION clause to alter the added partition with the new limit key.

With this APAR applied, a single ADD PARTITION is generated, which also reduces the REORG statements that are generated in some cases.

**Related information:**

[PH48016](#)

## The comparison process

---

Db2 Object Comparison Tool compares Db2 object definitions (and the definitions of dependent objects) and reports the differences. As part of this process, Object Comparison Tool can optionally generate jobs to modify the objects to remove these differences.

The two sets of Db2 objects that are compared are called *source* and *target* objects. A *source object* is an object as you want it defined. A *target object* is an object that you want to match the source object.

When you select the source and target objects to compare, Object Comparison Tool extracts definitions of the objects and places them in a sequential data set called a *version file*. A *version file* is an internal representation of object definitions and represents a snapshot at a particular point in time. Two separate

version files are created, one for the source object and one for the target object. These version files are created before the objects are compared.

You can specify any of the following sources that you want Object Comparison Tool to use for the object definition:

**DDL file**

A file that contains data definition language (DDL), such as a SPUFI file. When the source is DDL, Object Comparison Tool processes everything in the DDL file. Objects are not selected based on type or name. If you are comparing DDL and your DDL only defines a table, only that table is used.

**Db2 catalog**

An extract of information from the Db2 catalog for one or more databases, table spaces, or tables and all the dependent objects. When the definition source is a Db2 catalog, Object Comparison Tool includes all dependent objects, such as views and indexes, in the comparison. These dependent objects are included regardless of whether you specify objects at the database level, the table space level, or the table level.

**Version file**

A version file that was created during a previous comparison. If a version file is used as the source of the comparison, a new version file is not created.

Using Object Comparison Tool, you can do any of the following comparisons:

Definition source for the source object	Definition source for the target object
Db2 catalog	Db2 catalog
DDL file	DDL file
Version file	Version file
DDL file	Db2 catalog
DDL file	Db2 catalog with objects that are automatically selected based on the source specification

After the source and target version files are created, Object Comparison Tool compares them and creates a difference file or *changes file*. Object Comparison Tool then generates reports that show the differences between the objects and, if requested, DDL to apply any changes to the target object. After reviewing the report, you can direct the generated DDL for the target object to apply jobs. Apply jobs can be stored in a work statement list (WSL) or a partitioned data set (PDS) and then propagated to several remote sites. This process allows for changes in a test environment to be easily migrated to the development or production environment.

As part of the comparison process, you can use masking and ignore fields to account for intentional differences between the objects, so that only the actual differences are reported. *Masking* handles different naming conventions between the objects that you are comparing. For example, the same object might have an owner name of TESTxxx on the test system and an owner name of PRODxxx on the production system. *Ignore fields* handle attribute differences between the objects that you are comparing. For example, primary and secondary quantities usually differ between test and production systems.

**Related concepts**

[“Components of the comparison process ” on page 17](#)

Db2 Object Comparison Tool compares objects by reading the Db2 catalog or DDL files. Object Comparison Tool produces comparison reports and then optionally generates either JCL jobs or work statement list (WSL) tasks with changes for the target objects.

## Components of the comparison process

Db2 Object Comparison Tool compares objects by reading the Db2 catalog or DDL files. Object Comparison Tool produces comparison reports and then optionally generates either JCL jobs or work statement list (WSL) tasks with changes for the target objects.

The following figure shows the detailed flow of processes in Db2 Object Comparison Tool:

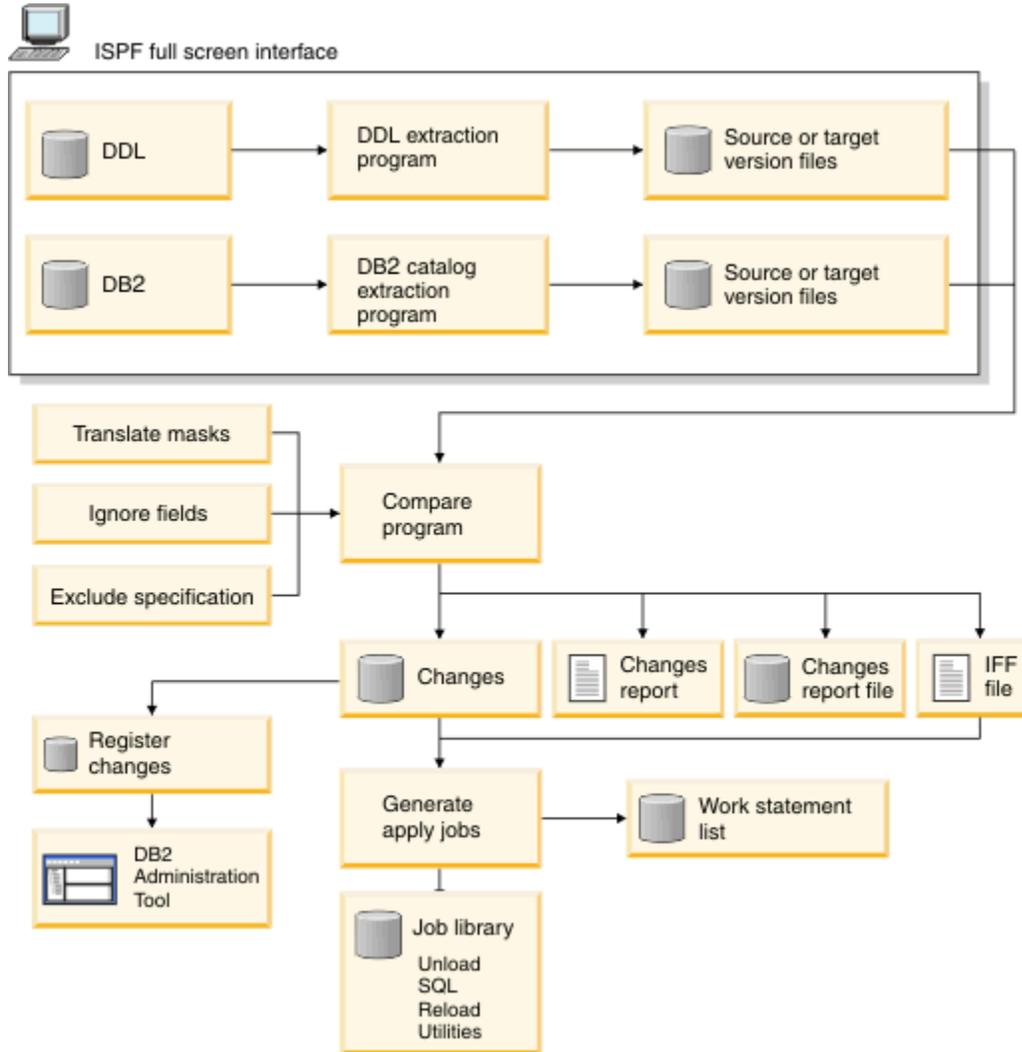


Figure 1. Db2 Object Comparison Tool processes and components

This figure includes the following processes and components:

### DDL extraction program

This program reads object definitions from DDL files into a version file.

### Db2 catalog extraction program

This program reads object definitions from the Db2 catalog into a version file.

### Compare program

This program compares two version files, produces a report to describe any differences, and generates the information that is needed to apply changes to the target object. This program accounts for any specified masks, ignore fields, or exclude specifications when doing the comparison.

### Program to register the changes

This program registers the changes in the Change Management (CM) database in Db2 Administration Tool, where you can then analyze and run the job.

### Function to generate apply jobs

This function performs one of the following operations:

- Creates the UNLOAD, DROP, CREATE, ALTER, and LOAD jobs that are necessary to apply the changes to the target object.
- Creates WSL tasks to apply the necessary changes to the target object.

### Related concepts

[“The comparison process” on page 15](#)

Db2 Object Comparison Tool compares Db2 object definitions (and the definitions of dependent objects) and reports the differences. As part of this process, Object Comparison Tool can optionally generate jobs to modify the objects to remove these differences.

## Terminology in Db2 Object Comparison Tool

---

Db2 Object Comparison Tool uses several terms that are unique to the product.

### Alternate form of syntax

Another acceptable syntax for a statement.

Certain functions in Object Comparison Tool and Db2 Administration Tool (Db2 Admin Tool) support or produce statements that are used by Db2 for z/OS or by these two products. IBM might provide an alternate statement or alternate form for clauses in statements. IBM might identify one as the preferred syntax while still supporting the alternate form.

Object Comparison Tool and Db2 Admin Tool might use preferred or alternate forms of syntax. If the statement produced is accepted by the products or by Db2, the statement is considered valid. When necessary to produce an accepted statement, the products convert to the newer syntax. However, the products might retain older syntax even if Db2 considers the newer syntax the preferred syntax. This situation might be the case even if no possible use of the older syntax is needed. The use of older syntax might persist until IBM no longer supports it.

### Changes file

The file that Object Comparison Tool creates when the source and target objects are compared. This file is used by Object Comparison Tool to generate a report of the differences between the objects. This file is also used by the generate apply jobs function.

The changes file contains the following items:

- DROP, CREATE, and ALTER statements
- UNLOAD requests
- Table space information records, which allow the generate apply jobs function to determine the size of the UNLOAD jobs

The name of a typical changes file might be NBRON.PQ76055N.CHANGES.

### CMDELTA mode

A mode of operation in which the compare process generates a delta change in the CHANGES file instead of an apply job. This delta change is intended to be imported into Change Management (CM) (whereas apply jobs can be executed directly).

CMDELTA mode is used if you specify **Generate apply jobs = CHANGE** on the **Generate Compare Jobs (GOC5)** panel or if you specify YES on the **Change Management Prompt (ADB2CMRO)** panel. In CM batch, CMDELTA mode is used if you specify **ACTION\_COMPARE = 'Y'** with **ACTION\_IMPORT\_CHANGE = 'Y'**.

The statements that are generated by CMDELTA mode are intended for the register process of CM. The reports produced are unique to this mode. Messages indicating that an action will be taken for those changed objects, such as the object will be dropped or altered, are not applicable in CMDELTA mode

and should be ignored. Instead, use the reports produced by the analyze phase of CM to understand the actions taken on the changed objects.

Additionally, CMDELTA mode does not process dependencies between objects. Therefore, CMDELTA can show that only one object is changed; the analyze process will then show any dependent objects that are also affected and will be dropped and re-created.

**Exclude**

A specified object or authorization to exclude from input to the compare process.

**Exclude specification**

A specification that lists objects that you want to exclude from the compare process.

**Interchange File Format (IFF) file**

A file that is produced by the compare program. This file and the changes file are used by Object Comparison Tool to generate the apply jobs.

**Ignore change**

A specified change to an object that you want to ignore.

**Ignore change specification**

A specification that identifies changes that you want ignored during the comparison process. You can select the changes that you want ignored from a saved comparison report. Object changes that you specify as ignored are reported, but no SQL statements are generated for the changes.

**Ignore fields**

Fields that Object Comparison Tool ignores when comparing Db2 catalog records.

**Source**

The structure of the objects as you want them to look. For example, the source can be the structure of objects in a development environment. The source can be from DDL, a version file, or the Db2 catalog.

**Suppress DROP of objects**

An option that prevents dropping objects that exist in the target but not in the source.

By default, Object Comparison Tool drops objects from the target that are not in the source. For example, if the source contains only object A, but the target contains both objects A and B, Object Comparison Tool drops object B. This behavior is the default.

To change this default behavior, set the **Suppress DROP of objects** option to Yes. Generally, you should set this option to Yes if your source is a subset of the target and you want to avoid possible dropped objects. For example, if you specify DDL as the source and a database in the Db2 catalog as the target, your catalog contains many tables other than the one table that you are changing. Because all the additional tables are not in the source, those tables are dropped unless you specify Suppress DROP of objects =Yes.

**Target**

The destination for the changes. For example, the target can be a production system. The target is where the differences from the source can be applied to make the target the same as the source. The target definition can be an explicit specification of DDL, a version file, or the Db2 catalog, or an implicit selection of objects based on the source.

In the situation where you want to change the structure of your production system to match the structure of your development system, the development system is considered the source and the production system is considered the target. In another scenario, you might want to identify the differences between two sets of objects, without applying any changes. In this case the source and target represent two different sets of objects that are being compared.

**Translation mask**

A functionality that allows a match to be found when the source and target objects use different naming conventions. Before Object Comparison Tool compares Db2 catalog record fields, masks are applied to owner and name fields.

**Version file**

An internal representation of object definitions. Object Comparison Tool creates a version file for each source and target and then uses those files to perform a comparison.

A version file is a variable-length data set that contains all the information that was extracted about the Db2 objects. The version file contains a header record and all the Db2 catalog records that represent the objects. The records in a version file are prefixed with information that allows the compare process to sort the records but also keep multiple records for the same object together.

Version files can be saved for subsequent comparison operations. You can also use them to restore application objects to a previous version (undo) or compare a new version with several production versions (clones) of the objects.

## Product documentation and updates

---

The documentation for Db2 Object Comparison Tool is regularly updated with information about new features and any corrections.

The Object Comparison Tool documentation is available in the following two formats:

### Topics in IBM Documentation

Underneath the title of each topic, you can see the date it was last updated.

You can find IBM Db2 Object Comparison Tool for z/OS in IBM Documentation at <https://www.ibm.com/docs/en/db2objectcompare>

**Tip:** When searching IBM Documentation, use quotation marks to ensure exact matches only. For example, the search term "ADB226E" returns only those topics that contain ADB226E. If you do not use quotation marks, close or partial matches might be returned. For example, a search on ADB001E might return ADB901E.

### PDF file

The PDF file is titled "IBM Db2 Object Comparison Tool for z/OS User's Guide." The date when the PDF file was created is listed at the bottom of page 2, near the copyright information.

The latest copy of the PDF file is always posted at [https://www.ibm.com/docs/en/SSAUVH\\_13.1.0/pdf/gocug131.pdf](https://www.ibm.com/docs/en/SSAUVH_13.1.0/pdf/gocug131.pdf).

Both of these formats contain the same information and are updated at the same time.

### Revision marks for changed content

Revisions for the following types of content changes are marked like this sentence, with bars in the left margin:

- Technical revisions for changed externals that are introduced by the new release or by maintenance after the general availability of this release.
- Technical clarifications in response to customer and internal feedback.

Editorial and organizational changes that do not affect the technical meaning of the content are generally not marked.

### How to send your comments

Your feedback is important in helping IBM provide the most accurate and highest quality information. If you have any comments about this or any other Db2 Tools information, see [How to provide feedback in IBM Documentation](#).

When you provide feedback, include as much information as you can about the content you are commenting on, where it is found, and what your suggestions for improvement might be.

## Accessibility features

---

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use a software product successfully.

The major accessibility features in this product enable users to perform the following activities:

- Use assistive technologies such as screen readers and screen magnifier software. Consult the documentation for the specific assistive technology for information about using it to access z/OS interfaces.
- Customize display attributes such as color, contrast, and font size.
- Operate specific or equivalent features by using only the keyboard. See the following publications for information about accessing ISPF interfaces:
  - [z/OS ISPF User's Guide, Volume 1](#)
  - [z/OS TSO/E Primer](#)
  - [z/OS TSO/E User's Guide](#)

These guides describe how to use the ISPF interface, including the use of keyboard shortcuts or function keys (PF keys), the default settings for the PF keys, and how to change their functions.



## Chapter 2. Customization

When you customize Db2 Admin Tool, you can also enable Db2 Object Comparison Tool for immediate use. At that time, if you choose not to enable the Db2 Object Comparison Tool, you can later customize the tool separately.

To customize Object Comparison Tool, use IBM Tools Customizer for z/OS 1.1 (5655-TC1), also known as TCz. Formerly a component of IBM Tools Base, TCz is a standard tool for customizing IBM tools that run on z/OS. It provides a single, common, and consistent ISPF interface for customization of these tools after installing them.

**Migration to a new Db2 version, mode, or function level:** When you migrate to a new Db2 version, mode, or function level, you do not need to recustomize Object Comparison Tool. Because the product relies on Db2 Admin Tool to access Db2, you need only recustomize Db2 Admin Tool.

### Related information

[Video playlist - IBM Tools Customizer for z/OS \(TCz\)](#)

[IBM Tools Customizer for z/OS 1.1.0](#)

## Customization checklist for Db2 Object Comparison Tool

The following checklist describes each significant customization step. Use this checklist to guide you through the entire customization process for Object Comparison Tool.

### Tips:

- Print this checklist and record your status during the customization process.
- If you are not familiar with Tools Customizer (TCz) and the customization process, consider reviewing the following terminology and other basic TCz information before you begin: [Tools Customizer terminology and data sets \(IBM Tools Customizer for z/OS 1.1\)](#)

Task	Link to detailed instructions	Status
<b>Verify software requirements</b>		
Verify that your environment meets the minimum software requirements.	<a href="#">“Software requirements for Object Comparison Tool” on page 24</a>	
<b>Verify SMP/E installation</b>		
Verify that Object Comparison Tool is installed. SMP/E installation instructions are in the program directory.  To verify that the installation completed correctly, specify the following command on any Db2 Admin Tool panel:  PANEL GOCMENU  The <b>DB2 Object Comparison Tool Menu (GOCMENU)</b> panel should be displayed. If this panel is not displayed, the installation was not successful and you must reinstall Object Comparison Tool.	<a href="#">Program Directory for Db2 Object Comparison Tool 13.1 (GI13-4643)</a>	
Verify that TCz is installed. SMP/E installation instructions are in the program directory.	<a href="#">Program Directory for IBM Tools Customizer for z/OS 1.1 (GI13-4653)</a>	
<b>Gather data set names</b>		

Task	Link to detailed instructions	Status
Record the data set names that you will need during the customization process.	<a href="#">“Data sets used by Tools Customizer” on page 25</a>	
<b>Optional: Determine LPAR strategy</b>		
If you have a multiple-LPAR environment, determine your customization strategy.	<a href="#">“Using Tools Customizer in a multiple-LPAR environment” on page 29</a>	
<b>Customize Db2 Object Comparison Tool</b>		
Complete the steps in the appropriate customization roadmap based on the type of customization that you are performing.	<a href="#">“Roadmap: Customizing Db2 Object Comparison Tool for the first time” on page 25</a> <a href="#">“Roadmap: Recustomizing Db2 Object Comparison Tool” on page 28</a>	
<b>Allocate libraries</b>		
Before you can use Db2 Object Comparison Tool, you must allocate the libraries to your ISPF session.	<a href="#">“Allocating libraries for Db2 Object Comparison Tool” on page 34</a>	
<b>Optional: Customize JCL</b>		
Customize the JCL that Object Comparison Tool uses to adhere to your installation standards.	<a href="#">“Customizing the JCL that Object Comparison Tool uses” on page 34</a>	
<b>Optional: Customize data set names</b>		
Align the Db2 Admin Tool data set names with your local data set naming conventions.	<a href="#">“Customizing data set names” on page 35</a>	
<b>Optional: Make Object Comparison Tool available from Db2 Administration Tool</b>		
When you customize Db2 Admin Tool, you can make Db2 Object Comparison Tool available from the main menu.	<a href="#">Making DB2I and IBM Db2 Object Comparison Tool for z/OS available from the Db2 Administration Tool main menu (IBM Db2 Administration Tool for z/OS 13.1.0)</a>	
<b>Optional: Enable product discovery</b>		
This step is strongly recommended if Db2 Administration Foundation is also installed.	<a href="#">“Enabling product discovery for Object Comparison Tool” on page 37</a>	

## Software requirements for Object Comparison Tool

Prior to beginning the customization process for Object Comparison Tool, ensure that your environment meets all software requirements.

Object Comparison Tool 13.1 requires the following software:

- One of the following supported versions of Db2 for z/OS:
  - Db2 13 (5698-DB2)
  - DB2® Value Unit Edition 13.1 (5698-DBV)
  - 5650-DB2
  - Db2 Value Unit Edition 12.1
- The requisite release of z/OS for the Db2 subsystems that you will be using with Object Comparison Tool
- IBM Db2 Administration Tool for z/OS 13.1 (5698-AT3).

**Memory recommendations:** Because Db2 Object Comparison Tool keeps information in memory for efficiency, use a minimum region of 256 MB of memory for both batch and TSO. Ideally, if allowed by your installation policy, set REGION=0M for batch jobs to allow for maximum below-the-bar storage and avoid reruns.

When 1000 or more objects are processed, additional region is recommended. More memory is also necessary if you suppress object dropping when generating the job, because object attributes are kept resident to process this option. If you are processing more than 10,000 objects, use a starting region of 256 MB. If LE storage failures occur, increase region parameters before assuming that a problem exists. Increase memory in 32 MB increments.

In all cases, ensure that the requested region size is not limited to a lower amount by the IEFUSI installation exit.

## Data sets used by Tools Customizer

Tools Customizer (TCz) uses the following data sets during the customization process:

Data set name	Description
SCCQEXEC	EXEC library for TCz
SCCQDENU	Metadata library for TCz
SCCQLOAD	Executable load module library for TCz
SCCQMENU	ISPF messages for TCz
SCCQPENU	ISPF panels for TCz
SCCQSAMP	Sample members for TCz
SCCQTENU	Table library for TCz

## Customizing Db2 Object Comparison Tool

After Db2 Object Comparison Tool is SMP/E installed, you can customize the configuration by running IBM Tools Customizer for z/OS (TCz).

Customization involves tasks such as allocating libraries, customizing your JCL and data set names, and defining product-specific parameters. For an overview of the entire process, see [“Customization checklist for Db2 Object Comparison Tool”](#) on page 23. For more information about TCz in general, see [IBM Tools Customizer for z/OS 1.1.0](#)

## Roadmap: Customizing Db2 Object Comparison Tool for the first time

When you install Db2 Object Comparison Tool for the first time, you must customize the configuration by using IBM Tools Customizer for z/OS (TCz).

Complete the steps in the following table to customize Object Comparison Tool for the first time.

**Tip:**

- For multiple-LPAR environments, determine your customization strategy first: [“Using Tools Customizer in a multiple-LPAR environment”](#) on page 29.
- For guidance on any input fields in TCz, position your cursor on the input field and press F1 (Help).

Table 1. Steps for customizing Object Comparison Tool for the first time

Step	Description	Instructions
Start Tools Customizer.	<ol style="list-style-type: none"> <li>1. Edit the CCQTCZ member in the <i>hlq.TCZ110.SCCQEXEC</i> data set.</li> <li>2. Locate TCZHLQ="&lt;TCz HLQ&gt;".</li> <li>3. Change "&lt;TCz HLQ&gt;" to the high-level qualifier of your TCz EXEC data set, as shown in the following example: <pre data-bbox="521 506 967 558">TCZHLQ="hlq.TCZ110"</pre> </li> <li>4. Save your changes.</li> <li>5. On the ISPF Command shell panel, issue the following command: <pre data-bbox="521 684 967 737">EX 'hlq.TCZ110.SCCQEXEC(CCQTCZ)'</pre> </li> </ol>	<ul style="list-style-type: none"> <li>• <a href="#">Starting Tools Customizer (IBM Tools Customizer for z/OS 1.1)</a></li> <li>• <a href="#">"Data sets used by Tools Customizer" on page 25</a></li> </ul>
Modify Tools Customizer settings.	<ol style="list-style-type: none"> <li>1. On the <b>IBM Tools Customizer for z/OS (CCQPHME)</b> panel, specify option 0 (<b>User settings for Tools Customizer</b>).</li> <li>2. Specify values for the following required sections: <ul style="list-style-type: none"> <li>• Customization library qualifier</li> <li>• Use Db2 group attach name</li> <li>• Metadata library</li> <li>• Discover output data set</li> <li>• Data store data set</li> <li>• User job card settings</li> </ul> </li> <li>3. Save your changes, and press Enter.</li> </ol>	<a href="#">Modifying Tools Customizer user settings (IBM Tools Customizer for z/OS 1.1)</a>
Specify the Object Comparison Tool metadata library.	<ol style="list-style-type: none"> <li>1. On the <b>IBM Tools Customizer for z/OS (CCQPHME)</b> panel, specify option 1 (<b>Customize a product</b>).</li> <li>2. On the <b>Specify the Product or Pack Metadata Library (CCQPHLQ)</b> panel, enter the following value in the <b>Product or pack metadata library</b> field, and press Enter: <pre data-bbox="521 1587 967 1640">DMTOOL.SGOCDENU</pre> </li> </ol>	<a href="#">Specifying the metadata library for the product or pack to customize (IBM Tools Customizer for z/OS 1.1)</a>

Table 1. Steps for customizing Object Comparison Tool for the first time (continued)

Step	Description	Instructions
Create Db2 entries.	<ol style="list-style-type: none"> <li>1. On the <b>Customizer Workplace (CCQPWRK)</b> panel, issue the ASSOCIATE primary command, and press Enter.</li> <li>2. On the <b>Associate DB2 Entry for Product (CCQPDAD)</b> panel, issue the CREATE primary command, and press Enter.</li> <li>3. On the <b>Create DB2 Entries (CCQPCDB)</b> panel, specify the information for the new Db2 entry, and press Enter.</li> <li>4. On the <b>Associate DB2 Entry for Product (CCQPDAD)</b> panel, issue the A line command against the new Db2 entry, and press Enter.</li> </ol> <p>Create new Db2 entries and associate them with Object Comparison Tool.</p>	<p><a href="#">Creating and associating DB2 entries (IBM Tools Customizer for z/OS 1.1)</a></p>
Define product parameters.	<ol style="list-style-type: none"> <li>1. On the <b>Customizer Workplace (CCQPWRK)</b> panel, specify the E line command against the <b>Product parameters</b> field.</li> <li>2. On the <b>Product Parameters: DB2 Object Comparison (CCQPPRD)</b> panel, specify your parameter values. Required parameters are indicated by an asterisk (*).</li> <li>3. Press Enter to save and exit.</li> </ol>	<p><a href="#">Defining product or component parameters (IBM Tools Customizer for z/OS 1.1)</a></p>
Generate the jobs.	<p>On the <b>Customizer Workplace (CCQPWRK)</b> panel, issue the G line command against the new Db2 entry, and press Enter.</p>	<p><a href="#">Generating customization jobs (IBM Tools Customizer for z/OS 1.1)</a></p>
Optional: Edit the jobs.	<p>Ensure that the GOCFB2VB job contains the correct ADB and GOC SAMP data sets. If not, edit this job to correct those values.</p>	<p><a href="#">“Editing the GOCFB2VB job” on page 30</a></p>
Submit the jobs.	<p>On the <b>Finish Product Customization (CCQPCST)</b> panel, issue the E line command against the <i>abCUSTxy</i> member.</p>	<p><a href="#">“Submitting the customization jobs” on page 32</a></p>
Propagate the customizations to additional LPARs as needed.	<p>If you have a multiple-LPAR environment, use one of the specified methods to propagate your customization to other LPARs.</p>	<p><a href="#">“Using Tools Customizer in a multiple-LPAR environment” on page 29</a></p>

## Roadmap: Recustomizing Db2 Object Comparison Tool

After you have initially customized Db2 Object Comparison Tool by using Tools Customizer (TCz), you might later need to recustomize it to change one or more parameter values. For example, when you apply maintenance, the instructions might direct you to recustomize Object Comparison Tool.

The new customization jobs will replace the customization jobs that were previously generated and stored in the customization library. Part of the recustomization process includes selecting or deselecting optional tasks or steps, changing the definitions of parameters, or both. Use the method in this roadmap instead of deleting customization jobs from the customization library.

To recustomize Object Comparison Tool, complete the steps in the following table.

### Tips:

- For multiple-LPAR environments, determine your customization strategy first: [“Using Tools Customizer in a multiple-LPAR environment”](#) on page 29.
- Use a new customization library every time that you apply maintenance and regenerate all the TCz jobs (by using the GENERATEALL command). For example, append a date as show in the following example:

```
Customization lib: RSTEST.AOC.$RS01$.ADB1210.D200716
```

This practice provides a backup and a way to compare the jobs to a previous customization by using ISPF option 3.12.

- For guidance on any input fields in TCz, position your cursor on the input field and press F1 (Help).

Table 2. Required steps for recustomizing Object Comparison Tool

Step	Description	Instructions
Start TCz.	<ol style="list-style-type: none"> <li>1. On the ISPF Command shell panel, issue the following command:   <pre>EX 'hlq.TCZ110.SCCQEXEC(CCQTCZ)'</pre> </li> </ol>	Starting Tools Customizer ( <a href="#">IBM Tools Customizer for z/OS 1.1</a> )
Specify the Object Comparison Tool metadata library.	<ol style="list-style-type: none"> <li>1. On the <b>IBM Tools Customizer for z/OS (CCQPHME)</b> panel, specify option 1 (<b>Customize a product</b>).</li> <li>2. On the <b>Specify the Product or Pack Metadata Library (CCQPHLQ)</b> panel, enter the following value in the <b>Product or pack metadata library</b> field, and press Enter:   <pre>DMTOOL.SGOCDENU</pre> </li> </ol>	Specifying the metadata library for the product or pack to customize ( <a href="#">IBM Tools Customizer for z/OS 1.1</a> )
Define product parameters.	<ol style="list-style-type: none"> <li>1. On the <b>Customizer Workplace (CCQPWRK)</b> panel, specify the E line command against the <b>Product parameters</b> field, and press Enter.</li> <li>2. Edit the specific tasks, steps, or parameters that you want to change.</li> <li>3. Press Enter to save and exit.</li> </ol>	Defining product or component parameters ( <a href="#">IBM Tools Customizer for z/OS 1.1</a> )
Generate the jobs	On the <b>Customizer Workplace (CCQPWRK)</b> panel, issue the G line command against a site-specific SSID, and press Enter.	Generating customization jobs ( <a href="#">IBM Tools Customizer for z/OS 1.1</a> )

Table 2. Required steps for recustomizing Object Comparison Tool (continued)

Step	Description	Instructions
Optional: Edit the jobs.	Ensure that the GOCFB2VB job contains the correct ADB and GOC SAMP data sets. If not, edit this job to correct those values.	<a href="#">“Editing the GOCFB2VB job” on page 30</a>
Submit the jobs.	On the <b>Finish Product Customization (CCQPCST)</b> panel, submit the generated jobs in the order they are displayed.	<a href="#">“Submitting the customization jobs” on page 32</a>
Propagate the customizations to additional LPARs as needed.	If you have a multiple-LPAR environment, use one of the specified methods to propagate your customization to other LPARs.	<a href="#">“Using Tools Customizer in a multiple-LPAR environment” on page 29</a>

## Using Tools Customizer in a multiple-LPAR environment

Tools Customizer (TCz) supports customizations on only the local LPAR. However, you can propagate customizations to additional LPARs.

### About this task

In a multiple-LPAR environment, TCz identifies the LPAR to which you are logged on and uses this LPAR name for several parameter settings, including the data store. Therefore, you can use the TCz data store to customize only that LPAR.

### Procedure

To customize products that run against Db2 subsystems on multiple LPARs, use one of the following methods:

- **Method 1: Customize a single Db2 subsystem or data sharing group and copy the customization jobs to each LPAR**

a) Customize one Db2 subsystem or member.

For example, you might customize member DB1S in group DBGS in your sandbox environment.

b) If you are using data sharing, propagate that customization to the other members in the group:

a. Copy the customization jobs to the other members.

For example, copy the jobs for DB1S to member DB2S.

b. Edit the jobs as needed for the subsystem and LPAR.

For example, replace the member names. Depending on your environment, you might also need to replace data set names. You can use a REXX exec to do this customization.

c. Run those jobs.

Some jobs do not need to be run on every member in a group. Some jobs only need to run once per LPAR or Sysplex. To determine where a job needs to be run, look at the job listings on the **Finish Product Customization (CCQPCST)** panel. Depending on the values of the **SSID** and **GrpAttch** columns, take the following actions for each job:

Table 3.			
SSID column value	GrpAttch column value	Action	Comments
--	--	Run once per LPAR	None
--	A group name	Run once per group	None

<i>Table 3. (continued)</i>			
<b>SSID column value</b>	<b>GrpAttch column value</b>	<b>Action</b>	<b>Comments</b>
A member name	A group name	Run once per member in the group	None
An SSID	--	Run once	This entry is for a stand-alone Db2 subsystem.

c) Copy the jobs from the initial customized subsystem or member to all your other subsystems or groups. Then, edit those jobs, preferably with a REXX exec, and run them.

For example, copy the jobs for DB1S in group DBGS to the members DB1D and DB2D in your development group DBGD, edit those jobs as needed, and run them. Then, copy the jobs for DB1S to the members DB1T and DB2T in your test group DBGT, edit those jobs, and run them. Continue until all groups are customized.

- **Method 2: Generate customization jobs for each Db2 subsystem and copy those jobs to the appropriate LPARs**

- Associate all Db2 entries in one instance of TCz on one LPAR, regardless of the LPARs on which the Db2 subsystem resides.
- Generate customization jobs for each Db2 entry.
- Copy the generated customization jobs to the LPAR to run against the specific Db2 entries. You might need to edit these customization jobs for specific LPARs. For example, you might need to edit the data set names. (Otherwise, you generally do not need to make manual changes to the jobs that are customized by TCz.)

## Editing the GOCFB2VB job

The GOCFB2VB job is generated by TCz if you specified that you wanted to create variable-blocked (VB) versions of the Db2 Admin Tool and Object Comparison Tool CLIST and EXEC libraries. You might need to edit this job to specify the correct ADB and GOC SAMP data sets.

### About this task

GOCFB2VB is generated based on the information specified on the **Product Parameters: DB2 Object Comparison (CCQPPRD)** panel under the following field:

#### Create Variable Block CLIST and EXEC libraries

If you use CLIST and EXEC libraries that are variable blocked (VB), create VB versions of these libraries. The data set names of the new VB libraries are the same as the fixed blocked (FB) libraries but are suffixed with .VB.

GOCFB2VB is based on the GOCFB2VB template and is in member *job\_sequence\_number\_FB2V\_Db2\_entry\_ID*.

The following parameters in GOCFB2VB correspond to the indicated field on **Product Parameters: DB2 Object Comparison (CCQPPRD)** panel:

<i>Table 4. GOCFB2VB parameters</i>	
<b>Parameter name</b>	<b>Field</b>
CCQ_GOC_HLQ	<b>DB2 Object Comparison hlq</b>
CCQ_GOC_FB2VB_VLSRNM	<b>Fixed to variable blocked VOLSER</b>
CCQ_GOC_FB2VB_DASD	<b>Fixed to variable blocked UNIT</b>
CCQ_GOC_ADB_HLQ	<b>DB2 Admin Tool hlq</b>

## Procedure

To edit the GOCFB2VB job:

1. Open the GOCFB2VB job in the ISPF editor.
2. Edit the job step that creates the VB version of the CLIST library. Check the low-level qualifier for the VB data set and correct it if needed.

For example, in the following job step, SGOCCCLST is specified as the low-level qualifier:

```
//*****  
//*  
//CLIST EXEC PGM=IKJEFT01,  
// PARM=('%ADBFVBV DMT00L.SGOCCCLST',  
// 'DMT00L.SGOCCCLST.VB')  
//SYSEXEC DD DISP=SHR,DSN=DMT00L.SADBSAMP  
//SYSTSPRT DD SYSOUT=*  
//SYSTSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSIN DD DISP=SHR,  
// DSN=DMT00L.SADBSAMP(ADBIEBVB)  
//MEMBERS DD *  
*  
//*
```

Figure 2. Example GOCFB2VB job step that creates the VB version of the CLIST library

This job creates the GOCC10.*low-level-qualifier*.VB data set, where *low-level qualifier* is the low-level qualifier that you specify.

3. Edit the job step that creates the VB version of the EXEC library. Check the low-level qualifier for the VB data set and correct it if needed.

For example, in the following job step, SGOCEXEC is specified as the low-level qualifier:

```
//*****  
//*  
//EXEC EXEC PGM=IKJEFT01,  
// PARM=('%ADBFVBV DMT00L.SGOCEXEC',  
// 'DMT00L.SGOCEXEC.VB')  
//SYSEXEC DD DISP=SHR,DSN=DMT00L.SADBSAMP  
//SYSTSPRT DD SYSOUT=*  
//SYSTSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSIN DD DISP=SHR,  
// DSN=DMT00L.SADBSAMP(ADBIEBVB)  
//MEMBERS DD *  
*  
//*
```

Figure 3. Example GOCFB2VB job step that creates the VB version of the EXEC library

This job creates the GOCC10.*low-level-qualifier*.VB data set, where *low-level qualifier* is the low-level qualifier that you specify.

4. Save the file.

## What to do next

Submit the job.

## Submitting the customization jobs

After TCz generates the customization jobs for Db2 Object Comparison Tool, you must submit them to complete the customization process. SYSADM or equivalent authority is required to run the generated jobs.

### About this task

TCz generates customization jobs based on the tasks and steps that you select. The following table shows the relationship between the tasks and steps that you select, and the member that contains the jobs that TCz generates.

Tasks	Steps	Template name	Template type
Create the VB CLIST and EXEC libraries.	Create the VB libraries.	GOCFB2VB	perhlq

The following figure shows part of the **Finish Product Customization (CCQPCST)** panel. The table on this panel shows the customization jobs that are generated by TCz. They are grouped by job sequence number.

```

CCQPCST          Finish Product Customization          Row 1 to 2 of 2
Command ==>>>          Scroll ==>>> PAGE

For a first-time customization, submit the jobs in the members in the order
in which they apply to the DB2 entries. Otherwise, submit only the necessary
jobs that were generated after changes were made. To submit jobs, browse
the members and issue the TSO SUBMIT
command.

Line Commands: E - Edit  B - Browse

Product customization library .: CCQTCZ.SYSADM.CUST.$3090$.GOC1020

Cmd Member  New SSID GrpAttch Template Date      Description
-----
  A0FB2VB   YES  --   --      GOCFB2VB 2013/01/10 Copy the FB libraries to the
VB
----- End of customized jobs -----

```

Figure 4. The **Finish Product Customization (CCQPCST)** panel

The member-naming conventions depend on whether the customization jobs are for Db2 entries, an LPAR, or the product, as follows:

#### Customization jobs for Db2 entries

The members use the following naming convention:

```
<job_sequence_number><job_ID><DB2_entry_ID>
```

where

##### **job\_sequence\_number**

Two alphanumeric characters, A0 - Z9, that TCz assigns to a customization job. The number for the first template in the sequence is A0, the number for the second template is A1, and so on.

##### **job\_ID**

Characters 4 - 7 of the template name, if the template name has five or more characters. Otherwise, only character 4 is used. Object Comparison Tool assigns the template name.

##### **DB2\_entry\_ID**

Two alphanumeric characters, AA - 99, that TCz assigns to a Db2 entry.

For example, the XYZBNDDDB2\_entry\_ID\_1 and XYZBNDDDB2\_entry\_ID\_2 jobs are generated from the XYZBNDGR template, and the XYZ4DB2\_entry\_ID\_1 and XYZ4DB2\_entry\_ID\_2 jobs are generated from the XYZ4 template. If the jobs are generated on two Db2 entries, the following member names are listed sequentially: A0BNDGAA, A0BNDGAB, A14AA, A14AB.

### Customization jobs for an LPAR or the product

The members use the following naming convention:

```
<job_sequence_number><job_ID>
```

where

#### **job\_sequence\_number**

Two alphanumeric characters, A0 - Z9, that TCz assigns to a customization job. The number for the first template in the sequence is A0, the number for the second template is A1, and so on.

#### **job\_ID**

Characters 4 - 8 of the template name, if the template name has five or more characters.

Otherwise, only character 4 is used. For example, for the XYZMAKE template, the job ID is MAKE.

For the XYZM template, the job ID is M. Object Comparison Tool assigns the template name, and it is displayed in the Template column.

For example, the XYZBNDGR job is generated from the XYZBNDGR template, and the XYZ4 job is generated from the XYZ4 template. The following member names are listed sequentially: A0BNDGR, A14.

Use the **New** column to determine whether the job member needs to be submitted:

#### **YES**

The job member is newly created or updated and needs to be submitted for customization.

#### **NO**

The job member is not newly created or updated and does not need to be submitted for customization.

### Procedure

Submit the generated customization jobs by following the process that you use in your environment or by using the following method:

1. Specify B or E against a customization job or the product customization library, and press Enter.  
An ISPF browsing or editing session is started.
2. Browse the customization job or each member in the library to ensure that the information is correct.
3. Run the TSO SUBMIT command.
4. Press End.

### Results

Object Comparison Tool is customized, and the **Customizer Workplace (CCQPWRK)** panel is displayed. For the Db2 entries on which Object Comparison Tool was customized, the status is Customized .

### What to do next

You can generate more customization jobs for other Db2 entries, view a list of customization jobs that you previously generated, or recustomize Object Comparison Tool.

## Allocating libraries for Db2 Object Comparison Tool

Before you can use Db2 Object Comparison Tool, you must first allocate the libraries to your ISPF session.

### Procedure

- To allocate the Object Comparison Tool libraries to your ISPF session, choose one of the following three methods that is most appropriate for your installation:
  - Use the PRODADD and LIBAPRE parameters on the ADBL CLIST to specify the unique library names for Object Comparison Tool libraries.

If you are using the ADBL CLIST to allocate the Db2 Administration Tool ISPF libraries (by using the LIBDEF service), you should also use the ADBL CLIST to allocate the Object Comparison Tool libraries, as shown in the following example:

```
TSO %ADBL PRODADD(GOCB10) LIBAPRE(SGOC)
```

- If your installation copied the Db2 Administration Tool ISPF libraries to a set of libraries that are allocated before you start ISPF, copy Object Comparison Tool ISPF libraries into these same libraries or allocate additional ISPF data sets.
  - If you have a personal set of ISPF libraries, copy the Object Comparison Tool ISPF libraries to these data sets. To verify that you have allocated (using LIBDEF) the correct ISPF libraries, you can use the ISPF command ISPLIBD. You can also use the TSO ATLIB DISPLAY command to verify the CLIST and EXEC library allocations.
- If you plan to run compare jobs online, also ensure that the Db2 libraries are set up properly.

A compare job can be run either in batch or online. Compare jobs that run online require access to the DSNHDECP module and access is available only if the Db2 libraries are set up properly. If the Db2 load library data set does not exist in the system LINKLIST, the data set must be added to the STEPLIB of the TSO logon procedure. If the Db2 load library data set does not exist in the system LINKLIST or in the STEPLIB, the following error is returned in the compare output when an online compare is run:

```
Unable to load DB2 DECP module: rc = 8. Compare function is terminated.
```

### Related concepts

[“ADBL CLIST for invoking Db2 Object Comparison Tool” on page 38](#)

The ADBL CLIST, in the SADBCLST library, is provided for running Db2 Admin Tool or Object Comparison Tool.

## Customizing the JCL that Object Comparison Tool uses

You might need to customize the Object Comparison Tool JCL to adhere to your installation standards. You can configure the JCL that is used by Object Comparison Tool to run Db2 utilities and other Db2 functions by modifying the skeletons in the SADBSLIB and SGOCSLIB libraries. Most other skeletons do not require configuration.

### Procedure

To customize the JCL that Object Comparison Tool uses:

- Configure the following members of SADBSLIB as needed:

#### **ADBAPY**

Generates an apply job or step (uses ISPF batch)

**Tip:** Because member ADBAPY uses ISPF batch for its generated apply job, its skeleton might require more extensive configuration than the other skeletons.

#### **ADBDCMD**

Executes Db2 commands

**ADBEDDL**

Executes DDL files (DROP, CREATE, ALTER)

**ADBTCHK**

Generates a CHECK DATA job or step

**ADBTHPU**

Generates a High Performance Unload job or step

**ADBTIMC**

Generates an image copy job or step

**ADBTREL**

Generates a LOAD or RELOAD job or step

**ADBTREO**

Generates a REORG job or step

**ADBTREN**

Generate a RUNSTATS job or step

**ADBTUNL**

Generates an UNLOAD job or step

**ADBS27AC**

Generates a convert job or step

- Configure the following members of SGOCSLIB as needed:

**GOCCMP**

Generates a compare job or step

**GOCDDB2**

Generates extractions from the Db2 catalog for the source or target

**GOCDL**

Generates extractions from the DDL for the source or target

## Customizing data set names

You can set up Db2 Admin Tool to use your local naming conventions for data sets.

### Procedure

To customize data set names, modify the ADB2UCUS skeleton that resides in the ISPSLIB library as follows:

- Edit any data set names that are preceded by SET statements as needed. (SET statements are indicated by )SET.)
- Use variables for the data set names as needed. A complete list of variables is included in the SLIB member ADB2UCUT. Some variables you can use are:

**&AJDATE**

Julian date (YYDDD)

**&AJDAY**

Julian day (DDD)

**&AYEAR4**

4-digit year (YYYY)

**&AGDATE**

Gregorian date (YYMMDD)

**&ANMON**

Numeric month (MM)

**&ADAY**

Day (DD)

**&AYEAR**

2-digit year (YY)

**&ACMON**

3-character month (XXX)

**&ATIME**

Time (HHMMSS)

**&ATIME7**

Time with tenths of seconds (HHMSST)

**&ATIME4**

Time without seconds (HHMM)

**&AHOUR**

Hour (HH)

**&AMIN**

Minute (MM)

**&ASEC**

Seconds (SS)

- Ensure that data set names do not extend beyond column 71 in the ADB2UCUS data set. Any characters beyond column 71 are truncated.
- Ensure that generated data set names, including periods, will not be longer than 44 bytes.

When you subsequently run SMP/E to receive and apply SMP/E usermod ADBU002, the updated ISPF JCL skeletons are added to the SADBSLIB library.

**Tip:** For testing purposes, copy the ADB2UCUS skeleton to a private skeleton library and make your changes. This private skeleton library must be allocated first in the ISPSLIB concatenation (using the USERADD parameter of the ADBL CLIST). After testing is complete, use an SMP/E USERMOD to update the Db2 Admin Tool product libraries. A sample SMP/E USERMOD is provided in member ADBU002 in the SADBSAMP library. Instructions for completing this step are provided in sample job ADBU002.

**Example**

This example demonstrates several different types of changes to the variable ASYCPY1.

The variable ASYCPY1 is shipped as follows:

```
)SET ASYCPY1 = &PREFIX..&DB2SYS..IC.&DBNAME..&NAME(+1)
```

To change the high-level qualifier from the current TSO PREFIX to MYHLQ, specify:

```
)SET ASYCPY1 = MYHLQ.&DB2SYS..IC.&DBNAME..&NAME(+1) /* CHANGE HLQ TO FIXED STRING
```

To change the second-level qualifier from the Db2 subsystem ID to TEST, specify:

```
)SET ASYCPY1 = &PREFIX..TEST.IC.&DBNAME..&NAME(+1) /* CHANGE SUBSYSTEM TO 'TEST'
```

To insert a high-level qualifier of MYHLQ in front of the current TSO PREFIX and to remove the Db2 database name, specify:

```
)SET ASYCPY1 = MYHLQ.&PREFIX..&DB2SYS..IC.&NAME(+1) /* CHANGE HLQ TO FIXED STRING,  
/* INCLUDE PREFIX, REMOVE DBNAME
```

To use sequential data sets rather than a GDG data set, specify a data set name that contains date and time values to generate unique data set names:

```
)SET ASYCPY1 = &PREFIX..IC.&DBNAME..&NAME..D&AJDATE..T&ATIME
```

## Enabling product discovery for Object Comparison Tool

*Product discovery* is the ability of one product to determine whether another product is installed without invoking that second product. For example, IBM Db2 Administration Foundation for z/OS can determine, or *discover*, whether Object Comparison Tool is installed without invoking the object compare options in Db2 Administration Tool.

Enabling product discovery for Object Comparison Tool is optional but strongly recommended if Db2 Administration Foundation is also installed. Enabling this product discovery allows you to use certain additional functions in Db2 Administration Foundation.

### Before you begin

The PTF for APAR [PH55178](#) must be installed.

### About this task

The following files are used for product discovery. Both files are provided as members of the SGOCSAMP data set and are in YAML format:

#### GOCDSCVP

The product file, which contains basic information about the product ID and release. Do not edit this file.

#### GOCDSCVS

The base for the customization file, which contains the names of the installation target library data sets and the location of the corresponding product file. You will edit this file to provide this information as part of the following procedure.

**Note:** Any future updates to GOCDSCVP or GOCDSCVS will be indicated in the ++HOLD information for the relevant PTF that updates the file. For example, GOCDSCVP might be updated if additional product features need to be discovered.

### Procedure

To enable product discovery for Object Comparison Tool:

1. Copy GOCDSCVP and GOCDSCVS in the SGOCSAMP library to another location where they can be accessed by products discovering Object Comparison Tool.

Copying these files ensures that maintenance to the files does not result in unexpected updates to the execution environment.

You can copy these files to sequential data sets, members of a PDS or PDSE, or UNIX System Services files. Name these files to conform to your installation naming standards, but also consider the following recommendations.

#### Naming recommendations:

- If the files are stored in sequential data sets, use the low-level qualifiers PROD (product file) and CUST (customization file).

#### Example:

DB2T00LS.GOC131.PROD — a copy of the GOCDSCVP product file  
DB2T00LS.GOC131.CUST — a customized version of the GOCDSCVS file

- If the files are stored in a PDS or PDSE, base the member names on the Object Comparison Tool release number and the type of file. Including the release in the member name allows you to retain files for multiple versions if your installation has multiple versions in use.

#### Example:

DB2T00LS.GOC.DISCOVER(GOC131P) — a copy of the GOCDSCVP product file  
DB2T00LS.GOC.DISCOVER(GOC131C) — a customized version of the GOCDSCVS file

- If the files are stored in the UNIX System Services file system, use the following paths:

- <optional prefix>/usr/lpp/db2tools/goc/prod/goc131P.yaml* – a copy of the GOCDSCVP product file
- <optional prefix>/usr/lpp/db2tools/goc/site/goc131C.yaml* – a customized version of the GOCDSCVS file
2. Edit the copy of GOCDSCVS as described in the file comments.
  3. If Zowe-based products that use IBMUnified Management Server for z/OS (UMS), such as Db2 Administration Foundation, will be used with Object Comparison Tool, update the UMS parameters to indicate the location of the product and customization files. For details, see [Unified Management Server for z/OS 1.2 documentation](#).

## ADBL CLIST for invoking Db2 Object Comparison Tool

---

The ADBL CLIST, in the SADBCLST library, is provided for running Db2 Admin Tool or Object Comparison Tool.

The ADBL CLIST opens the Db2 Admin Tool main menu. Use the PANEL(GOCMENU) parameter to instead display the Db2 Object Comparison Tool main menu.

You can start the ADBL CLIST from any ISPF panel or from the ISPF command processor panel (usually ISPF option 6). You can add the % prefix to the beginning of the CLIST name to ensure that TSO/E searches only the CLIST libraries.

### **Related information**

[Invoking Db2 Admin Tool \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

# Chapter 3. Getting started with Db2 Object Comparison Tool

Object Comparison Tool runs as an extension to Db2 Admin Tool. The ISPF full-screen interface uses Db2 Admin Tool functions to display panels and run SQL statements.

As part of the panel interface, Object Comparison Tool provides a walk-through option that leads you through the process of creating a job to compare Db2 objects. This end-to-end framework guides you through the options that you need to specify. ISPF help panels are also available in Object Comparison Tool. To display a help panel, enter HELP or press PF1.

## Opening Object Comparison Tool

You invoke Object Comparison Tool from the main menu in Db2 Admin Tool.

### Procedure

On the **DB2 Administration Menu (ADB2)** panel, specify option C, and press Enter:

```
ADB2 dmin ----- DB2 Administration Menu 13.1.0 ----- 17:50
Option ==> C

  1 - DB2 system catalog           DB2 System: DD1A
  2 - Execute SQL statements       DB2 SQL ID: ADM001
  3 - DB2 performance queries     Userid   : ADM001
  4 - Change current SQL ID       DB2 Schema: ADM001
  5 - Utility generation using LISTDEFS and TEMPLATES DB2 Rel   : 1315
  P - Change DB2 Admin parameters DB2 F.Lvl : V13R1M501
  DD - Distributed DB2 systems     Max ApplC : V13R1M500
  E - Explain                     ApplCompat: V13R1M500
  Z - DB2 system administration   Cat Level : V13R1M501
  SM - Space management functions
  W - Manage work statement lists
  X - Exit DB2 Admin
  CC - DB2 catalog copy version maintenance
  CM - Change management

Interface to other DB2 products and offerings:
  I DB2I
  C DB2 OBJECT COMPARISON TOOL

More:      +
```

Figure 5. **DB2 Administration Menu (ADB2)** panel

The **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed:

```

Compare ----- DB2 Object Comparison Tool Menu ----- 09:38
Option ==>
                                     Specification Status:

1 - Specify compare source (new)      Specification Status:
2 - Specify compare target (old)      Incomplete
3 - Specify compare masks             Incomplete
4 - Specify ignores                   None specified
5 - Generate compare job              Using defaults
                                     Not generated

W - Walk through steps 1 - 5 in sequence
V - Generate job to extract version file from source only

R - Reset all
RS - Reset source
RT - Reset target

S - Save dialog
M - Manage/Restore dialog
MC - MultiCompare
MR - Manage saved compare results

```

Figure 6. **DB2 Object Comparison Tool Menu (GOCMENU)** panel

### Related reference

“Object Comparison Tool main menu” on page 40

Use the **DB2 Object Comparison Tool Menu (GOCMENU)** panel to specify the criteria for the comparison that you want to run.

## Object Comparison Tool main menu

Use the **DB2 Object Comparison Tool Menu (GOCMENU)** panel to specify the criteria for the comparison that you want to run.

```

GOCMENU ----- DB2 Object Comparison Tool Menu 13.1.0 ----- 10:05
Option ==>
                                     Specification Status:

1 - Specify compare source (new)      Incomplete
2 - Specify compare target (old)      Incomplete
3 - Specify compare masks             None specified
4 - Specify ignores                   Using defaults
5 - Generate compare job              Not generated

W - Walk through steps 1 - 5 in sequence
V - Generate job to extract version file from source only

R - Reset all
RS - Reset Source
RT - Reset Target

S - Save dialog
M - Manage/Restore dialog
MC - MultiCompare
MR - Manage saved compare results

```

Figure 7. **DB2 Object Comparison Tool Menu (GOCMENU)** panel

This panel has the following options:

#### 1 - Specify compare source (new)

Select this option to begin specifying the Db2 source objects to be compared. For detailed instructions, see [“1. Specifying source objects” on page 56](#).

#### 2 - Specify compare target (old)

Select this option to begin specifying the Db2 target objects to be compared. For detailed instructions, see [“2. Specifying target objects” on page 68](#).

### **3 - Specify compare masks**

Select this option to specify that names and qualifiers are to be translated by using masks before the comparison is performed. For detailed instructions, see [“3. Specifying compare masks”](#) on page 71.

### **4 - Specify fields to ignore**

Select this option to specify that certain fields should be ignored when the comparison is performed. For detailed instructions, see [“4. Specifying ignores”](#) on page 80.

### **5 - Generate compare job**

Select this option to generate the batch compare job. For detailed instructions, see [“5. Generating a compare job”](#) on page 84.

### **W - Walk through steps 1 – 5 in sequence**

Select this option to proceed directly to each step in succession without returning to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel and selecting them individually.

### **V - Generate job to extract version file from source only**

Generates a batch job that creates a version file from the source only. This version file can be used for the source or target in subsequent compare operations. This option can be used to create a version file on one system, transfer the version file to another system, and then generate a compare job on the other system.

### **R - Reset all**

Clears the **Specification Status** fields for all options. You can then enter new specifications for each option.

### **RS - Reset Source**

Clears the **Specification Status** field for option 1. You can then specify a new source.

### **RT - Reset Target**

Clears the **Specification Status** field for option 2. You can then specify a new target.

### **S - Save dialog**

Stores the current selections for later retrieval and subsequent reuse. For information about how to save a dialog, see [“Saving dialogs”](#) on page 160.

### **M - Manage/Restore dialog**

Select this option to retrieve, rename, or delete a previously saved dialog. For information about how to use these functions, see [Chapter 10, “Managing and restoring dialogs,”](#) on page 159.

### **MC - MultiCompare**

Select this option to compare one or more saved dialogs. For information about comparing multiple objects, see [Chapter 12, “Comparing multiple sources and targets,”](#) on page 165.

### **MR - Manage saved compare results**

Select this option to manage and view the saved compare results.

## **Db2 Object Comparison Tool scenarios**

---

You can use Db2 Object Comparison Tool to compare development objects with production objects. If needed, you can also use Object Comparison Tool to undo any resulting changes. You can also use this tool to compare DDL to objects in the catalog, to copy objects, and to convert table spaces to UTS. These operations are illustrated in the following scenarios.

### **Scenario: Comparing a Db2 development catalog to a Db2 production catalog**

When you make changes on your development system, such as creating a new table or view or changing an existing table, you might want to eventually make those same changes on your production system. To do so, you can use Db2 Object Comparison Tool to compare your development catalog to your production catalog. Then, Object Comparison Tool can make changes in the production catalog so that the objects in both systems are the same.

## Procedure

To compare a Db2 development catalog to a Db2 production catalog:

1. Specify the source (your development catalog):

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **1 - Specify compare source (new)**, and press Enter.

**Tip:** Issue the PANELID command so that you can see the name of the panel in the upper left corner.

- b) On the **Specify Compare Source (GOC1)** panel, select option **2 - Source is from the DB2 catalog**, and press Enter.

- c) On the **Specify DB2 Source Catalog Extract (GOC12)** panel, complete the following fields:

### Data set name

The name of the data set that you want to use for the version file for the source, such as devdb.v23.D080319. (Version files are created as part of the compare process. These files store information about the objects to be compared.)

### Tips:

- Save all your version files for future comparisons and the ability to undo changes at a later time if needed.
- Plan a naming convention to help keep track of the version files and easily find them. One possible naming convention is to include the date, as in the preceding example (D080319).

### Description

A description of the source, such as development database.

**Tip:** For this scenario, the description is simple. When you are doing your own comparisons, assign descriptive names to your version files so that you can easily find them, and include the date that they were created. For example:

```
Accounting V9 R10 M08 2019-04-01
```

- d) Select option **1 - Source is databases from the DB2 catalog**, and press Enter.

- e) On the **Specify Source DB2 Databases (GOC1D)** panel, use the I line command to insert a line.

- f) On the **Compare Add Databases (GOC1DA)** panel, specify the source database or databases by entering a partial database name and pressing Enter.

For example, if you enter AGBL in the **Partial database name** field, all databases that begin with AGBL are displayed.

- g) On the **Compare Add Databases (GOC1DD)** panel, use the S line command to select the specific databases that you want to compare.

- h) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that the **Specification Status:** next to option **1** is now **Database extract specified**.

2. Specify the target (your production catalog):

- a) Select option **2 - Specify compare target (old)**, and press Enter.

- b) On the **Specify Compare Target (GOC1)** panel, select option **2 - Target is from the DB2 catalog**, and press Enter.

- c) On the **Specify DB2 Target Catalog Extract (GOC12)** panel, complete the following fields:

### Data set name

The name of the data set to use for the version file for the target, such as proddb.v23.D080311.

### Description

A description of the target, such as production database scenario.

- d) Select option **1 - Target is databases from the DB2 catalog**, and press Enter.

- e) On the **Specify Target DB2 Databases (GOC1D)** panel, use the I line command to insert a line.

- f) On the **Compare Add Databases (GOC1DA)** panel, specify the database that contains the target by entering a partial database name (such as DGWD) and a location name (such as STLEC1), and press Enter.
- g) On the **Compare Add Databases (GOC1DD)** panel, use the **S** line command to select the target database or databases that you want to compare with the source.
- h) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that the **Specification Status:** next to option **2** is now **Database extract specified.**

3. Specify any compare masks:

Often, the names of objects in your development system are not the same as the names in your production system. Even if the names are the same, the owner IDs might be different. You can use compare masks to account for these differences. Db2 Object Comparison Tool can then match the appropriate objects for the comparison, even if the names are different.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **3 - Specify Compare Masks**, and press Enter.
- b) On the **Specify Compare Masks (GOC3)** panel, complete the following fields, and press Enter:

**Mask DSN**

The name of the data set for the masks. If the data set does not exist, it is created.

**Edit Mask**

YES

This scenario shows you how to define masks in a data set. Alternatively, if Change Management is enabled, you can define masks in the Change Management repository.

- c) On the **Edit Compare Masks (GOCEDIT)** panel, insert a line for each mask.

For example, the following lines define name masks:

```
OWNER: ABC*, DEF*
DBNAME: *TDB, *PDBA
TSNAME: T*T, P*P
TBNAME: T*, P*
```

For information about mask definitions and syntax, see [“Translation masks” on page 73](#).

For example, `OWNER: ABC*, DEF*` specifies that all owner names of `ABC*` in the source are translated to `DEF*` for the comparison. (The asterisk is a wildcard.) In this case, `ABCDDBA` in the source matches `DEFDBA` in the target.

**Tip:** Usually, the compare process is iterative. You generate a compare job and then analyze the differences in the report to see what masks you need to create for the next run of the compare job.

- d) Issue the SAVE command
- e) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that the **Specification Status:** next to option **3** is now **Mask specified.**

The next step that is listed on the **DB2 Object Comparison Tool Menu (GOCMENU)** panel is to specify ignore fields (option **4 - Specify ignores**). Ignore fields are characteristics that you want to be ignored during the comparison. For example, different buffer pool names in the source and target might be acceptable, and you do not want this difference to result in a change. At this point, assume that you do not know of any such differences that you want to ignore. So skip this step for now.

4. Generate and run a compare job:

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **5 - Generate compare job**, and press Enter.
- b) On the **Generate Compare Jobs (GOC5)** panel, specify the following options, and press Enter:

**Worklist name**

TEST

**Suppress DROP of objects**

YES

**PDS for jobs**

TEST

**Prefix for data sets**

TEST

**Single compare job**

YES

**Member name**

COMPARE

Set all the remaining options to NO, N, or blank.

Notice that for this first comparison, you are not requesting that any apply jobs be generated. Typically, you want to look at the comparison report first and make any changes before you generate apply jobs.

**Tip:** The settings for parameters are persistent; they settings that you specified previously remain until you change them.

- c) In the generated JCL job that is displayed, make any changes to the JCL as needed. For example, you might need to change the JOB statement.
  - d) Type the SUB command, and press Enter to submit the job.
  - e) Check that the job completed successfully.
5. Check the report to see the differences between the source and target:
- a) In the job output, look at the information under the line OBJECT COMPARISON REPORT.  
This report shows the differences between the source and target objects. It lists the differences as changes that need to be made to the target so that it matches the source.

For this scenario, suppose that you notice the following items in the output:

```
Compare tablespace source (AGBLTDB.TBMT001T) and target (AGBLPDB.PBMT001P)
  (A)Field CLOSE changed from NO to YES
  (A)Field PRIQTY changed from 192 to 48
  (A)Field USING changed from 'STOGROUP AGBLPSG' TO 'STOGROUP AGBLTSG'
Tablespace will be altered
```

The CLOSE attribute, PRIQTY attribute, and STOGROUP name are all listed as changed. The preceding lines in the output mean that the values are different in the source and target. However, in this case, suppose that you do not want to change the name of STOGROUP or the values of the CLOSE and PRIQTY attributes. Therefore, you need to set a mask for STOGROUP and ignore fields for CLOSE and PRIQTY and then run a comparison job again.

Suppose that you also notice in the report that objects are altered, dropped, and added:

```
Tablespace AGBLPDB.PBMT037P not found on source
Tablespace AGBLPDB.PBMT037P will be dropped
```

```
Tablespace AGBLPDB.PBMT0009P not found on target
Tablespace AGBLPDB.PBMT0009P will be added
```

```
Compare table source(DBA128.TBMT001_S_M_WORK) and target
(DBA128.PBMT001_S_M_WORK)
  (A)Add primary key : CD_USER(CD_USER,NO_SEQ)
Tables have identical column lists
Table will be altered
```

For this scenario, assume that these changes are changes that you want to make to your production system.

6. Add the additional mask and ignore fields.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **3 - Specify Compare Masks**, and press Enter.

The **Specify Compare Masks (GOC3)** panel should list the same data set name that you originally specified (in step [“3”](#) on page 43).

- b) Make sure the **Edit Mask** field is still set to YES, and press Enter.

- c) On the **Edit Compare Masks (GOCEDIT)** panel, add SGNAM: \*TSG, \*PSG.

(This mask accounts for the difference in the STOGROUP names AGLPSG and AGLTSG.)

- d) Exit back to **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **4 - Specify ignores**, and press Enter.

- e) On the **Specify Compare Ignores (GOC4)** panel, specify the following values, and press Enter:

**Data Set Name**

The name of a data set for the ignore file, such as IGNORE . DATA

**Edit Ignore Fields Specification**

YES

- f) On the **Specify Ignore Fields : Objects (GOCCI)** panel, use the **U** line command to update the SYSTABLESPACE object, and press Enter.

- g) In the **Specify Ignore Fields for object (GOCCIF)** panel, use the **S** line command to select CLOSERULE and PQTY.

Ignore fields are specified according to columns in the Db2 catalog. In this case, you want to ignore the CLOSE and PRIQTY attributes of the table space. Those values are captured in the CLOSERULE and PQTY columns of SYSIBM.SYSTABLESPACE. Therefore, CLOSERULE and PQTY need to be selected.

- h) Press PF3 to exit.

On the **Specify Ignore Fields : Objects (GOCCI)** panel, CLOSERULE and PQTY are listed in the **Ignore Fields** column for SYSTABLESPACE.

- i) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel.

Notice that the **Specification Status:** next to option **4** is now **Ignore fields specified**.

7. Generate another compare job with the new mask and ignore fields:

Db2 Object Comparison Tool created version files during the first comparison operation. You can now use these version files instead of choosing the objects from the catalog. Because version files are compressed to save space, using them can save you both time and CPU.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **1 - Specify compare source (new)**, and press Enter.

- b) This time, on the **Specify Compare Source (GOC1)** panel, specify option **3 - Source is from a compare version file**, and press Enter.

- c) On the **Specify Source Compare Version File (GOC13)** panel, specify the name of the data set that contains the version file for the source, and press Enter.

This data set name is the one you specified in step [“1”](#) on page 42.

- d) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **2 - Specify compare target (old)**, and press Enter.

- e) This time, on the **Specify Compare Target (GOC1)** panel, select option **3 - Target is from a compare version file**, and press Enter.

- f) On the **Specify Target Compare Version File (GOC13)** panel, specify the name of the data set that contains the version file for the target, and press Enter.

This data set name is the one you specified in step [“2”](#) on page 42.

After you press Enter, notice that on the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, the **Specification Status:** next to options **1** and **2** is now **Compare version file specified**.

- g) Select option **5 - Generate compare job**, press Enter, and complete the steps that you did before to generate and run the job.
  - h) Check the report output.  
You should see the mask and ignore fields that you specified.
8. Apply the changes to synchronize your production and development systems:
- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **5 - Generate compare job** again, and press Enter.
  - b) This time, on the **Generate Compare Jobs (GOC5)** panel, set the **Generate Apply Job** field to Yes and set any other fields as needed, and press Enter.
  - c) If the **Change Management Prompt (ADB2CMRO)** panel opens, specify NO.  
(For this scenario, assume that you are not using Change Management.)
  - d) On the **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel, enter the name of a data set where you want the apply jobs generated.
  - e) Edit the generated comparison job as needed, and submit the job.
  - f) Check the output to confirm that the job completed successfully.
  - g) Run the generated apply job to make the changes to your production catalog.

## Scenario: Undoing changes that were made in a catalog-to-catalog comparison

Suppose that you used Object Comparison Tool to compare two Db2 catalogs and then apply changes to the target catalog so that it matches the source catalog. Later, you decide that you do not want those changes. Db2 Object Comparison Tool can undo those changes for you.

### Before you begin

To undo the changes, you need the version files from the catalog-to-catalog comparison.

### About this task

Assume that you want to undo the changes that you made in [“Scenario: Comparing a Db2 development catalog to a Db2 production catalog”](#) on page 41 and restore the target (the production catalog) to its state before the comparison.

**Important:** Any data that is added between the time that the compare synchronization is done (step [“3”](#) on page 47) and the time that the undo changes process is done (step [“4”](#) on page 47) might be lost.

### Procedure

To undo changes that were made in a catalog-to-catalog comparison:

1. Specify the compare source.

In this scenario, the source is the version file for the target in the original comparison. This version file represents the production catalog before the changes.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **1 - Specify compare source (new)**.
- b) On the **Specify Compare Source (GOC1)** panel, specify option **3 - Source is from a compare version file**.
- c) On the **Specify Source Compare Version File (GOC13)** panel, specify the name of the data set that contains the version file, and press Enter.

(This data set name is the name that you specified in step [“2”](#) on page 42 in [“Scenario: Comparing a Db2 development catalog to a Db2 production catalog”](#) on page 41.)

On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, notice that the **Specification Status:** next to option **1** is now **Compare version file specified**.

2. Specify the compare target.

In this scenario, you want Object Comparison Tool to determine the target objects from the current production catalog based on the source version file.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **2 - Specify compare target (old)**.
- b) On the **Specify Compare Target (GOC1)** panel, select option **4 - Target is from the DB2 catalog and objects are automatically selected based on the selected source objects**.
- c) On the **Specify Target DB2 Location (GOC14)** panel, enter the following information, and press Enter:
  - The location of your production subsystem.
  - The name of a data set to use for the target version file. If the data set does not already exist, it is created.

On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, notice that the **Specification Status:** next to option **2** is now **Automatic (DB2 catalog extract)**.

For this scenario, do not specify any masks or ignore fields.

3. Generate the compare job:

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **5 - Generate compare job**.
- b) On the **Generate Compare Jobs (GOC5)** panel, specify the following options, and press Enter:

**Worklist name**

TEST

**Suppress DROP of objects**

YES

**PDS for jobs**

TEST

**Prefix for data sets**

TEST

**Single compare job**

YES

**Member name**

COMPARE

Set all the remaining options to NO or N.

- c) Edit the generated compare job as needed, and submit the job.
  - d) Check the output to confirm that the job completed successfully.
  - e) Check the compare report to make sure that the expected changes are listed.
4. Generate the apply job to undo the changes that you made previously in the catalog-to-catalog comparison:

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **5 - Generate compare job**.
- b) On the **Generate Compare Jobs (GOC5)** panel, specify the following additional options, and press Enter:

**Generate apply jobs**

YES

**Generate one job**

YES

**Member prefix**

APPLY

**Content of apply job(s)**

ALL

**Unload method**

U

**IDENTITY START value**

ORIGINAL

**Run REORG/REBUILD**

A

- c) If the **Change Management Prompt (ADB2CMRO)** panel opens, specify NO.  
(For this scenario, assume that you are not using Change Management.)
- d) On the **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel, enter the name of the data set where you want the apply job generated.
- e) Edit the generated compare job as needed, and submit the job.
- f) Check the output to confirm that the job completed successfully.
- g) Run the generated apply job to restore the production catalog to the state before the changes were made.

## Scenario: Comparing DDL to a catalog

You can compare the DDL for a single object to the Db2 catalog to make changes on the system for only that object.

### About this task

Suppose you want to change a table on your test system. For example, you might add a column in the middle or at the end of the table. So, you generate DDL that shows how the table will look after the change. The DDL will be your source for the comparison. It includes only a CREATE TABLE statement for the table. The DDL does not include any related indexes, foreign keys, or other related objects. Those objects will not be changed, because they are not included in the source DDL.

You then specify that the compare target be selected automatically from the Db2 catalog. In this case, Db2 Object Comparison Tool determines how to change the table. If the table does not currently exist in the target, Object Comparison Tool creates the table. If the table exists, Object Comparison Tool uses the version file instead of the catalog. Object Comparison Tool might alter the table or drop and re-create it, depending on the changes that need to be made. Db2 Object Comparison Tool restores objects and dependencies, such as indexes. If the table needs to be dropped and re-created, Db2 Object Comparison Tool also re-creates objects that have been dropped as a result of dropping the table. The table data is unloaded and, after the object definitions are applied, reloaded back into the table.

### Procedure

To compare DDL to a catalog:

1. Specify the compare source.

In this scenario, the source is the table definition in the DDL.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **1 - Specify compare source (new)**, and press Enter.
- b) On the **Specify Compare Source (GOC1)** panel, select option **1 - Source is from a DDL file**, and press Enter.
- c) On the **Specify Source DDL File (GOC11)** panel, specify the following information, and press Enter:
  - The name of the data set that contains the DDL.
  - The name of a data set to use for the source version file.

On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, notice that the **Specification Status:** next to option **1** is now **DDL file specified**.

2. Specify the compare target.

In this scenario, you want Object Comparison Tool to determine the target objects from the Db2 catalog based on the source.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **2 - Specify compare target (old)**.
- b) On the **Specify Compare Target (GOC1)** panel, select option **4 - Target is from the DB2 catalog and objects are automatically selected**.
- c) On the **Specify Target DB2 Location (GOC14)** panel, enter the following information, and press Enter:
  - The location of your subsystem.
  - The name of a data set to use for the target version file. If the data set does not already exist, it is created.

On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, notice that the **Specification Status:** next to option **2** is now **Automatic (DB2 catalog extract)**.

For this scenario, do not specify any masks or ignore fields.

3. Generate the compare job:

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **5 - Generate compare job**.
- b) On the **Generate Compare Jobs (GOC5)** panel, specify the following options, and press Enter:

**Worklist name**

TEST

**Suppress DROP of objects**

YES

**PDS for jobs**

TEST

**Prefix for data sets**

TEST

**Single compare job**

YES

**Member name**

COMPARE

Set all the remaining options to NO, N, or blank.

4. Edit the generated compare job as needed, and submit the job.
5. Check the output to confirm that the job completed successfully.
6. If needed, make any necessary corrections, generate the compare job again, and recheck the comparison report.
7. Apply the changes to the target table:

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **5 - Generate compare job**.
- b) On the **Generate Compare Jobs (GOC5)** panel, specify the following additional options, and press Enter:

**Generate apply jobs**

YES

**Generate one job**

YES

**Member prefix**

APPLY

**Content of apply job(s)**

ALL

**Unload method**

U

**IDENTITY START value**

ORIGINAL

**Run REORG/REBUILD**

A

- c) If the **Change Management Prompt (ADB2CMRO)** panel opens, specify NO.  
(For this scenario, assume that you are not using Change Management.)
- d) On the **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel, enter the name of the data set where you want the apply job generated.
- e) Edit the generated compare job as needed, and submit the job.
- f) Check the output to confirm that the job completed successfully.
- g) Run the generated apply job to change the target table.

## Scenario: Copying objects

You can use Db2 Object Comparison Tool to copy objects. For example, you might want to copy objects in your production environment to a test environment.

### About this task

For this scenario, assume that you created a database, PRODDb, in your production environment, and you want to copy the objects in that database to your test environment, in database TESTDB.

### Procedure

To copy PRODDb objects to TESTDB, on a different subsystem:

1. Specify the source object definitions to be compared:
  - a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select **Option 1 Specify compare source (new)**, and press Enter.
  - b) On the **Specify Compare Source (GOC1)** panel, select **2 - Source is from the DB2 catalog**, and press Enter.
  - c) On the **Specify DB2 Source Catalog Extract (GOC12)** panel, specify the following information:
    - In the **Data set name** field, enter the data set name for the version file.  
**Tip:** Include the date as part of the name. For example: proddb.v23.D080311
    - In the **Description** field, enter a description of the source. For example: production database
  - d) Select **1 - Source is databases from the DB2 catalog**, and press Enter.
  - e) On the **Specify Source DB2 Databases (GOC1D)** panel, specify the **I** line command to insert a database to the list.
  - f) On the **Compare Add Databases (GOC1DA)** panel, specify a partial data set name and the location name (for example STLEC1) to identify the data set that you want to copy, and press Enter.
  - g) On the **Compare Add Databases (GOC1DD)** panel, use the **S** line command to select the database that you want to copy.
  - h) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that the **Specification Status:** next to option **1** is now **Database extract specified**.
2. Specify the target objects:
  - a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select **Option 2 Specify compare target (old)**, and press Enter.

- b) On the **Specify Compare Target (GOC1)** panel, select **4 Target is from the DB2 catalog and the objects are automatically selected**, and press Enter.

In this scenario, choose automatic selection, because these objects might already exist in the target.

- c) On the **Specify Target DB2 Location (GOC14)** panel, enter the following information, and press Enter:

- The location of your test subsystem.
- The name of a data set to use for the target version file. If the data set does not already exist, it is created.

On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, notice that the **Specification Status:** next to option **2** is now **Automatic (DB2 catalog extract)**.

### 3. Specify the masks:

If naming differences exist between objects in the test database and objects in the production database, use masks to account for these naming differences. For example, owner, table name, or table space names might be different.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select **Option 3, Specify Compare Masks**, and press Enter.
- b) On the **Specify Compare Masks (GOC3)** panel, complete the following fields, and press Enter:

**Mask DSN**

The name of the data set for the masks. If the data set does not exist, it is created.

**Edit Mask**

YES

This scenario shows you how to define masks in a data set. Alternatively, if Change Management is enabled, you can define masks in the Change Management repository.

- c) On the **Edit Compare Masks (GOCEDIT)** panel, specify the masks that you want to use.

When you specify masks, make sure that the first value is the name in the production database and the second value is the name that you want used in the test database. For example:

```
Keyword: name in Source (production) ,name in Target (test)
```

Also consider that you might want to overwrite some values, such as the COMPRESS attribute.

For help in defining masks and overwriting values, see [“Translation masks” on page 73](#) and [Mask definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

- d) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that the **Specification Status:** next to option **3** is now **Mask specified**.

### 4. Specify fields to ignore:

You probably do not want to build test objects exactly the same as production objects. For example, you might want to ignore fields for buffer pools, PRIQTY, or SECQTY.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select **4 - Specify fields to ignore**.
- b) On the **Specify Compare Ignores (GOC4)** panel, complete the following fields, and press Enter:

**Data Set Name**

The name of the data set for the ignores. If the data set does not exist, it is created.

**Edit Ignore Fields Specification**

YES

This scenario shows you how to define ignores in a data set. Alternatively, if Change Management is enabled, you can define ignores in the Change Management repository.

- c) On the **Specify Ignore Fields : Objects (GOCCI)** panel, specify the **U** line command for SYSTABLESPACE.

- d) On the **Specify Ignore Fields for object (GOCCIF)** panel, specify the **S** line command for BPOOL, PQTY, and SECQTYI. Press Enter after each selection.  
The **Action** column indicates that the field is selected.
  - e) Press PF3 to return to the **Specify Ignore Fields : Objects (GOCCE)** panel. Notice that the **Ignore Fields** column for SYSTABLESPACE lists BPOOL, PQTY, SECQTYI.
  - f) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that the **Specification Status:** next to option 4 is now **Ignore fields specified**.
5. Generate compare jobs:
    - a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select **5 - Generate compare job**, and press Enter.
    - b) On the **Generate Compare Jobs (GOC5)** panel, specify the following values, and press Enter:
      - Worklist name:** TEST
      - Scope Warning Messages:** YES
      - PDF for batch jobs:** TEST
      - Prefix for data sets:** TEST
      - Generate one job:** YES
 Set all the remaining options to NO or N or the default.
    - c) If the **Change Management Prompt (ADB2CMRO)** panel opens, specify NO.  
(For this scenario, assume that you are not using Change Management.)
    - d) On the **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel, enter the name of a data set where you want the apply jobs generated.
    - e) Edit the generated JCL job as needed and submit it to run the comparison.
    - f) Check that the job completed successfully.
  6. Check the comparison report.  
In the job output, look at the information under the line OBJECT COMPARISON REPORT. (For help in evaluating the output, see [“Scenario: Comparing a Db2 development catalog to a Db2 production catalog”](#) on page 41.)
  7. Correct any problems with the job by changing the masks and ignore fields. Then, regenerate and re-run the comparison job. Repeat this process until the object comparison report contains the differences that you want apply to the target.
  8. Regenerate the comparison job and an apply job:
    - a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select **5 - Generate compare job**, and press Enter.
    - b) On the **Generate Compare Jobs (GOC5)** panel, set the **Generate apply jobs** field to Yes, and press Enter.  
When you generate the apply job, if you are modeling a complete set of new objects based on the original objects, data is not loaded or unloaded. No objects are dropped or altered. You are creating the objects, but not populating any data.
    - c) If the **Change Management Prompt (ADB2CMRO)** panel opens, specify NO.  
(For this scenario, assume that you are not using Change Management.)
    - d) On the **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel, enter the name of the data set where you want the apply job generated.
    - e) Edit the generated compare job as needed, and submit the job.
    - f) Check the output to confirm that the job completed successfully.
  9. Run the generated apply job to create the new TESTDB objects.

## Scenario: Converting partitioned table spaces to partition-by-range universal table spaces

You can use Db2 Object Comparison Tool to change a group of partitioned table spaces to partition-by-range (PBR) universal table spaces (UTS) without having to alter each table space individually.

### About this task

This process does not generate batch jobs. However, you can modify existing compare batch jobs to perform a similar function. The key is to create a mask that overwrites the SEGSIZE value of the table spaces.

This scenario assumes that the partitioned table spaces use table-controlled partitioning. If your partitioned table spaces uses index-controlled partitioning, you must convert them to use table controlled-partitioning before following the steps in this scenario. See [Converting table spaces to use table-controlled partitioning \(Db2 1.3 for z/OS\)](#).

### Procedure

To change partitioned table spaces to partition-by-range universal table spaces:

1. Specify the compare source.

The source of the comparison operation can be a database, which includes all table spaces in that database, or a specified list of table spaces. This scenario specifies the source as a database that contains the partitioned table spaces.

- a) On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **1 - Specify compare source (new)** and press Enter.
- b) On the **Specify Compare Source (GOC1)** panel, select option **2 - Source is from the DB2 catalog** and press Enter.
- c) On the **Specify DB2 Source Catalog Extract (GOC12)** panel, in the **Data set name** field, specify the name of the data set for the version file for the source.
- d) Select option **1 - Source is databases from the DB2 catalog** and press Enter.
- e) On the **Specify Source DB2 Databases (GOC1D)** panel, use the **I** line command to insert a line in the database list.
- f) On the **Compare Add Databases (GOC1DA)** panel, specify the source database by entering a partial database name and pressing Enter.
- g) On the **Compare Add Databases (GOC1DD)** panel, use **S** line command to select the database that you want to use as the compare source.

2. Specify the compare target.

In this case, you want the database to be compared to itself, so you need to specify that the target objects are to be automatically selected.

- a) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel and select option **2 - Specify compare target (old)**.
- b) On the **Specify Compare Target (GOC1)** panel, select option **4 - Target is from the DB2 catalog and the objects are automatically selected based on the selected source objects** and press Enter.
- c) On the **Specify Target DB2 Location (GOC14)** panel, complete the following fields and press Enter:

**Specify location name:**

The location of the Db2 subsystem.

**Data set name**

The name of the data set for the version file for the target. If the data set does not exist, it is created.

3. Create a mask to overwrite the SEGSIZE value of the table spaces.

- a) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel and select option **3 - Specify Compare Masks**.
- b) On the **Specify Compare Masks (GOC3)** panel, complete the following fields:

**Mask DSN**

The name of the data set for the masks. If the data set does not exist, it is created.

**Edit Mask**

YES

- c) On the **Edit Compare Masks (GOCEDIT)** panel, insert a line to create a mask to overwrite the SEGSIZE value. For example:

```
SEGSIZE:*,64
```

This line specifies that Db2 Object Comparison Tool is to find all table spaces in the compare scope that match this mask specification (for example, a table space named TS01) and change the value of its SEGSIZE to 64.

4. Generate and run the compare job:

- a) Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel and select option **5 - Generate compare job**.
- b) On the **Generate Compare Jobs (GOC5)** panel, specify the appropriate options, including the following settings, and press Enter.

**Suppress DROP of objects**

YES

**Generate apply jobs**

NO

**Run REORG/REBUILD**

A

- c) Submit the generated JCL job and check that it runs successfully.

5. Check the object comparison report.

The report shows that the only change to the affected table spaces is the one that was specified by the mask: the ALTER SEGSIZE operation. Additionally, it confirms that the table space changed from partitioned to partition-by-range.

**Example message in report:**

```
>ADB3320W :SEGSIZE was masked from 0 to 64 for table space DB5772.TS5772. The value might change the
table space type.
Compare tablespace source(Q79A.Q79A0100) and target(DB5772.TS5772)
(A)Tablespace change from partitioned to partition-by-range
(A)Field SEGSIZE changed from 0 to 64
Tablespace will be altered
```

6. Apply these changes or use Change Management to implement the changes.

As a result, any partitioned table space in the database is now a PBR UTS.

---

## Chapter 4. Comparing Db2 objects

The panel interface in Db2 Object Comparison Tool guides you through the options for comparing Db2 objects. Then, based on those options, the tool generates comparison batch jobs that you can save and reuse.

### Before you begin

Before you compare objects, complete the following actions:

- Make sure that the **Get DB2 ZPARM** field (the GETDB2ZP parameter) on the **DB2 Admin Defaults (ADB2P2)** panel is set to YES. For detailed instruction on how to change this value, see [Changing defaults \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

This option enables Object Comparison Tool to get Db2 subsystem parameter values by calling the stored procedure ADMIN\_INFO\_SYSPARM. These values are needed so that Object Comparison Tool can write a version file for each object that is being compared. *Version files* are snapshots of an object at a particular time and include the object definition.

- If you plan to request that Object Comparison Tool also generate apply jobs, make sure that you are connected to the target Db2 subsystem. *Apply jobs* are jobs that apply changes to the target; these changes are found during the comparison.

This prerequisite ensures that the apply jobs use the correct libraries. If you are connected to a different Db2 subsystem, you must manually update the apply jobs to use the correct Db2 libraries.

**Tip:** Before comparing objects, read “Performance considerations for Db2 Object Comparison Tool” on page 221 and [Chapter 18, “Recommendations when comparing a large number of objects,”](#) on page 217.

### Procedure

To compare Db2 objects:

**Note:** You can specify the W option to be guided directly through the next five steps [options 1 -5 on the **DB2 Object Comparison Tool Menu (GOCMENU)** panel] in succession without returning to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel and selecting the next option manually. When using the W option, the final panel for the current option includes a Continue command that you can use to display the next panel in the sequence.

1. [Specify source objects.](#)
2. [Specify target objects.](#)
3. Optional: [Specify compare masks.](#)
4. Optional: [Specify ignores.](#)
5. [Generate a compare batch job.](#)
6. Run the generated batch job to compare the objects.

### What to do next

After you run the comparison, [check the generated compare report](#) and then optionally [apply the changes to the target objects](#).

**Tip:** Before you exit Db2 Object Comparison Tool, consider saving your current compare batch job selections for later use. See [“Saving dialogs”](#) on page 160.

### Related tasks

[“Managing and restoring dialogs”](#) on page 159

In Object Comparison Tool, a *dialog* is a set of user selections for compare batch jobs. You can restore, rename, and delete any previously saved dialogs.

**Related reference**

“Object Comparison Tool main menu” on page 40

Use the **DB2 Object Comparison Tool Menu (GOCMENU)** panel to specify the criteria for the comparison that you want to run.

## 1. Specifying source objects

The first step in comparing Db2 objects is to specify the definition for the source object or objects. A *source object* is an object as you want it defined. The target object is compared to this source object and optionally changed to match it.

**About this task**

The definitions for source objects can be in a DDL file, the Db2 catalog, or a version file. When you specify one of these definition sources, Db2 Object Comparison Tool uses the object definition (and attributes) from that source for the comparison.

**Procedure**

To specify source object definitions:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option 1, and press Enter.
2. On the **Specify Compare Source (GOC1)** panel, specify where you want Object Comparison Tool to retrieve the definitions for the source objects.

```
GOC1 re ----- Specify Compare Source ----- 17:01
Option ==> VS

  1 - Source is from a DDL file
  2 - Source is from the DB2 catalog
  3 - Source is from a compare version file

VS - Source is from the DB2 catalog and the objects are selected from
    a version scope

Exclude Specification:
Owner . . . . . > (Optional, default is ELACZ, ? to lookup)
Name . . . . . > (Required, ? to lookup)
Edit objects . . . . YES (Yes/No)
```

Figure 8. **Specify Compare Source (GOC1)** panel

You can specify one of the following options:

Option	Description
<b>1</b>	Specifies that the definitions of the source objects are to be retrieved from a <i>DDL file</i> , that is, a file that contains SQL CREATE statements.
<b>2</b>	Specifies that the definitions of the source objects are to be extracted from the Db2 catalog. The definitions are extracted for one or more databases, table spaces, or tables and all dependent objects.  With option 2, you can optionally specify your own SQL SELECT statement against the catalog to identify the objects.
<b>3</b>	Specifies that the definitions of the source objects are to be retrieved from a previously created version file.

Option	Description
VS	Specifies that the definitions of the source objects are to be extracted from the Db2 catalog, and the objects are selected based on a version scope. Change Management must be enabled to select this option.

3. Optional: If you want to exclude specific objects, specify an exclude specification in the **Exclude Specifications** fields.

You can specify an existing exclude specification or create one. If you specify an existing one, you can edit it.

For detailed instructions on how to create an exclude specification or edit an existing one, see [“Creating or editing exclude specifications during a comparison” on page 144](#).

4. Press Enter, and complete one of the following procedures depending on that the option that you chose in step “2” on page 56:

Option	Description
1	<a href="#">“Specifying a DDL file for the source or target definition” on page 57</a>
2	If you want to identify individual objects: <a href="#">“Specifying the Db2 catalog for the source or target definition” on page 59</a> If you want to specify a SELECT statement against the catalog: <a href="#">“Specifying a SELECT statement for the source or target definition” on page 64</a>
3	<a href="#">“Specifying a version file for the source or target definition” on page 66</a>
VS	<a href="#">“Specifying a version scope for the source or target definition” on page 67</a>

#### Related concepts

[“Terminology in Db2 Object Comparison Tool ” on page 18](#)

Db2 Object Comparison Tool uses several terms that are unique to the product.

## Specifying a DDL file for the source or target definition

Db2 Object Comparison Tool can use a file that contains data definition language (DDL) for the definitions of the source or target objects. Object Comparison Tool processes everything in the DDL file; objects are not selected based on type or name. If your DDL defines a single table, only that table is used.

### Before you begin

This procedure assumes that you have completed the steps in [“1. Specifying source objects” on page 56](#) or [“2. Specifying target objects” on page 68](#) and specified option 1 on the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel. The **Specify Source DDL File (GOC11)** panel or the **Specify Target DDL File (GOC11)** panel should be displayed.

Ensure that the DDL that you specify meets the requirements for the DDL file extraction program, as documented in [“Supported SQL statements for DDL file extraction” on page 171](#). (The DDL file extraction program reads the specified DDL files and generates version files for the comparison.)

**Note:** If the source and target are both DDL, the SYSPRINT data set lists the values for Db2 function level and APPLCOMPAT as NA, because no Db2 connection exists to obtain accurate values.

### Procedure

To specify a DDL file for the source or target definition:

1. On the **Specify Source DDL File (GOC11)** panel or the **Specify Target DDL File (GOC11)** panel, in the **DDL data set** field, specify the name of the data set that contains the DDL for the source or target object or objects:

```

GOC11 e                               Specify Source DDL File
Option ==>

More:      +

Specify input DDL file:
DDL data set . .

Specify compare version file output:
Version table entry:
  Owner . . .      >          (? to look up)
  Name  . . .      >          (? to look up)
Data set:
  Data set name .

Enter a description (optional):
Description . .

-----
| ----- DB2 Object Compare Warning ----- |
| You have asked to generate a version file, but the source for these |
| objects is a DDL file.  If this DDL does not include all dependent objects |
| and authorizations, any subsequent comparison with this version file |
| may lead to loss of these dependent objects.  Press ENTER to continue |
| or END to stop this operation. |
|                               Please read this carefully |
| -----
-----

```

**Note:** If Change Management (CM) is not enabled, the **Owner** and **Name** fields are not available on this panel.

Figure 9. *Specify Source DDL File (GOC11) panel*

The data set that you specify must contain valid SQL statements and must adhere to TSO naming conventions. The data set can be either of the following types:

- A fixed-block sequential data set (RECFM=Fx, LRECL=80)
- A member of a partitioned data set with a logical record length of 80 (RECFM=Fx, LRECL=80)

The SQL statements that define the objects must be in columns 1-72 of the data set. Elements of a DDL statement can span records in the data set. Column 1 of a record is considered to immediately follow column 72 of the previous record. This convention can be used for long names or long string constants.

2. Specify where the version file is to be written.

The *version file* contains the object definitions that are extracted by the Object Comparison Tool DDL extract program and become input to the compare process. You can specify one of the following locations for the version file:

**Output data set**

To specify that the version file be written to an output data set, enter a value in the **Data set name** field. The specified data set must be one of the following types:

- A variable-block sequential data set
- A member of a partitioned data set with a record length of 16 KB (RECFM=Vx, LRECL=16384)

If the specified data set does not exist, it is created. If an existing data set is specified, it is overwritten.

**CM database**

To specify that the version file be written to the CM database, specify values in the version table **Owner** and **Name** fields. To select a version file from the CM database, specify a question mark (?) in the field. You can choose a value on the subsequent **CM Versions (ADB2C41)** panel.

If the **Owner** and **Name** fields are not visible, CM is not enabled, and you must specify an output data set instead.

If you select the option to store the version file in the CM database, an additional step is created in the compare job to store the version file for both the source and target objects in the database.

If both the database and the data set are specified, the information in the data set field is used to determine the output destination for the version file.

3. Optional: In the **Description** field, specify a description of the source or target Db2 objects.

The description is printed in the comparison report, placed in the header record of the version file, and used to describe the common properties of the Db2 objects.

4. Press Enter.

The **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed. DDL file specified is listed as the **Specification Status** for the source or target, depending on which one you specified.

## What to do next

If you specified the source object definitions, specify target object definitions.

If you specified the target object definitions, specify compare masks (optional), specify ignore fields (optional), or generate a compare batch job.

### Related concepts

“Batch DDL file extraction program ” on page 171

The DDL file extraction program interprets a source file of DDL statements that define Db2 objects. The program generates a *version file*, which contains records that are similar in format to those in the Db2 catalog that defines the same objects.

### Related information

Change Management (CM) (IBM Db2 Administration Tool for z/OS 13.1.0)

## Specifying the Db2 catalog for the source or target definition

Db2 Object Comparison Tool can extract the source or target object definitions from the Db2 catalog.

### Before you begin

This procedure assumes that you have completed the steps in “1. Specifying source objects” on page 56 or “2. Specifying target objects” on page 68 and specified option 2 on the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel. The **Specify DB2 Source Catalog Extract (GOC12)** panel or the **Specify DB2 Target Catalog Extract (GOC12)** panel should be displayed.

### Procedure

To specify the Db2 catalog for the source or target definition:

1. On the **Specify DB2 Source Catalog Extract (GOC12)** panel or the **Specify DB2 Target Catalog Extract (GOC12)** panel, specify where the version file is to be written:

```

Compare ----- Specify DB2 Catalog Extract ----- 11:08
Option ==>

  1 - Source is databases from the DB2 catalog
  2 - Source is table spaces from the DB2 catalog
  3 - Source is tables from the DB2 catalog
  4 - Add schema objects to the DB2 Source catalog extract
  5 - Target is the result of an SQL SELECT statement

Specify compare version file output:
Version table entry:
  Owner . . .                (? to look up)
  Name . . .                 (? to look up)
Data set:
  Data set name . .

Enter a description (optional):
  Description . .

```

Figure 10. **Specify DB2 Source Catalog Extract (GOC12)** panel

The *version file* contains the object definitions that are extracted by the Object Comparison Tool DDL extract program and become input to the compare process. You can specify one of the following locations for the version file:

### Output data set

To specify that the version file be written to an output data set, enter a value in the **Data set name** field. The specified data set must be one of the following types:

- A variable-block sequential data set
- A member of a partitioned data set with a record length of 16 KB (RECFM=Vx, LRECL=16384)

If the specified data set does not exist, it is created. If an existing data set is specified, it is overwritten.

### CM database

To specify that the version file be written to the CM database, specify values in the version table **Owner** and **Name** fields. To select a version file from the CM database, specify a question mark (?) in the field. You can choose a value on the subsequent **CM Versions (ADB2C41)** panel.

If the **Owner** and **Name** fields are not visible, CM is not enabled, and you must specify an output data set instead.

If you select the option to store the version file in the CM database, an additional step is created in the compare job to store the version file for both the source and target objects in the database.

If both the database and the data set are specified, the information in the data set field is used to determine the output destination for the version file.

2. Optional: In the **Description** field, specify a description of the source or target Db2 objects. The description is printed in the comparison report, placed in the header record of the version file, and used to describe the common properties of the Db2 objects.
3. Specify which object definitions you want extracted from the Db2 catalog for the source or target by specifying one of the following options, and press Enter:

Option	Description
<b>1</b>	Databases and all dependent objects, such as table spaces, tables, views, indexes, aliases, synonyms, and so forth. You can subsequently select which databases. <b>Tip:</b> If the database scope is too large, you can select table spaces or tables.
<b>2</b>	Table spaces and all dependent objects. You can subsequently select which table spaces.
<b>3</b>	Tables and all dependent objects. You can subsequently select which tables.

Option	Description
4	<p>Schema-based objects, such as user-defined functions, user-defined types, stored procedures, and sequences.</p> <p>If you also have a trigger as a schema-defined object, it is extracted whenever you extract a table where a trigger is defined. Therefore, you do not need to define a trigger as a separate object.</p> <p>You can choose this option (4) separately or as an additional option combined with option 1, 2, or 3.</p>

If you want to specify a SELECT statement (option 5), see [“Specifying a SELECT statement for the source or target definition”](#) on page 64.

4. On the one of the following **Specify Source** or **Specify Target** panels, issue the I line command to add objects to the list, and press Enter:

- **Specify Source DB2 Databases (GOC1D)** panel
- **Specify Source DB2 Table Spaces (GOC1S)** panel
- **Specify Source DB2 Tables (GOC1T)** panel
- **Specify Source DB2 Schema (GOC1C)** panel
- **Specify Target DB2 Databases (GOC1D)** panel
- **Specify Target DB2 Table Spaces (GOC1S)** panel
- **Specify Target DB2 Tables (GOC1T)** panel
- **Specify Target DB2 Schema (GOC1C)** panel

For example:

```

Compare ----- Specify Source DB2 Databases ----- Row 1 of 4
Command ==>                                         Scroll ==> PAGE

Commands: RESET
Line commands:
D - Delete I - Insert

Select Database      Location
      *              *
-----
I .....
  AB                STPLEX4A_DSN8
  ADBDB010          STPLEX4A_DSN8
  ADBDCHD           STPLEX4A_DSN8

```

Figure 11. **Specify Source DB2 Databases (GOC1D)** panel

**Note:** The list of objects is empty the first time that the **Specify Source** or **Specify Target** panel is displayed.

**Tip:** If objects are listed that you do not want to include, use the D line command to remove them from the list. To remove all objects from the list, issue the RESET command.

5. On one of the following **Compare Add** panels, specify the Db2 catalog search criteria for the objects, and press Enter:

- **Compare Add Databases (GOC1DA)** panel
- **Compare Add Table Spaces (GOC1SA)** panel
- **Compare Add Tables (GOC1TA)** panel
- **Compare Add Schema (GOC1CA)** panel

For example:

```
Compare ----- DSN8 Compare Add Databases ----- 14:00
Option ==>

Enter the partial name of the database you want to add to the compare
operation:

  Partial database name . . . : DS%
  Location name . . . . . :

Press enter to search for the database.
```

Figure 12. **Compare Add Databases (GOC1DA)** panel

Depending on which panel is displayed, specify one or more of the following values:

**Partial database name**

The database name.

**Partial table space name**

The table space name

**Partial table owner**

The authorization ID of a table owner

**Partial table name**

The table name

**Partial schema name**

The schema name

**Location name**

A unique location name for an accessible server. If you do not specify a location name, the location name of the current server is used.

One of the following **Compare Add** panels displays the qualifying objects:

- **Compare Add Databases (GOC1DD)** panel
- **Compare Add Table Spaces (GOC1SD)** panel
- **Compare Add Tables (GOC1TD)** panel
- **Compare Add Schema (GOC1CD)** panel

For example:

```

Compare ----- DSN8 Compare Add Databases ----- Row 1 of 24
Command ==>                                         Scroll ==> PAGE

Valid line commands are:                               Location: STPLEX4A_DSN8
S - Select (add)

Select Database Action
      *         *
-----
DSG24D0G
DSG24D0X
DSG24D1Z
DSNAE71A
DSNAE71P
DSNATPDB
DSNDB04
DSNDB06
DSNDB07
DSNDPSM
DSNRGFDB
DSNRLST
DSNRTSDB
DSN8CDDB
DSN8D71L
DSN8TEMP

```

Figure 13. **Compare Add Databases (GOC1DD)** panel

6. Issue the S line command next to the objects that you want to select, and press Enter.

The panel shows which objects are added. For example, the following panel shows that databases DSG24D0G and DSG24D1Z were added to the source for the comparison:

```

Compare ----- DB2 Compare Add Databases ----- Row 1 of 16
Command ==>                                         Scroll ==> PAGE

Valid line commands are:                               Location: STPLEX4A_DSN8
S - Select (add)

Select Database Action
      *         *
-----
*   DSG24D0G Added
      DSG24D0X
      DSG24D1Z Added
      DSNAE71A
      DSNAE71P
      DSNATPDB
      DSNDB04
      DSNDB06
      DSNDB07
      DSNDPSM
      DSNRGFDB
      DSNRLST
      DSNRTSDB
      DSN8CDDB
      DSN8D71L
      DSN8TEMP

```

Figure 14. **Compare Add Databases (GOC1DD)** panel

**Tip:** If you want to select all objects listed, issue the ALL command, which automatically selects the objects and exits the panel.

7. Exit (PF3) back to one of the following **Specify Source** or **Specify Target** panels:

- **Specify Source DB2 Databases (GOC1D)** panel
- **Specify Source DB2 Table Spaces (GOC1S)** panel
- **Specify Source DB2 Tables (GOC1T)** panel
- **Specify Source DB2 Schema (GOC1C)** panel

- **Specify Target DB2 Databases (GOC1D)** panel
- **Specify Target DB2 Table Spaces (GOC1S)** panel
- **Specify Target DB2 Tables (GOC1T)** panel
- **Specify Target DB2 Schema (GOC1C)** panel

The objects that you added are listed.

**Restriction:** You cannot compare objects from different locations, so ensure that all listed objects are from the same location.

8. Exit (PF3) to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel.

*object types* extract specified is listed as the **Specification Status** for the source or target, depending on which one you specified.

## What to do next

If you specified the source object definitions, specify target object definitions.

If you specified the target object definitions, specify compare masks (optional), specify ignore fields (optional), or generate a compare batch job.

## Specifying a SELECT statement for the source or target definition

You can specify a SELECT statement against the Db2 catalog to identify the source or target for a comparison. Db2 Object Comparison Tool uses all objects that are returned by the query for the source or target definition. The advantage of using a SELECT statement to identify the source or target is that you can use clauses to filter the object list.

### Before you begin

This procedure assumes that you have completed the steps in “1. Specifying source objects” on page 56 or “2. Specifying target objects” on page 68 and specified option 2 on the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel. The **Specify DB2 Source Catalog Extract (GOC12)** panel or the **Specify DB2 Target Catalog Extract (GOC12)** panel should be displayed.

### Procedure

To specify a SELECT statement for the source or target definition:

1. On the **Specify DB2 Source Catalog Extract (GOC12)** panel or the **Specify DB2 Target Catalog Extract (GOC12)** panel, specify where the version file is to be written:

```
Compare ----- Specify DB2 Catalog Extract ----- 11:08
Option ==>

    1 - Source is databases from the DB2 catalog
    2 - Source is table spaces from the DB2 catalog
    3 - Source is tables from the DB2 catalog
    4 - Add schema objects to the DB2 Source catalog extract
    5 - Source is the result of an SQL SELECT statement

Specify compare version file output:
Version table entry:
  Owner . . .                (? to look up)
  Name  . . .                (? to look up)
Data set:
  Data set name . .

Enter a description (optional):
  Description . .
```

Figure 15. **Specify DB2 Source Catalog Extract (GOC12)** panel

The *version file* contains the object definitions that are extracted by the Object Comparison Tool DDL extract program and become input to the compare process. You can specify one of the following locations for the version file:

### Output data set

To specify that the version file be written to an output data set, enter a value in the **Data set name** field. The specified data set must be one of the following types:

- A variable-block sequential data set
- A member of a partitioned data set with a record length of 16 KB (RECFM=Vx , LRECL=16384)

If the specified data set does not exist, it is created. If an existing data set is specified, it is overwritten.

### CM database

To specify that the version file be written to the CM database, specify values in the version table **Owner** and **Name** fields. To select a version file from the CM database, specify a question mark (?) in the field. You can choose a value on the subsequent **CM Versions (ADB2C41)** panel.

If the **Owner** and **Name** fields are not visible, CM is not enabled, and you must specify an output data set instead.

If you select the option to store the version file in the CM database, an additional step is created in the compare job to store the version file for both the source and target objects in the database.

If both the database and the data set are specified, the information in the data set field is used to determine the output destination for the version file.

2. Optional: In the **Description** field, specify a description of the source or target Db2 objects. The description is printed in the comparison report, placed in the header record of the version file, and used to describe the common properties of the Db2 objects.
3. Specify option 5, and press Enter.

The **Define SQL SELECT Statement For SOURCE (GOCSQ)** panel or **Define SQL SELECT Statement For TARGET (GOCSQ)** panel is displayed:

```
GOCSQ min ----- Define SQL SELECT Statement For SOURCE ----- 12:12
Option ==>

1 - Edit SQL SELECT statement
Data set name . . (Default if blank: 'TS6462.GOCSQL.SOURCE')
Location name . . RS22DC1A >
                  (Default if blank: RS22DD1A)
```

Figure 16. **Specify Source DB2 Databases (GOC1D)** panel

4. Optional: Specify a data set name and the location of the subsystem that contains the objects to be compared.

If you do not specify a data set name, a temporary data set is created with the default name listed. If you do not specify a location, the local subsystem is used.

If the specified data does not already exist, it is created.

5. Specify option 1, and press Enter.
6. Write or edit the SQL statement as needed.

The SELECT statement must query the Db2 catalog and return the following columns:

- TYPE CHAR(2)
- QUAL VARCHAR(128)
- NAME VARCHAR(128)

The SELECT statement can also optionally return the following additional columns:

- VERSION VARCHAR(122)
- INCLUDE CHAR(8)
- XDTYPE CHAR(8) (exclude)

For example, the following query returns table spaces created by TS3071:

```
SELECT 'TS' AS TYPE, DBNAME AS QUAL, NAME
FROM SYSIBM.SYSTABLESPACE
WHERE CREATOR = 'TS3071'
```

**Note:** If the SQL returns XDTYPE requests, make sure that the SQL returns those rows first.

7. Exit (PF3) to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel.

SQL SELECT (DB2 catalog extract) is listed as the **Specification Status** for the source or target, depending on which one you specified.

## What to do next

If you specified the source object definitions, [specify target object definitions](#).

If you specified the target object definitions, [specify compare masks \(optional\)](#), [specify ignore fields \(optional\)](#), or [generate a compare batch job](#).

## Specifying a version file for the source or target definition

If the object or objects were previously part of another comparison, you can specify an existing version file to be used for the source or target definition. A *version file* is created by Db2 Object Comparison Tool during the compare process. This file includes the object definitions at a particular point in time.

### Before you begin

This procedure assumes that you have completed the steps in [“1. Specifying source objects”](#) on page 56 or [“2. Specifying target objects”](#) on page 68 and specified option 3 on the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel. The **Specify Source Compare Version File (GOC13)** panel or the **Specify Target Compare Version File (GOC13)** panel should be displayed.

**Tip:** if the version file is old, consider converting to the latest Db2 version to save time. See [Chapter 11, “Converting version files to the latest Db2 version,”](#) on page 163.

### Procedure

On the **Specify Source Compare Version File (GOC13)** panel or the **Specify Target Compare Version File (GOC13)** panel, specify the version file that you want to use, and press Enter.

```
Compare ----- Specify Source Compare Version File -----
Command ==>
```

```
Specify compare version file (Source):
```

```
Version table entry:
```

```
Owner . . . > (? to look up)
Name . . . > (? to look up)
```

```
Data set:
```

```
Data set name . . .
Data set rewritable . . . (Yes/No)
Estimated record num. . . 0 (Only for DS cannot be rewritten)
```

Figure 17. **Specify Source Compare Version File (GOC13)** panel

You can specify one of the following sources for the previously created version file that you want to use as the source or target:

## CM database

To specify that the version file is in the CM database, specify values in the version table **Owner** and **Name** fields. To select a version file from the CM database, specify a question mark (?) in the field. You can choose a value on the subsequent **CM Versions (ADB2C41)** panel.

If the **Owner** and **Name** fields are not visible, CM is not enabled, and you must specify a data set instead.

## Data set

To specify that the version file is in a data set, enter a value in the **Data set name** field.

If you specify a data set, also specify whether the version file can be rewritten (in the **Data set rewritable** field). The default value is YES. If you specify NO, also specify a value for the **Estimated record num** field. If the version file can be rewritten, Db2 Object Comparison Tool counts this record number for you and writes it in the version file.

## Results

The **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed, and Compare version file specified is listed as the **Specification Status** for the source or target, depending on which one you specified.

## What to do next

If you specified the source object definitions, specify target object definitions.

If you specified the target object definitions, specify compare masks (optional), specify ignore fields (optional), or generate a compare batch job.

## Specifying a version scope for the source or target definition

If Change Management (CM) is enabled, you can select a version scope for your source or target. A *version scope* is a predefined set of objects. Version scope definitions are stored in CM tables and can include databases, table spaces, tables, indexes, views, stored procedures, triggers, and other objects. If you specify a version scope as your source or target, the definitions of the specified objects are retrieved from the Db2 catalog for the comparison.

## Before you begin

This procedure assumes that you have completed the steps in “[1. Specifying source objects](#)” on page 56 or “[2. Specifying target objects](#)” on page 68 and specified option VS on the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel. The **Specify Source Version Scope (GOC1VS)** panel or the **Specify Target Version Scope (GOC1VS)** panel should be displayed.

## Procedure

To specify a version scope for the source or target definition:

1. On the **Specify Source Version Scope (GOC1VS)** panel or the **Specify Target Version Scope (GOC1VS)** panel, specify values in the version table **Owner** and **Name** fields to identify the version scope that you want to use to extract definitions from the Db2 catalog:

```

GOC1VS ----- Specify Source Version Scope -----
Command ==>

Specify version scope (Source):
  Owner . . . . . >          (? to look up)
  Name  . . . . . >          (? to look up)

Specify compare version file output:
  Data set name .

Enter a description (optional):
  Description . .

```

Figure 18. **Specify Source Version Scope (GOC1VS)** panel

2. In the **Data set name** field, specify the output data set name for the version file.

The specified data set must be one of the following types:

- A variable-block sequential data set
- A member of a partitioned data set with a record length of 16K (RECFM=Vx , LRECL=6144)

If the specified data set does not exist, it is created. If an existing data set is specified, it is overwritten.

3. Optional: In the **Description** field, specify a description of the source or target Db2 objects.

The description is placed in the header record of the version file.

4. Press Enter.

The **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed, and Version scope specified (DB2 catalog) is listed as the **Specification Status** for the source or target, depending on which one you specified.

## What to do next

If you specified the source object definitions, [specify target object definitions](#).

If you specified the target object definitions, [specify compare masks \(optional\)](#), [specify ignore fields \(optional\)](#), or [generate a compare batch job](#).

### Related information

[Version scopes \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## 2. Specifying target objects

After you specify the comparison source, the next step is to specify the target. The *target* is the object or objects that you want to compare to the source.

### Procedure

To specify the target object definitions:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option 2, and press Enter.
2. On the **Specify Compare Target (GOC1)** panel, specify where you want Object Comparison Tool to retrieve the definitions for the target objects, and press Enter:

```

GOC1 re ----- Specify Compare Target ----- 17:15
Option ==>

1 - Target is from a DDL file
2 - Target is from the DB2 catalog
3 - Target is from a compare version file
4 - Target is from the DB2 catalog and the objects are automatically
  selected based on the selected source objects
VS - Target is from the DB2 catalog and the objects are selected from
    a version scope

Exclude Specification:
Owner . . . . . > (Optional, default is ELACZ, ? to lookup)
Name . . . . . > (Required, ? to lookup)
Edit objects . . . . YES (Yes/No)

```

You can specify one of the following options:

Option	Description
<b>1</b>	Specifies that the definitions of the target objects are to be retrieved from a <i>DDL file</i> , that is, a file that contains SQL CREATE statements.
<b>2</b>	Specifies that the definitions of the target objects are to be extracted from the Db2 catalog. The definitions are extracted for one or more databases, table spaces, or tables and all dependent objects.  With option 2, you can optionally specify your own SQL SELECT statement against the catalog to identify the objects.
<b>3</b>	Specifies that the definitions of the target objects are to be retrieved from a previously created version file.
<b>4</b>	Specifies that the target objects are selected based on the source objects.
<b>VS</b>	Specifies that the definitions of the targets objects are to be extracted from the Db2 catalog, and the objects are selected based on a version scope. Change Management must be enabled to select this option.

3. Optional: If you want to exclude specific objects, specify an exclude specification in the **Exclude Specifications** fields.

You can specify an existing exclude specification or create one. If you specify an existing one, you can edit it.

For detailed instructions on how to create an exclude specification or edit an existing one, see [“Creating or editing exclude specifications during a comparison”](#) on page 144.

4. Press Enter, and complete one of the following procedures depending on that the option that you chose in step “2” on page 68:

Option	Description
<b>1</b>	<a href="#">“Specifying a DDL file for the source or target definition”</a> on page 57
<b>2</b>	If you want to identify individual objects: <a href="#">“Specifying the Db2 catalog for the source or target definition”</a> on page 59  If you want to specify a SELECT statement against the catalog: <a href="#">“Specifying a SELECT statement for the source or target definition”</a> on page 64
<b>3</b>	<a href="#">“Specifying a version file for the source or target definition”</a> on page 66
<b>4</b>	<a href="#">“Specifying that the target definition is automatically selected based on the source”</a> on page 70
<b>VS</b>	<a href="#">“Specifying a version scope for the source or target definition”</a> on page 67

## Related tasks

[“1. Specifying source objects” on page 56](#)

The first step in comparing Db2 objects is to specify the definition for the source object or objects. A *source object* is an object as you want it defined. The target object is compared to this source object and optionally changed to match it.

## Specifying that the target definition is automatically selected based on the source

The target can be defined based on the source objects. In this case, Object Comparison Tool uses the source object names, in combination with masks and renames, to determine the target object names. Object Comparison Tool then extracts the definitions of the target objects from the Db2 catalog.

### Before you begin

This procedure assumes that you have completed the steps in [“2. Specifying target objects” on page 68](#) and specified option 4 on the **Specify Compare Target (GOC1)** panel. The **Specify Target DB2 Location (GOC14)** panel should be displayed.

**Important:** If you specify option 4 and the source is not a table space, objects that exist only in the target might be dropped. You can request that a warning message be issued for this situation by using the **Scope Warning Messages** option on the **Generate Compare Jobs (GOC5)** panel or the CM batch parameter SCOPE\_WARNING.

### Procedure

To specify that the target definition is automatically selected based on the source:

1. On the **Specify Target DB2 Location (GOC14)** panel, specify the Db2 location name for the target and the name of the output data set for the version file:

```
Compare ----- Specify DB2 Location -----
Option ==>

Specify location name: DB8A

Specify output compare version file:
Version data set : VERLIB.NEW(V2)

Enter a description (optional):
Description . . : APPLICATION V2

----- DB2 Object Compare Warning -----
| Target objects will be automatically selected based on the objects you |
| selected for the source. If the source version file does not include all |
| dependent objects, any subsequent comparison with the target may lead to |
| a loss of these dependent objects. To prevent a loss of objects, select |
| "Suppress DROP of objects" when you generate the compare job(s). |
| Press ENTER to continue or END to stop this operation. |
| Please read this carefully |
-----
```

The specified output data set must be one of the following types:

- A variable-block sequential data set
- A member of a partitioned data set with a record length of 16 KB (RECFM=Vx, LRECL=16384)

If the specified data set does not exist, it is created. If an existing data set is specified, it is overwritten.

2. Optional: In the **Description** field, specify a description of the Db2 objects that you are comparing. The description is included in the comparison report and in the header record of the version file.
3. Press Enter.

Object Comparison Tool determines the set of objects that are defined in the source version file and extracts the definitions of these objects from the Db2 catalog for the target.

The **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed, and the **Specification Status** for the target is listed as Automatic (DB2 catalog extract) .

## What to do next

[Specify compare masks \(optional\)](#), [specify ignore fields \(optional\)](#), or [generate a compare batch job](#).

### Related information

[Version scopes \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## 3. Specifying compare masks

If you need to account for different naming conventions between the objects you are comparing or overwrite certain attributes, specify masks. This step is optional.

You can define a mask either in a data set or in a table in the Change Management repository. Storing masks in a data set makes copying mask files easy. Storing masks in a table makes them easy to share, manage, and recover.

**Tip:** If you want to mask the schema and the owner, you must specify masks for both schema and owner, even if the values are the same.

### Before you begin

If you plan to define your masks in a table, Change Management must be enabled and the repository tables must be defined as part of the customization process of Db2 Administration Tool.

### Procedure

To specify compare masks:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option 3.
2. If you want to use a table in the Change Management repository for your masks, complete the following steps. Otherwise, if you want to use a data set for your masks, skip to step “3” on page 72.
  - a) On the **Specify Compare Masks (GOC3)** panel, specify an owner and name. This name identifies a row in the mask table in the Change Management database. That row contains (or will contain) the masks that you want to use for the comparison operation. You can specify either an existing name to identify an existing row in the mask table or a specify a new name to create a row in the mask table.

**Tip:** If the **Owner** and **Name** fields are not displayed, Change Management is not enabled. Either enable it or use a data set for your masks (as described in step “3” on page 72).

```
GOC3 re ----- Specify Compare Masks -----
Option ===>

Mask Table Entry:
  Owner . . MYID > (? to look up)
  Name . . MYMASK > (? to look up)
Data Set:
  Mask DSN . .
Options:
  Edit Mask . . YES (Yes/No)
```

Figure 19. **Specify Compare Masks (GOC3)** panel

- b) In the **Edit Mask** field, specify whether you want to edit this set of masks, and press Enter. If the values that you specified in the **Owner** and **Name** fields do not identify an existing set of masks, you must specify YES.

One of the following panels is displayed:

- If you specified the name of an existing set of masks and NO for **Edit Mask**, the **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed. Notice that for step 3 - **Specify compare masks**, the **Specification Status** field shows Mask specified. You have completed specifying your compare masks. You can skip the rest of this procedure and continue the process of [comparing Db2 objects](#).
- If you specified the name of an existing set of masks and YES for **Edit Mask**, the **Mask Lines (ADB2C2L)** panel is displayed. Skip to step “2.f” on page 72.
- If you specified a new name for a set of masks, the **Insert Mask (ADB2C22)** panel is displayed:

```
ADB2C22 n ----- CM - Insert Mask ----- 12:25
Command ==>

Owner . . . MYID      > (Optional, default is USERID)
Name  . . . MYMASK   > (Required, ? to look up)
Comment . . .                >
```

Figure 20. **Insert Mask (ADB2C22)** panel

- Optional: In the **comment** field, enter a description for this set of masks.
- Press Enter to insert a row into the Change Management repository table for this new set of masks.
- Press F3 to go to the **Mask Lines (ADB2C2L)** panel where you can define the masks.
- Insert and update lines as needed to define the masks that you want to use for the comparison.

For example, on the following panel, the TBNAME mask specifies that any table names of TB\_TEST are to be translated to TB\_PROD for the comparison. (This mask also affects the child masks SYNNAME, ALNAME, and VWNAME.)

For a complete list of mask names and syntax, see [Mask definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

```
ADB2C2L n ----- CM - Mask Lines ----- Row 1 to 1 of 1
Command ==>                               Scroll ==> PAGE

Mask lines for mask "MYID"."MYMASK"
Commands: CANCEL
Line commands:
  I - Insert  D - Delete  R - Repeat  M - Move  A - After  B - Before

Sel Sequence Req Type      From          To          Oper.  T
----- * * * * * -----> -----> -----
-
*          1    TBNAME    TB_TEST      TB_PROD      UPDATE
*          2    COLNAME    CELLNO      MOBILENO     UPDATE
*          3    SINGLECH  +
*          4    ALNAME     ALS+_TEST   ALS+_PROD
***** END OF DB2 DATA *****
```

Figure 21. **Mask Lines (ADB2C2L)** panel

- Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that for step 3 - **Specify compare masks**, the **Specification Status** field shows Mask specified. You have completed specifying your compare masks. You can skip the rest of this procedure and continue the process of [comparing Db2 objects](#).
- If you want to use a data set for your masks, complete the following steps:
    - On the **Specify Compare Masks (GOC3)** panel, in the **Mask DSN** field, specify the name of the data set.

You can specify an existing data set that already contains masks or a new data set that you want to use for masks. If the specified data set exists, it is reused. Otherwise, it is created.

The mask data set must adhere to TSO naming conventions and be one of the following types:

- A fixed-block sequential data set (RECFM=F $x$ )
  - A member of a partitioned data set with a record length of 80 (LRECL=80)
- b) In the **Edit Mask** field, specify whether you want to edit the mask data set by using ISPF edit, and press Enter. If the data set is new or does not contain mask definitions, specify YES.

One of the following panels is displayed:

- If you specified NO for **Edit Mask**, the **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed. Notice that for step **3 - Specify compare masks**, the **Specification Status** field shows Mask specified. You have completed specifying your compare masks. You can skip the rest of this procedure and continue the process of [comparing Db2 objects](#).
  - If you specified YES for **Edit Mask**, the **Edit Compare Masks (GOEDIT)** panel is displayed.
- c) On the **Edit Compare Masks (GOEDIT)** panel, add and change mask definitions as needed, and issue the SAVE command.

For a listing of content that is displayed on the **Edit Compare Masks (GOEDIT)** panel, see [“Mask data set”](#) on page 76.

For a complete list of mask names and syntax, see [Mask definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

- d) When you are done making changes, exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel. Notice that for step **3 - Specify compare masks**, the **Specification Status** field shows Mask specified.

## What to do next

Optionally [specify ignore fields](#) or [generate a compare batch job](#).

### Related concepts

[“Translation masks”](#) on page 73

In Object Comparison Tool, you can use translation masks to account for differences in naming conventions between source and target objects when doing a comparison. You can also use masks to overwrite values for object attributes.

### Related information

[Masks \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## Translation masks

In Object Comparison Tool, you can use translation masks to account for differences in naming conventions between source and target objects when doing a comparison. You can also use masks to overwrite values for object attributes.

This purpose is slightly different than the way masks are used in Db2 Administration Tool. In Db2 Admin Tool, you use masks to change the naming conventions that are used in the generated SQL. In Object Comparison Tool, masks are mainly used for translation. A name in the source can be translated with the mask so that it matches a name in the target. For example, if the source database is named SSEMMDB1 and the target is SSEMMDBA, a mask can tell Object Comparison Tool to compare the two databases even though they have different names.

For mask syntax, see [Mask definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#). The syntax is different depending on whether you are specifying a mask to translate names or a mask to overwrite attribute values.

If you specify both a translation mask and ignore fields, the ignore fields specification overrides the mask.

## Masks that translate names

Object Comparison Tool supports three types of translation masks to process names:

### AUTHID masks

AUTHID masks are applied to all fields that contain Db2 authorization IDs, such as OWNER and CREATOR.

For example, the following mask specifies that all authorization IDs that have the value SYSIBM in the source are translated to COPY:

```
AUTHID: SYSIBM, COPY
```

With the following example mask, an owner of PROD01 in the source is translated to PRODDB01.

```
AUTHID: *01*, *DB01*
```

You can also specify that you want to translate names for only specific authorization IDs, such as the table space owner. In that case, specify TSOWNER instead of AUTHID. For a complete list of AUTHID masks, see [Mask definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

### NAME masks

NAME masks are applied to all fields that name objects.

For example, the following mask specifies that any name that starts with ABC in the source is translated to a name that starts with DEF.

```
NAME: ABC*, DEF*
```

With the following example mask, name HLQ47D9 in the source is translated to NEW479 before it is compared with the target.

```
NAME: HLQ*D*, NEW**
```

You can also specify that you want to translate names for only specific types of objects, such as table spaces. In that case, specify TSNAME instead of NAME. For a complete list of name masks, see [Mask definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

### RENAME specifications

RENAME specifications indicate that an object in the source was renamed and should be related to an existing object in the target.

The syntax for RENAME is:

```
renameobj: old-name, new-name
```

where *old-name* is the previous name of the object and *new-name* is the new name of the object. You can use the wildcard character, an asterisk (\*), in the object names.

*renameobj* is a keyword that indicates the object that was renamed. *renameobj* can have one of the following values:

#### RENAMEDB

A database was renamed.

Example syntax: RENAMEDB: X\*, P\*

#### RENAMETS

A table space was renamed.

Example syntax: RENAMETS: X\*.X\*, P\*.P\*

#### RENAMETB

A table was renamed.

Example syntax: RENAMETB: \*.X\*, \*.P\*

## RENAMEIX

An index was renamed.

Example syntax: RENAMEIX:\*.X\*,\*.P\*

## RENAMEGV

A global variable was renamed.

Example syntax: RENAMEGV:\*.GVT\*,\*.GVS\*

## RENAMECOL

A column was renamed.

Example syntax: RENAMECOL:OWNER.MYTB.OLDCOLNAME,NEWCOLNAME

An error message is generated in any of the following situations:

- If the name of compared columns is specified as an input mask in the RENAME column mask.
- A column with the name of the output mask does not exist.

You can also specify masks to translate names for a specific object, such a specific table space instead of all table spaces. For details, see [Mask definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

When you specify compare masks, you do not have to specify the same number of characters for both the input mask and the output mask.

## Masks that overwrite attributes

The following example mask specifies that table spaces that start with TESTHRTS in the TESTDB database are to be compressed.

```
COMPRESS: TESTDB.TESTHRTS*, YES
```

The following example mask specifies that the SEGSIZE attribute for all source table spaces is to be changed to 64.

```
SEGSIZE:*, 64
```

Using masks to overwrite attributes can be useful when you want to overwrite attributes for a large group of objects. For an example, see [“Scenario: Converting partitioned table spaces to partition-by-range universal table spaces”](#) on page 53

For more flexibility, you can also use a REXX user exit to specify the overwrite value for table space and index space attributes. For more information about these REXX user exits, see [Specifying a REXX user exit for the overwrite value \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

## Db2 catalog records and associated masks

For a list of the Db2 catalog columns that correspond to each mask, see [Db2 catalog columns and the corresponding masks \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

## Mask processing

Masks are applied to the source objects before they are compared with the target objects. Before the comparison process, any masks are applied to Db2 catalog fields in the version file for the source object so that the names match the naming convention of the target object.

You can use one or more translation masks on the source object to make it match the target object. Masks that translate names are processed first and then any masks that specify overwrite values are applied. Within each of those categories, masks are processed in the order that you list them.

When a value is translated, such as a name, the masks are processed one by one until a match is detected. A match means that both of the following conditions are true:

- The mask name is applicable to the value. For example, for a table name, mask names TBNAME and NAME are applicable.
- The value conforms to *inputmask* in the mask syntax. For example, PRODTAB1 conforms to mask PROD\*1.

The value is translated based on the *outputmask* value in the syntax, or, in the case where an attribute value is overwritten, the value of the attribute is overwritten to the new value. Only the first matching mask is used for a given value. If no matching mask is found, the value is not translated. Generally, you should put the most specific translation masks at the top of the mask file and the more general ones at the end.

**Performance tip:** Using many masks that translate names might increase processing time. If a match is not found early in the process, the program must search through the list of translation masks until a match is found.

### Related concepts

[“Special considerations for comparing Db2objects” on page 180](#)

You can perform most comparisons field by field, comparing the catalog records that represent the objects. However, special considerations are needed in some situations.

### Related tasks

[“3. Specifying compare masks” on page 71](#)

## Mask data set

The mask data set is prepopulated with information about the available masks.

When you edit a mask data set, the **Edit Compare Masks (GOEDIT)** panel displays the content of this data set. The following screen shows this panel as displayed when it first opens. The message lines (identified by ==MSG) list the available translation mask names. These lines also show the hierarchy of the mask names.

```

***** ***** Top of Data *****
==MSG>
==MSG> Mask Syntax:
==MSG>   field:[qual<.name>:]inmask,outmask
==MSG> Fields (hierarchy):
==MSG>   SINGLECH
==MSG>   COLNAME
==MSG>   NAME
==MSG>     DBNAME, TSNAME, IXNAME, UDFNAME, CONSNAME,
==MSG>     UDTNAME, COLLNAME, PKGNAME, PGMNAME, PLNNAME
==MSG>     DBRMNAME, STPNAME, SFNAME, TGNAME, GRPNAME,
==MSG>     VCATNAME, GBPNAME, TCNAME, PMNAME, MKNAME
==MSG>     SEQNAME, GVNAME
==MSG>     TBNAME
==MSG>     SYNNAME, ALNAME, VWNAME
==MSG>     BPNAME
==MSG>     TSBPNAME, IXBPNAME
==MSG>     SGNAME
==MSG>     TSSGNAME, IXSGNAME
==MSG> AUTHID
==MSG>   SQLID
==MSG>   SCHEMA
==MSG>     IXSCHEMA, PMSHEMA, MKSCHEMA, SETPATHSC
==MSG>     TGSCHEMA, UDTSHEMA, SEQSCHEMA, STPSCHEMA
==MSG>     UDFSCHEMA, GVSHEMA
==MSG>     TBSHEMA
==MSG>     ALSHEMA, VWSHEMA, SYNSHEMA
==MSG> OWNER
==MSG>   DBOWNER, TSOWNER, IXOWNER, SGOWNER
==MSG>   PKGOWNER
==MSG>   TOWNER
==MSG> GRANTID
==MSG>   GRANTOR, GRANTEE
==MSG> ROLE
==MSG>   DBROLE, TSROLE, TBROLE, IXROLE
==MSG> XMLSCHID
==MSG> WLMENV
==MSG> LOCATION
==MSG>

```

```

==MSG> Overwrite Syntax:
==MSG> Field:inmask,Overwrite_value
==MSG> Fields: Overwrite values:
==MSG> COMPRESS YES,NO,REXX exit (table spaces and indexes)
==MSG> TSCOMPRES YES,NO,FIXED,HUFFMAN,REXX exit (table spaces only)
==MSG> IXCOMPRES YES,NO,REXX exit (indexes only)
==MSG> SEGSIZE n (4-64 must be multiple of 4),REXX exit
==MSG> TSDSSIZE nG,REXX exit (table spaces only)
==MSG> IXDSSIZE nG,REXX exit (indexes only)
==MSG> PRIQTY n,n%,REXX exit (table spaces and indexes)
==MSG> TSPRIQTY n,n%,REXX exit (table spaces only)
==MSG> IXPRIQTY n,n%,REXX exit (indexes only)
==MSG> SECQTY n,n%,REXX exit (table spaces and indexes)
==MSG> TSSECQTY n,n%,REXX exit (table spaces only)
==MSG> IXSECQTY n,n%,REXX exit (indexes only)
==MSG> DEFER YES,NO,REXX exit (indexes only)
==MSG> DEFINE YES,NO,REXX exit (table spaces and indexes)
==MSG> TSDEFINE YES,NO,REXX exit (table spaces only)
==MSG> IXDEFINE YES,NO,REXX exit (indexes only)
==MSG> HASHSPC nK,nM,nG,REXX exit
==MSG> TBINLOBL n,REXX exit (tables only)
==MSG> DTINLOBL n,REXX exit (distinct types only)
==MSG> AUDIT CHANGES,ALL,NONE,REXX exit (tables only)
==MSG> CLOSE YES,NO,REXX exit (table spaces and indexes)
==MSG> TSCLOSE YES,NO,REXX exit (table spaces only)
==MSG> IXCLOSE YES,NO,REXX exit (indexes only)
==MSG> TRACKMOD YES,NO,REXX exit (table spaces only)
==MSG> DCAPTURE NONE,CHANGES,REXX exit (tables only)
==MSG> FREEPG n,REXX exit (table spaces and indexes)
==MSG> TSFREEPG n,REXX exit (table spaces only)
==MSG> IXFREEPG n,REXX exit (indexes only)
==MSG> PCTFREE n,REXX exit (table spaces and indexes)
==MSG> TSPCTFREE n,REXX exit (table spaces only)
==MSG> IXPCTFREE n,REXX exit (indexes only)
==MSG> TSPCTFUPD n,REXX exit (table spaces only)
==MSG> LOCKMAX n,SYSTEM,REXX exit (table spaces only)
==MSG> ERASE YES,NO,REXX exit (table spaces and indexes)
==MSG> TSERASE YES,NO,REXX exit (table spaces only)
==MSG> IXERASE YES,NO,REXX exit (indexes only)
==MSG> RESONDROP YES,NO,REXX exit (tables only)
==MSG> EDITPROC string,REXX exit (tables only)
==MSG> VALIDPROC string,REXX exit (tables only)
==MSG> TSPARTS n,REXX exit (table spaces)
==MSG> LOGGED YES,NO,REXX exit (table spaces only)
==MSG> LOCKSIZE TABLE,TABLESPACE,PAGE,ROW,LOB,ANY,REXX exit
==MSG> (table space only)
==MSG> MAXROWS n,REXX exit (tables only)
==MSG> GBPCACH SYSTEM,CHANGED,ALL,NONE,REXX exit
==MSG> (table spaces and indexes)
==MSG> TSGBPCACH SYSTEM,CHANGED,ALL,NONE,REXX exit
==MSG> (table spaces only)
==MSG> IXGBPCACH SYSTEM,CHANGED,ALL,NONE,REXX exit
==MSG> (indexes only)
==MSG> VOLATILE YES,NO,REXX exit (tables only)
==MSG> APPEND YES,NO,REXX exit (tables only)
==MSG> PADDED YES,NO,REXX exit (indexes only)
==MSG> COPY YES,NO,REXX exit (indexes only)
==MSG> MEMCLUS YES,NO,REXX exit (table spaces only)
==MSG> FIELDPROC string,REXX exit (tables only)
==MSG> INSALGO n (0-2),REXX exit (table spaces only)
==MSG> SGKEYLABL string,NO,NOKEYLABEL,REXX exit (stogroup only)
==MSG> TBKEYLABL string,NO,NOKEYLABEL,REXX exit (tables only)
==MSG>
==MSG> Verification mask Syntax:
==MSG> VER,Field:operand,value(,values),RC=x
==MSG> or
==MSG> VER,rexField:REXX(exitproc,parm1,parm2,...,parmn)
==MSG> where:
==MSG> Field: Same fields used by overwrites
==MSG> RextField Can be one of three options:
==MSG> 1. same fields used by overwrites
==MSG> 2. special REXX only field, OBJNAME or TSPARTS
==MSG> 3. two char object type code listed below:
==MSG> code Object type Catalog record
==MSG> SG Storage group SYSSTOGROUP
==MSG> DB Database SYSDATABASE
==MSG> TS Table space SYSTABLESPACE
==MSG> TB Table SYSTABLES
==MSG> IX Index SYSINDEXES
==MSG> TG Trigger SYSTRIGGERS
==MSG> FK Foreign Key SYSRELS
==MSG> PK Primary key SYSTABCONST

```



```

==MSG> LOCATION:LOC3*,LOCT*
==MSG> SETPATHSC:SYSIBM,SYSFUN
==MSG> SINGLECH:_
==MSG> SINGLECH:_,+
==MSG>
==MSG> Object-specific mask examples:
==MSG> TBSHEMA:CREATOR1.TB2:CREATOR1,NEW_CRE1
==MSG> IXNAME:IXOWN*.IX3*:IX3*,IX4*
==MSG> IXBPNAME:IXOWN1.INDX2:BP1,BP3
==MSG>
==MSG> Overwrite examples:
==MSG> COMPRESS:MYDB*.MYTS*,YES
==MSG> SEGSIZE:MYDB*.MYTS*,8
==MSG> DSSIZE:MYDB*.MYTS*,4G
==MSG> PRIQTY:*.*,REXX(MYPRIQTY,DBNAME='MYDBTEST')
==MSG> TSPRIQTY:MYDB*.MYTS*,30
==MSG> IXPRIQTY:MYCR*.MYIX*,25%
==MSG> IXSECQTY:MYCR*.MYIX*,REXX(MYSECQTY,IXNAME,IXCREATOR,PCT=20%)
==MSG> DEFER:USER001.*IXNAME,NO
==MSG> DEFINE:DBNAME*. *TSPC,REXX(MYDEFINE,DEFINE='YES')
==MSG> HASHSPC:TBCREATOR.MYTBNAME,100M
==MSG> TBINLOBL:TBCREATOR.MYTBNAME.COLNAME,16000
==MSG> DTINLOBL:DTCRE*.DTNAME*,16000
==MSG> IXCLOSE:MYCR*.MYIX*,NO
==MSG> AUDIT:MYDB*.MYTB*,CHANGES
==MSG> TRACKMOD:MYDB*.MYTS*,NO
==MSG> DCAPTURE:TBCRE*.MYTB*,NONE
==MSG> FREEPG:ABC*.DEF*,6
==MSG> IXPCTFREE:IXSCH1.IXNAME1,9
==MSG> LOCKMAX:DBTEST2.TSTEST2,SYSTEM
==MSG> TSERASE:DBTEST1.TSTEST1,NO
==MSG> RESONDR0P:TBCRE*.MYTB*,NO
==MSG> TSPCTFUPD:DB1.TS1,25
==MSG> INSALGO:DB1.TS1,2
==MSG> SGKEYLABL:SG1,DB2SYS_KEY01
==MSG> SGKEYLABL:SG1,NOKEYLABEL
==MSG> SGKEYLABL:SG1,NO
==MSG> TBKEYLABL:TBCRE*.MYTB*,DB2SYS_KEY02
==MSG> TBKEYLABL:TBCRE.MYTB,NOKEYLABEL**
==MSG> TBKEYLABL:TBCRE.MYTB,NO**
==MSG>
==MSG> Verification mask examples:
==MSG> VER,COMPRESS:EQ,YES,RC=4
==MSG> VER,EDITPROC:NE,PROC1,RC=8
==MSG> VER,TSPARTS:LT,65,RC=8
==MSG> VER,PCTFREE:GT,20,RC=8
==MSG> VER,SEGSIZE:LIST,4,8,12,RC=8
==MSG> VER,PCTFREE:RANGE,0,5,RC=4
==MSG> VER,OBJNAME:REXX(OBJTST)
==MSG> VER,SEGSIZE:REXX(SEGTST,MYSEGSZ)
==MSG> VER,MEMCLUS:EQ,NO,RC=8
==MSG> VER,FLDPROC:NE,FLDPROC1,RC=8
==MSG> VER,IXCOMPRES:EQ,YES,RC=4
==MSG> VER,TSCOMPRES:REXX(COMPTST,DBNAME,NAME,COMPRESS)
==MSG> VER,TSPCTFUPD:RANGE,10,20,RC=8
==MSG> VER,INSALGO:RANGE,0,2,RC=8
==MSG> VER,TBKEYLABL:NE,DB2SYS.KEY02,RC=8
==MSG>
==MSG> Verification object type mask examples:
==MSG> VER,IX:REXX(VERIX,TBCREATOR,TBNAME,NAME)
==MSG> VER,DB:REXX(VERDB,NAME,CREATOR,BPOOL)
==MSG>
==MSG> Syntax for info about renamed objects/columns:
==MSG> renameobj:old-name,new-name
==MSG> RENAMECOL:table-name.old-colname,new-colname
==MSG> ( + in col 72 indicates continuation on next line col 1)
==MSG> renameobj:
==MSG> RENAMEDB, RENAMETS, RENAMETB, RENAMEIX,
==MSG> RENAMEGV
==MSG> Examples:
==MSG> RENAMETB:OLDOWNER.OLDNAME,NEWOWNER.NEWNAME
==MSG> RENAMECOL:OWNER.MYTB.OLDCOLNAME,NEWCOLNAME
==MSG>
***** ***** Bottom of Data *****

```

## 4. Specifying ignores

When you compare objects, you can optionally specify ignores and ignore changes.

An *ignore* is one or more fields that Object Comparison Tool is to ignore when comparing Db2 catalog records. Specify ignores when the source and target objects have differences, but you do not want these differences to result in changes to the target.

An *ignore change* is a specified change to an object that was previously reported in a compare result and that you want to ignore. Ignore changes are reported, but no SQL statements are generated for the changes.

### Before you begin

If you want to specify an ignore change, you must have saved compare results. Optionally, you can define an ignore change specification in advance; see [“Creating ignore changes specifications” on page 151](#).

### About this task

When specifying ignores in Db2 Object Comparison Tool, you can define the ignores either in the Change Management (CM) database or in a data set. For more information about ignores, see [Ignore fields \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

### Procedure

To specify ignores to be used during the compare process:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option 4, and press Enter. The **Specify Compare Ignores (GOC4)** panel is displayed.

```
Compare ----- Specify Compare Ignores -----
Option ==>

Ignore Fields Specification:
Owner . . . . . > (? to look up)
Name . . . . . > (? to look up)
Data Set:
  Data Set Name . .
Options:
  Edit Ignore Fields Specification . . . NO (Yes/No)

Ignore Changes Specification:
Owner . . . . . > (? to look up)
Name . . . . . > (? to look up)
Edit Ignore Changes Specification . . . NO (Yes/No)
Display using a saved compare result . . NO (Yes/No)
Saved Compare Results:
  Owner . . . . . > (? to look up)
  Name . . . . . > (? to look up)
```

Figure 22. **Specify Compare Ignores (GOC4)** panel

**Note:** If Change Management is not enabled, only the **Data Set Name** and **Edit Ignore Fields Specification** fields are displayed.

2. Complete one or both of the following tasks, depending on the ignores that you want to use:
  - [“Specifying ignore fields” on page 81](#)
  - [“Specifying ignore changes” on page 83](#)

### Related information

[Ignore fields \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

# Specifying ignore fields

## Procedure

1. On the **Specify Compare Ignores (GOC4)** panel, specify the following information in the **Ignore Fields Specification** section:

```
Ignore Fields Specification:
Owner . . . . . > (? to look up)
Name . . . . . > (? to look up)
Data Set:
  Data Set Name . .
Options:
  Edit Ignore Fields Specification . . . NO (Yes/No)
```

Figure 23. Ignore Fields Specification section on the **Specify Compare Ignores (GOC4)** panel

- Specify one of the following sources that contains (or will contain) the ignore fields specifications:

### CM table

Specify **Owner** and **Name** values for the ignore. This name identifies a row in the Change Management repository table. That row contains (or will contain) the ignore fields that you want to use for the comparison operation. You can specify either an existing name to identify an existing row in the table or a specify a new name to create a row in the table.

### Data set

Specify a value for **Data set name**. The data set must adhere to TSO naming conventions and be one of the following types:

- A fixed-block sequential data set
- A member of a partitioned data set with a logical record length of 80 (RECFM=Fx, LRECL=80)

The input must be in columns 1-72 of the ignore data set.

If CM is not enabled, you must specify a data set. If both a CM table and a data set are specified, the information in the data set field is used.

- In the **Edit Ignore Fields Specification** field, specify whether you want to edit the ignore fields.

2. Press Enter.

One of the following panels is displayed:

- If you specified that you do not want to edit the ignore fields, the **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed. You have completed specifying ignore fields. You can skip the rest of this procedure.
- If you specified that you wanted to edit the ignore fields and your source is a data set or an existing row in the CM repository table, the **Specify Ignore Fields : Objects (GOCCI)** panel is displayed. Skip to step “4” on page 82.
- If you specified that you wanted to edit the ignore fields and your source is a new row in the CM repository table, the **Insert Ignore (ADB2C22)** panel is displayed.

3. On the **Insert Ignore (ADB2C22)** panel, create an ignore specification by completing the following steps:

```

DB2 Admin ----- Create Ignore Specification ----- 12:25
Command ==>

Owner . . . MYID      > (Optional, default is USERID)
Name . . . MYIGNORE  > (Required, ? to look up)
Comment . . .

```

Figure 24. **Insert Ignore (ADB2C22)** panel

- a) Optional: In the **Comment** field, specify a description of the ignore.
  - b) Press Enter to insert a row into the Change Management repository table for this new set of ignore fields.
  - c) Exit (PF3) to display the **Specify Ignore Fields : Objects (GOCCI)** panel.
4. On the **Specify Ignore Fields : Objects (GOCCI)** panel, edit the ignore fields as needed by completing the following steps:

```

Compare ----- Specify Ignore Fields: Objects ----- Row 1 of 17
Command ==>                                           Scroll ==> PAGE

Valid line commands are:
U - Update Ignore Fields

Select Object          Ignore Fields          Qualifier Name
*                    *                               *         *
-----
GENERIC                None
SYSCHECKS              None
SYSCOLUMNS            COLTYPE , LENGTH , SCALE , DEFAULT , DEFAULTVALUE
SYSCONTROLS            None
SYSDATABASE            CREATOR , STGROUP , BPOOL , INDEXBP
SYSDATATYPES           None
SYSENVIRONMENT         APPLCOMPAT
SYSFIELDS              None
SYSINDEXES             None
SYSINDEXPART           PQTY , SQTY , SECQTYI
SYSKEYS                None
SYSPACKAGE             None
SYSPARMS               None
SYSRELS                None
SYSROUTINES            None
SYSSEQUENCES           None
SYSSTOGROUP            VCATNAME
SYSTABLEPART           PQTY , SQTY , SECQTYI
SYSTABLES              STATUS , LABEL          SYSADM   LRC*87
SYSTABLES              LABEL , CHECKRID       SYSADM   LRC*88
SYSTABLES              AUDITING
SYSTABLESPACE          None
SYSTRIGGERS            None
SYSVIEWS               None
SYSVOLUMES             VOLID
XMLMODIFIER            None

```

Figure 25. **Specify Ignore Fields : Objects (GOCCI)** panel

**Tip:** Use caution when specifying ignores. If possible, use the generic ignore field specifications. These specifications provide for some common sets of fields that are often intentionally different on source and target systems. For more information about generic ignores, see [Generic ignores \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

- a) Issue the U (update) line command for the appropriate object (catalog table or the generic object), and press Enter.

The **Select Ignore Fields for object (GOCCIF)** panel is displayed for the selected object:

```

GOCCIF ----- Select Ignore Fields for SYSINDEXPART ----- Row 1 to 18 of 22

Valid line commands are:
  S - Select (add) field  U - Un-select  R - Repeat row

Select Fields          Action  Qualifier Name
  *                   *      *      *
----->
AVGKEYLEN
CREATEDTS
DSNUM
EXTENTS
FREEPAGE
GBPCACHE
INDEXTYPE
LEAFFAR
LEAFNEAR
LIMITKEY
OLDEST_VERSION
PARTITION
PCTFREE
PQTY
PSEUDO_DEL_ENTRIES
RBA_FORMAT
SECQTYI
SPACEF

```

Figure 26. **Select Ignore Fields for object (GOCCIF)** panel

- b) Select the fields that are to be ignored by using the S line command, and press Enter.  
 You can further limit an ignore field to certain objects by specifying values in the **Qualifier** and **Name** columns with wildcard characters in any field. You can modify the qualifier and name of a field only when that row is selected. To apply multiple *qualifier.name* combinations to a field, issue the R line command to create additional rows for that field.
- c) When you are done selecting the ignore fields, exit (PF3) back to the **Specify Ignore Fields : Objects (GOCCI)** panel.
- d) Exit (PF3) back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel.

**Related tasks**

[Specifying ignore changes](#)

## Specifying ignore changes

### Procedure

1. On the **Specify Compare Ignores (GOC4)** panel, specify one of the following sets of information in the **Ignore Changes Specification** section:
  - a. If you want to use a defined ignore changes specification, specify the owner and name of the specification and whether you want to edit the specification in the following fields:

```

Ignore Changes Specification:
Owner . . . . . > (? to look up)
Name . . . . . > (? to look up)
Edit Ignore Changes Specification . . . NO (Yes/No)

```

Figure 27. *Ignore Changes Specification* section on the **Specify Compare Ignores (GOC4)** panel

- b. If you want to use a saved compare result, specify that you want to display the saved compare result and an owner and name for the result in the following fields:

```

Display using a saved compare result . . YES (Yes/No)
Saved Compare Results:
  Owner . . . . . > (? to look up)
  Name . . . . . > (? to look up)

```

Figure 28. Ignore Changes Specification section on the **Specify Compare Ignores (GOC4)** panel

**Restriction:** You cannot specify both a and b; you can specify either an ignore changes specification or a saved compare result.

2. Press Enter.
3. Take one of the following actions, depending on which panel is displayed:
  - If you specified that you do not want to edit or display the ignore changes, the **DB2 Object Comparison Tool Menu (GOCMENU)** panel is displayed. You have completed specifying ignore changes. Skip the rest of this procedure.
  - If you specified that you want to edit the ignore changes specification, the **Ignored Changes List (ADBPCICL)** panel is displayed. Modify the specification as needed. See step “3” on page 153 in “Modifying ignore changes specifications” on page 152.
  - If you specified that you want to display the saved compare results, the **Compare Report (ADBPCRR)** panel is displayed. Select the changes that you want to ignore. See step “4” on page 151 in “Creating ignore changes specifications” on page 151.

## What to do next

[Generate a compare batch job.](#)

### Related tasks

[Specifying ignore fields](#)

## 5. Generating a compare job

A *compare job* performs the requested comparison.

### Before you begin

Before you can generate a compare batch job, you must specify the source and target objects. You can confirm whether these objects are specified by looking at the **Specification Status** column on the **DB2 Object Comparison Tool Menu (GOCMENU)** panel:

```

GOCMENU ----- DB2 Object Comparison Tool Menu 13.1.0 ----- 15:44
Option ==>

1 - Specify compare source (new)      Specification Status:
2 - Specify compare target (old)     Database extract specified
3 - Specify compare masks            Automatic (DB2 catalog extract)
4 - Specify ignores                  Mask specified
5 - Generate compare job              Ignore fields specified
                                      Not generated

```

If one or both these fields list **Incomplete**, you must specify these objects before continuing. Complete the following tasks as needed:

- “1. Specifying source objects” on page 56
- “2. Specifying target objects” on page 68

You can optionally complete the following additional tasks before generating a compare batch job:

- “3. Specifying compare masks” on page 71
- “4. Specifying ignores” on page 80

If you want to use previous compare batch job selections that were saved as a dialog, restore that dialog before you begin. See [Chapter 10, “Managing and restoring dialogs,” on page 159](#).

## Procedure

To generate a compare batch job:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option 5, and press Enter. The **Generate Compare Jobs (GOC5)** panel is displayed.
2. If want to use Change Management (CM) or do a multi-target import, skip the remaining steps and complete one of the following procedures instead:
  - [“Generating a compare batch job to make changes by using Change Management” on page 111](#)
  - [“Generating a compare batch job for a multi-target import” on page 114](#)
3. On the **Generate Compare Jobs (GOC5)** panel, specify your compare job options, and press Enter. For information about the options on this panel, see [“Compare job options” on page 87](#).

### Tips:

- To view only the minimal number of options that you need to set to run a comparison, use the simple display mode. You can toggle between simple and advance mode by using the SIM and ADV commands. For more information about these display modes, see [“Simple and advanced display modes” on page 89](#).
  - Set **Save compare results** to YES if you want to analyze data about the comparison, ignore changes, or increase the efficiency of subsequent comparisons. The saved compare results contain information about objects that were part of the comparison, including detected differences, changes to make, and how those changes are to be implemented. You can save the compare results only for tables, indexes, global variables, and distinct data types.
4. If the **Change Management Prompt (ADB2CMRO)** panel is displayed (because CM is enabled and optional for your ID), specify NO, and press Enter.
  5. Complete the requested input on any subsequent panels that are displayed.

One or more of the following panels might be displayed depending on the compare job options that you selected. For more information about one of these panels, see the related option or panel description.

Panel	Related option on Generate Compare Jobs (GOC5) panel or panel description
<b>Save Compare Results (ADB2C22)</b> panel	<a href="#">“Save compare results” on page 94</a>
<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<a href="#">“Compare reporting options” on page 110</a>
<b>Specify Data Set Name for Apply Jobs (GOC5AJ)</b> panel	<a href="#">“Generate apply jobs” on page 97</a>
<b>Specify Work Statement List Data Set (ADB2WLDA)</b> panel	<a href="#">“As work statement list” on page 98</a>
<b>Specify Work Statement List (ADB27WLD)</b> panel	<a href="#">“As work statement list” on page 98</a>
<b>Specify Work Statement List Data Set (GOC5WL)</b> panel	<a href="#">“As work statement list” on page 98</a>
<b>Specify Job Parameters (ADB2W1R)</b> panel	<a href="#">“As work statement list” on page 98</a>
<b>REBIND options (ADBPREBO)</b> panel	<a href="#">“REBIND options” on page 106</a>
<b>DB2 Object Compare Warning (GOCGMPW)</b> panel	<a href="#">“Generate apply jobs” on page 97</a>

6. If you requested a batch job (the **Generate online** option is set to NO), edit the generated JCL job as needed and submit it to run the comparison. Otherwise (if **Generate online** is set to YES) , the comparison process is run online.

For information about the Object Comparison Tool parameters in the generated JCL job, see [“Parameters in the generated compare batch job” on page 117.](#)

### **What to do next**

[Check the compare report.](#)

**Tip:** Consider saving your current compare batch job selections for later use. See [“Saving dialogs” on page 160.](#)

## Compare job options

When you generate a compare batch job, you can specify a number of options to control the behavior of the comparison operation and job. These options are listed on the **Generate Compare Jobs (GOC5)** panel.

```

GOC5 ----- Generate Compare Jobs -----
Command ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
  Worklist name . . . . . PQ76055N (also used as middle qualifier in DSNs)

Compare options:
  Suppress DROP of objects . NO          (Yes/No)
  Drop FKs not in source . . NO          (Yes/No)
  Suppress DROP of columns . NO          (Yes/No)
  Suppress adding columns . NO          (Yes/No)
  Run SQLID . . . . . (Blank, an SQLID, or <NONE>)
  Object Grantor . . . . . (Blank or an SQLID)
  Run Validate. . . . . V (Validate, None)
  Allow implicit drop of
    excluded objects . . . . NO          (Yes/No)
  Enable auth-switching . . . YES        (Yes/No)
  Disable REORG optimization YES        (Yes/No)
  Scope Warning Messages . . YES        (Yes/No)

Change reporting options . . YES        (Yes/No)
Save compare results . . . . YES        (Yes/No)

Data set information:
  PDS for batch jobs . . . . CMP.PQ76055N
  Prefix for data sets . . . NBRON
  Changes file data set name.
    Member name . . . . . (if Changes file is an existing PDS)

Options:
  Generate online . . . . . YES          (Yes/No)
  Single compare job . . . . NO          (Yes/No)
  Member name . . . . . COMPARE (default COMPARE)
  Allow deferred restart . . NO          (Yes/No)
  Generate apply jobs . . . . YES        (Yes, No, or (Delta) Change)
  Generate one job. . . . . YES        (Yes, No, or (Per) Process)
  Member prefix . . . . . APPLY (default APPLY)
  As work statement list . YES          (Yes/No to append to work stmt list)
  Embed IFF into WSL . . . NO          (Yes/No)
  Use customized util opts. YES          (Yes/No)
  Content of apply job(s) . ALL          (All, DDL)
  Unload method . . . . . P (Unload, Parallel unload, HPU)
  Generate templates. . . . NO          (Yes/No)
  Stop on conversion error. NO          (Yes/No)
  Use DEFER YES . . . . . YES          (Yes/No)
  Allow rotate parts . . . . YES        (Yes/No)
  Retain GENERATED ALWAYS:
    For ROWID . . . . . YES          (Yes/No)
    For ROW CHANGE TIMESTAMP. YES        (Yes/No)
  Retain START and RESTART values:
    For sequence object . . . (Yes/No)
  IDENTITY START value . . . ORIGINAL (Original, Computed)
  Mask ignored fields . . . . NO        (Yes/No)

  Optional jobs after Reload or Alter:
    Run CHECK DATA . . . . YES        (Yes/No)
    Take an image copy . . . R          (after: Reload/Alter/Both/None)
    Run REORG/REBUILD . . . M          (Mandatory, All relevant, None)
    Run RUNSTATS . . . . . R          (after: Reload/Alter/Both/Min/None)
    Run REBIND . . . . . M          (Mandatory, All relevant, None)
    REBIND options . . . . YES        (Yes/No)

BP - Change batch job parameters
TU - Specify TEMPLATE usage
UO - Customize utility options
CO - Change options common to change functions
ADV - Advanced options

```

**Note:** ADV is listed as a command if the display mode is simple. If the display mode is advanced, SIM is listed as a command instead.

Figure 29. **Generate Compare Jobs (GOC5)** panel

## Simple and advanced display modes

The **Generate Compare Jobs (GOC5)** panel can be displayed in the following modes:

### Simple mode

This display mode shows only those options that you must specify to run a basic comparison. You can switch to simple mode by using the SIM command.

### Advanced mode

This display mode shows all compare job options. You can switch to advanced mode by using the ADV command.

You can also change the display mode on the **Panel Display Options (ADBPPDO)** panel. For instructions, see [Setting panel display options \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

By default, the display mode is initially simple for new users when Db2 Admin Tool is first installed and advanced for existing users when Db2 Admin Tool is upgraded.

When simple display mode is used, Object Comparison Tool uses the default values for those options that are displayed only in advanced mode. The exception is if those advanced options were explicitly changed. If an advanced option was changed to a value that does not conflict with a simple option, that value is used. For example, if you toggle to advanced mode and specify a value for one or more options, those values are saved and used. Similarly, if those advanced options were changed in a previous session and saved, those saved values are used. In the case where a specified value for an advanced options conflicts with any simple options, the default value is used for that advanced option.

The following options are available in simple display mode:

- [“Worklist name” on page 91](#)
- [“Run SQLID” on page 92](#)
- [“Change reporting options” on page 94](#)
- [“PDS for jobs” on page 95](#)
- [“Prefix for data sets” on page 95](#)
- [“Changes file data set name” on page 95](#) and [“Member name” on page 95](#)
- [“Compare job member name” on page 96](#)
- [“Generate apply jobs” on page 97](#)
- [“Generate one job” on page 98](#)
- [“Member prefix” on page 98](#)
- [“Use customized util opts” on page 101](#)
- [“Generate templates” on page 101](#)

The following table shows those options that are available only in advanced display mode and the default value for each option. For a detailed description of the option, click the link.

*Table 6. Compare options that are displayed only in advanced mode*

Option	Default
<a href="#">“Suppress DROP of objects” on page 91</a>	YES
<a href="#">“Drop FKs not in source” on page 91</a>	NO
<a href="#">“Suppress DROP of columns” on page 92</a>	NO
<a href="#">“Suppress adding columns” on page 92</a>	NO
<a href="#">“Object Grantor” on page 92</a>	blank (an empty string)
<a href="#">“Run Validate” on page 93</a>	N
<a href="#">“Allow implicit drop of excluded objects ” on page 93</a>	NO

Table 6. Compare options that are displayed only in advanced mode (continued)

Option	Default
<a href="#">“Enable auth-switching” on page 93</a>	NO
<a href="#">“Disable REORG optimization” on page 94</a>	NO
<a href="#">“Scope Warning Messages” on page 94</a>	NO
<a href="#">“Save compare results” on page 94</a>	NO
<a href="#">“Generate online” on page 95</a>	NO
<a href="#">“Single compare job” on page 96</a>	YES
<a href="#">“Allow deferred restart” on page 96</a>	NO
<a href="#">“As work statement list” on page 98</a>	YES
<a href="#">“Embed IFF into WSL” on page 100</a>	NO
<a href="#">“Content of apply job(s)” on page 101</a>	ALL
<a href="#">“Unload method” on page 101</a>	U (Unload)
<a href="#">“Stop on conversion error ” on page 102</a>	NO
<a href="#">“Use DEFER YES” on page 102</a>	NO
<a href="#">“Allow rotate parts” on page 102</a>	YES
Retain GENERATED ALWAYS: <a href="#">“For ROWID” on page 103</a>	NO
Retain GENERATED ALWAYS: <a href="#">“For ROW CHANGE TIMESTAMP” on page 103</a>	NO
Retain START and RESTART values: <a href="#">“For sequence object:” on page 103</a>	NO
<a href="#">“IDENTITY START value” on page 103</a>	ORIGINAL
<a href="#">“Mask ignored fields” on page 104</a>	NO
<a href="#">“Run CHECK DATA” on page 104</a>	YES
<a href="#">“Take an image copy” on page 104</a>	B
<a href="#">“Run REORG/REBUILD” on page 105</a>	A
<a href="#">“Run RUNSTATS” on page 105</a>	B
<a href="#">“Run REBIND” on page 105</a>	A
<a href="#">“REBIND options” on page 106</a>	NO

## Option descriptions

The options on this panel are described in the following sections:

- [“Worklist information:” on page 91](#)
- [“Compare options:” on page 91](#)
- [“Reporting options:” on page 94](#)
- [“Data set information:” on page 95](#)
- [“Options:” on page 95](#)
- [“Optional jobs after Reload or Alter:” on page 104](#)

- [“Commands” on page 106](#)

## Worklist information:

### Worklist name

Specify the name of the work statement list (WSL) to use.

The specified name is also used for the following items:

- The middle qualifier in the names of the work data sets that are created for the job  
The prefix for these work data sets is the value in the **Prefix for data sets** field in the **Data set information** section of this panel. The complete data set name is the **Prefix for data sets** value, followed by the **Worklist name** value, and then a name that indicates the purpose of the data set. For example, for the changes file, the complete name might be NBRON.PQ76055N.CHANGES.
- A name for the Db2 Administration Tool SQL or DDL executor, which has a checkpoint facility.  
The **Worklist name** value is used as a key to the checkpoint table. Use a unique name for each comparison that you run.

### Related information:

[Work statement lists \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## Compare options:

### Suppress DROP of objects

Specify whether the compare process is to drop objects that are in the target but not in the source.

#### Values:

##### YES

Objects that are in the target but not in the source are retained. Specify YES to prevent the compare process from dropping any target objects.

##### NO

Objects that are in the target but not in the source are dropped.

#### Default value:

YES

Regardless of the value that you set for this option, Object Comparison Tool replaces all relationships between a parent and a child if a foreign key is specified in the source. To delete a foreign key, both the parent and the child must be present in the source (without a foreign key).

Also, if DROP statements are part of the source DDL, objects are dropped regardless of the value specified for this option.

Object Comparison Tool drops all explicit LOB objects from the target if they are not specified on the source. However, if the base table associated with the LOB objects is kept because **Suppress DROP of objects** is set to YES, all the LOB objects are kept.

### Drop FKs not in source

Specify whether the compare process is to drop from the target table any foreign keys that are not specified in the corresponding source table.

#### Values:

##### YES

Any foreign keys that are not specified in the corresponding source table are dropped.

##### NO

Drop behavior is determined by the **Suppress DROP of objects** field.

#### Default:

NO

### Suppress DROP of columns

Specify whether the compare process is to drop columns that are in the target tables but not in the source table.

#### Values:

##### YES

Any target table columns that are not in corresponding source table are retained. Specify YES to prevent the compare process from dropping any columns.

##### NO

Any target table columns that are not in corresponding source table are dropped.

#### Default value:

NO

### Suppress adding columns

Specify whether the compare process is to add source columns to the target. This option is useful if you have extra columns on your source that you do not want added to your target.

#### Values:

##### YES

Columns that exist in the source only are not added to the target. Specify YES to prevent columns in the source from being added to the target.

##### NO

Columns that exist in the source only are added to the target.

#### Default value:

NO

### Run SQLID

Specify a valid SQL ID to use when creating, dropping, or altering objects. This ID is typically an administrative SQL ID whose only privileges are to create objects.

#### Values:

##### blank

A SET CURRENT SQLID statement is generated in the DDL before each object that is created. Where possible, the SQL ID that was originally used to create the object is used in the SET statement.

##### SQLID

The specified SQL ID becomes the owner of the databases and table spaces. If the specified SQL ID is different from the current owner, the databases and table spaces (and all dependent objects) are dropped and recreated to change the owner.

##### <NONE>

A SET CURRENT SQLID statement is not generated in the DDL.

### Object Grantor

Specify an SQL ID to use in SET CURRENT SQLID statements that precede GRANT statements.

#### Values:

##### blank

No SET CURRENT SQLID statements are generated.

##### SQLID

The specified SQL ID is used in the SET CURRENT SQLID statements.

If an SQL ID is specified for this option, but the **Run SQLID** option is set to <NONE>, no SET CURRENT SQLID statements are generated to precede GRANT statements.

#### Default value:

blank (an empty string)

## Run Validate

Specify whether to perform consistency checking. Consistency checking verifies that for all the primary objects in the source DDL, any dependent objects exist. These dependent objects must exist in the source DDL or the target catalog.

### Values:

#### V (Validate)

The following checking is performed:

- The table space in a CREATE TABLE statement exists.
- The table in a CREATE INDEX statement exists.
- The child and parent tables in referential constraints exist.
- If the index in a CREATE INDEX statement is a clustering index, a clustering index does not already exist.
- For primary index and unique index changes, matching keys for primary keys and unique keys exist.
- For primary key and unique key changes, matching indexes for primary keys and unique keys exist.
- The number of index partitions matches the number of table space partitions.

Object Comparison Tool also checks that the dependent objects exist if the following statements are generated:

- CREATE TRIGGER
- CREATE VIEW
- CREATE MQT
- CREATE INDEX
- ADD FOREIGN KEY

When you request consistency checking, a consistency checks report (ADB2WVL) is generated. If a check fails, a message is written to the report with a return code of 8.

**Note:** For native stored procedures, even if validation is successful, the existence of the object in the native stored procedure body cannot be known at the procedure run time (or during the procedure call).

#### N (NONE)

Consistency checking is not done.

### Default value:

N

## Allow implicit drop of excluded objects

Specify whether excluded objects can be dropped implicitly.

### Values:

#### YES

Excluded objects can be dropped if needed and are then recreated according to the target definition.

#### NO

If an excluded object needs to be dropped, an error message is displayed and the compare fails.

### Default:

NO

## Enable auth-switching

Specify whether to generate DDL that is used by the authorization switching feature. *Authorization switching* enables you to run DDL and DCL under the authority of another user.

This field is visible only if the authorization switching facility is enabled for the subsystem during the customization process.

**Values:**

**YES**

DDL can be used by authorization switching.

**NO**

DDL cannot be used by authorization switching.

**Default value:**

NO

**Disable REORG optimization**

Specify whether you want to disable REORG optimization. REORG optimization reduces the number of REORG utility statements that are issued, and thus the number of times that your system halts.

**Values:**

**YES**

REORG optimization is disabled.

**NO**

REORG optimization is used.

**Default:**

NO

**Scope Warning Messages**

Specify whether to issue a warning message in the case where the target of an object comparison operation is automatically selected, and the source is not a table space. This message warns that objects that exist only in the target might be dropped.

**Values:**

**YES**

Message ADB7353 is issued for this situation. This message is issued regardless of the value of the **Suppress DROP of objects** option.

**NO**

No message is issued for this situation.

**Default:**

NO

**Reporting options:**

**Change reporting options**

Specify whether you want to change the options for reports.

**Values:**

**YES**

You are prompted to specify new reporting options on the **Specify Compare Reporting Options (GOC5RO)** panel after you press Enter. For more information about those options, see [“Compare reporting options” on page 110](#).

**NO**

No options are changed.

**Save compare results**

Specify whether compare results are to be saved.

**Restriction:** Compare results are saved for only the following objects:

- tables
- indexes
- global variables

- distinct data types

**Values:**

**YES**

Compare results are saved.

**NO**

Compare results are not saved.

**Default value:**

NO

If you specify YES, the following panel prompts you to specify a name for the saved result after you press Enter:

```
ADB2C22 n -----Save Compare Results ----- 08:22
Command ==>

Owner . . . OWN1      > (Optional, default is VNRG, ? to lookup)
Name . . . NEW1      > (Required, ? to lookup)
Comment . . MY FIRST COMPARE RESULT >
Eligible for auto-delete . . 30 (number of days, blank for no auto-delete)
```

Figure 30. **Save Compare Results (ADB2C22)** panel

**Data set information:**

**PDS for jobs**

Specify the name of the partitioned data set (PDS) where the compare jobs are to be generated.

**Prefix for data sets**

Specify the prefix to be used for data sets that are allocated by the batch job. For example, UNLOAD, DDL, and LOAD data sets can be allocated.

**Changes file data set name**

Specify the name of the data set to store the changes that are output by the compare job. This data set is used primarily for changes that are to be imported to Change Management. To generate these changes for Change Management, set **Generate apply jobs** to Change.

If this data set is partitioned, it must be preallocated, and you must specify a member name in the **Member name** field.

If you preallocate this data set, the data set must meet the following requirements:

- For delta changes, this data set must be either fixed length with an LRECL of 80 or variable length with an LRECL of 16384.
- For changes, this data set must be variable length with an LRECL of 16384.

**Member name**

If the CHANGES data set is partitioned, specify a member name.

**Options:**

**Generate online**

Specify whether the compare process is to be run online.

**Values:**

**YES**

The compare process runs immediately when you press the Enter key after specifying the compare job options. If **Generate apply jobs** and **As work statement list** are also set to YES, the work statement list is created online but is not run. You can run the work statement list later.

**NO**

A batch job is generated. You can submit this batch job later to perform the compare process in the background.

When the compare process is run in batch, messages are placed in the SYSPRINT data set. You can override this output data set by using the TU command to define the ADBWORK template. If you do not specify YES for the **Generate templates** option and define ADBWORK, the default data set name, *prefix.wsl.SYSPRINT*, is used.

**Default:**

NO

**Restriction:** This **Generate online** function is not available when comparing multiple sources and targets.

**Related information:**

[“Running a work statement list to apply changes” on page 157](#)

[“TU - Specify TEMPLATE usage” on page 108](#)

**Single compare job**

Specify whether all job steps are to be run in one job. Use separate jobs to run the source extraction on a system other than the target system.

**Values:****YES**

A single job is generated. Specify a member name for the job in the **Member name** field.

**NO**

Up to four jobs are generated for the following actions:

- Extract the source if the source is DDL or the Db2 catalog
- Extract the target if the target is DDL or the Db2 catalog
- Compare the source and target
- If **Generate apply jobs** is set to YES, generate apply jobs or register job when Change Management is enabled

**Default value:**

YES

**Compare job member name****Member name**

In simple mode, this field is called **Compare job member name**. In advanced mode, this field is called **member name**.

If you requested a single job, specify the name of the member where the compare job is to be generated. The default value is COMPARE.

**Allow deferred restart**

Specify whether the generated compare job is to support deferred step restart.

**Values:****YES**

The generated compare JCL is not to include backward references to previous steps, so that deferred step restart is possible. This option is ignored if **Generate online** is set to YES or **Single compare job** is set to NO.

**NO**

Generated compare JCL is to include backward references when allocating temporary data sets. Therefore, deferred restart is not possible without changing these backward references.

**Default:**

NO

## Related information:

[Backward references \(z/OS 3.1.0\)](#)

### Generate apply jobs

Specify whether to generate jobs to apply the changes that were found during the comparison to the target objects.

Instead of generating apply jobs, Object Comparison Tool can also generate a work statement list (WSL) or register a change in the Change Management database on the target subsystem. You can then use any of these sources (the apply jobs, work statement list, or change) to apply changes to the target object.

The **Generate apply jobs** function uses the following input:

- The shared variables file, which contains the variables that were specified in the panels.
- The changes file from the compare operation, which contains the following information:
  - The DROP, CREATE, and ALTER statements
  - The UNLOAD requests
  - Table space information records, which allow Object Comparison Tool to determine the size of the UNLOAD requests

**Restriction:** Do not attempt to import a changes file that is generated by a normal comparison job into Change Management as a change. Doing so can lead to loss of data when the change is run.

### Values:

#### YES

Jobs are generated to apply the changes.

If you request the generation of apply jobs, run the Object Comparison Tool dialog connected to the target Db2 subsystem to pick up the correct libraries for use in the apply jobs. Alternatively, if the Object Comparison Tool dialog is run on a different Db2 system, you must manually update the apply jobs to use the correct Db2 libraries.

When you specify YES, you also have the option to create a work statement list, use the utility options, select the unload method, and generate templates.

If you specify YES and **As work statement list** = NO, the following panel prompts you for a data set name for the apply jobs after you press Enter:

```
GOC5AJ  ----- Specify Data Set Name for Apply Jobs ----- 12:22
Enter/verify the following:
Data Set Name  ==> ADM001.COMPARE.D97220
```

Figure 31. **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel

#### NO

No apply jobs are generated.

#### CHANGE

Generate a delta change in the CHANGES file that can later be imported to Change Management. (The CMDELTA parameter for GOC2CMP). No apply jobs or work statement list is generated.

If you specify Change, the compare process is said to run in *CMDELTA mode*.

The **Generate apply jobs** function runs as an EXEC (ADBGAJOB) in a TSO/ISPF batch job and uses ISPF skeletons to generate the apply jobs or work statement list.

If you request an apply job where the source or target are from DDL input, they must include all dependent objects. Otherwise, the loss of objects or authorizations can result. In this case, Object

Comparison Tool issues the following warning message, which gives you the option to continue or to end the operation:

```
----- DB2 Object Compare Warning -----
You have asked to generate apply jobs, but the source and / or target
objects are not being extracted from the DB2 catalog. If your extracts
do not include all dependent objects and authorizations, this may
lead to the loss of these objects and/or authorizations.

Press ENTER to continue or END to stop this operation.
F1=HELP    F2=SPLIT    F3=END      F4=RETURN   F5=RFIND    F6=RCHANGE
F7=UP      F8=DOWN     F9=SWAP    F10=LEFT   F11=RIGHT

Please read this carefully |
```

Figure 32. **DB2 Object Compare Warning (GOCGMPW)** panel

### Generate one job

Specify whether to generate a single apply job.

**Restriction:** The **Generate one job** option does not apply when the **As work statement list** option is set to YES.

#### Values:

##### YES

A single apply job is generated. If the number of steps exceeds a maximum limit of 255, more than one job is generated.

If you specify Yes, specify a value in the **Member prefix** field.

##### NO

Multiple jobs are generated. For a list of those jobs by name, see [“Apply jobs” on page 155](#).

##### PROCESS

One job is created per process. For example, all UNLOAD jobs are merged into one job for each process. For a list of the generated, see [“Apply jobs” on page 155](#).

### Member prefix

Specify a prefix to use for the member name or names for the apply job. The default value is APPLY.

If the number of steps for the apply job exceed the limit of 255, more than one job is generated. For example, if **Member prefix** is APPLY, the member names are APPLY001, APPLY002 and so on.

**Member prefix** does not apply if you specify NO or PROCESS for **Generate one job**.

### As work statement list

Specify whether to put the *apply changes* (those changes that are generated when you set **Generate apply jobs** to Yes) in a work statement list (WSL).

#### Values:

##### YES

The apply changes are generated in a WSL.

##### NO

The apply jobs are generated in a separate data set.

#### Default value:

YES

If you specify YES, you are prompted for additional information after you press Enter, depending on whether the WSL (provided in the **Worklist name** option) already exists:

**If the WSL already exists:**

The **Specify Work Statement List Data Set (ADB2WLDA)** panel prompts you for the data set name. On this panel, you can also indicate whether you want to append to or replace the WSL and whether you want to build a batch job to run the WSL.

```
Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
Worklist name . . . . . : PQ76055N (also used as middle qualifier in DSNs)
Co -----
| DB2 Admin----- Specify Work Statement List Data Set ----- 12:18
| Existing Worklist
|
| Work stmt list dsn . . . WLIST.WSL
Ch| Work stmt list name . . . WLIST
|
Da| Existing name action . . . (Append or Replace)
| Build JCL to run work stmt list . . . (Yes/No)
Op -----
```

Figure 33. **Specify Work Statement List Data Set (ADB2WLDA)** panel

**Restriction:** The replace capability is not supported if you are using the MultiCompare function to compare more than one saved dialog. If you are using MultiCompare, WSLs are automatically appended. To replace WSLs for more than one dialog, you must run the comparisons individually.

If you are appending to an existing WSL, the **Specify Work Statement List (ADB27WLD)** panel prompts you to specify a different middle qualifier to avoid reusing data sets:

```
Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
Worklist name . . . . . : PQ76055N (also used as middle qualifier in DSNs)
Co -----
| DB2 Admin ----- Specify Work Statement List -----
| Existing Worklist
|
| Appending to an existing WSL may generate duplicate dataset names.
Ch| To avoid this, please specify a new middle qualifier.
|
Da| Middle Qualifier . . . . D5787
| -----
```

Figure 34. **Specify Work Statement List (ADB27WLD)** panel

**If the WSL does not exist:**

The **Specify Work Statement List Data Set (GOC5WL)** panel is displayed:

```

Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
Worklist name . . . . . : PQ76055N (also used as middle qualifier in DSNs)
Co ----- Specify Work Statement List Data Set ----- 13:48
| Compare ----- Specify Work Statement List Data Set ----- 13:48
| Enter/verify the following:
| Work stmt list dsn . .
Ch |
| Build JCL to run work stmt list                (Yes/No)
Da -----

```

Figure 35. **Specify Work Statement List Data Set (GOC5WL)** panel

If you specify YES for **Build JCL to run work stmt list**, the **Specify Job Parameters (ADB2W1R)** panel prompts you to specify the job library partitioned data set (PDS) and member prefix:

```

ADB2W1R ----- Specify Job Parameters ----- 09:08

Enter/verify the following:
Generate one job ==> NO      (Yes,No or Per Process)
Job library PDS  ==>
Member prefix   ==> RLS1   (Prefix, max 6 chars)
Jobname = member? ==>
(Yes/No)

```

Figure 36. **Specify Job Parameters (ADB2W1R)** panel

If you specify NO, The **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel prompts you for that data set name after you press Enter. If the data set does not exist, it is created.

```

Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
Worklist name . . . . . : PQ76055N (also used as middle qualifier in DSNs)
Co ----- Specify Data Set Name for Apply Jobs ----- 12:18
| Compare ----- Specify Data Set Name for Apply Jobs ----- 12:18
| Enter/verify the following:
| Data Set Name ==> APPLY.DEF1045
Ch |
|
Da -----

```

### Embed IFF into WSL

Specify whether the WSL is to include the contents of the IFF file in an encoded format. An IFF file is produced if the change requires an unload operation. The benefit of embedding the IFF contents in the WSL is that you can transport the WSL to another system without having to separately transport the IFF file.

#### Values:

##### YES

The IFF file is embedded in the WSL.

##### NO

The WSL does not include the IFF file.

#### Default:

NO

### Use customized util opts

Specify whether you want to use the options that you set for the COPY, CHECK DATA, MODIFY, REBUILD, REORG, RUNSTATS, UNLOAD, and LOAD utilities on the Db2 Administration Tool panels.

#### Values:

##### YES

Utility jobs and WSLs are generated based on the utility options that you specified.

##### NO

The default utility options are used.

#### Related information:

[“UO - Change utility options” on page 109](#)

### Content of apply job(s)

Specify whether to generate only changes to database objects.

#### Values:

##### ALL

All jobs and processes to reload the data are generated.

##### DDL

Only the DDL is generated. Object Comparison Tool does not generate UNLOAD statements, LOAD statements, or other utilities except for rebind and REORG operations that are needed to apply the pending definition changes and remove any restrictive states. These operations are necessary to allow the subsequent statements to be successful.

When DDL is specified, any data conversion errors are ignored and no conversion report is generated.

#### Default value:

ALL

### Unload method

Specify the method that you want to use to unload data.

#### Values:

##### U (Unload)

Use the Db2 UNLOAD utility.

##### P (Parallel unload)

Use the Db2 UNLOAD utility with parallel processing.

Parallel unload cannot be used in the following situations:

- A limit key change
- A change in number of partitions
- The use of an identity column in the partitioning key

If Db2 Object Comparison Tool determines that the operation is not eligible for a parallel unload, it uses **U** (Unload) instead.

If the operation is eligible for a parallel unload, a template is used to allocate the unload data sets.

##### H (HPU)

Use Db2 High Performance Unload for z/OS(HPU) to unload the data. HPU must be available.

#### Default value:

U

If **P** (Parallel Unload) and **H** (HPU) are not valid options for the current unload, Object Comparison Tool automatically uses **U** (Unload).

### Generate templates

Specify whether you want the compare process to generate templates for data sets.

**Values:****YES**

Templates are generated for non-utility data sets with the definitions that you specified in Db2 Administration Tool.

**NO**

The values for the **Prefix for data sets** and **Worklist name** options are used for the data set names.

If the **Take an image copy** or **Run REORG** options are set to be run, the utility templates are used.

**Related information:**

[“TU - Specify TEMPLATE usage” on page 108](#)

[LISTDEFS and TEMPLATEs \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

**Stop on conversion error**

Specify whether you want the compare process to stop if a conversion error occurs.

**Values:****YES**

If a conversion errors occurs, the APPLY job is not generated, and an error message similar to the following message is displayed:

```
Compare table source(aaaaaa.bbbbbbb) and target(aaaaaa.bbbbbbb)
Column COLNAME
Conversion not supported for Col COLNAME (TIME to INTEGER)
(D)Type changed from TIME to INTEGER
(E)This type change is not supported
Tables have identical column lists
Table aaaaaa.bbbbbbb will be dropped
Table will be recreated
Table data conversion jobstep will not be generated
Conversion will fail because of datatype mismatch
Run stopped because conversion(s) not supported
```

**NO**

The compare process does not stop if a conversion error occurs.

**Default value:**

NO

**Use DEFER YES**

Specify whether to use DEFER YES clauses on any eligible CREATE INDEX statements.

**Values:****YES**

DEFER YES is used for eligible indexes. However, any user-defined masks for the DEFER attribute take precedence over the value of this option.

If you specify DEFER YES = YES and Run REORG/REBUILD = A, REBUILD INDEX jobs are generated. The DDL still contains the DEFER YES clause, but that clause is ignored and the indexes are rebuilt.

**NO**

DEFER YES is not used.

**Default value:**

NO

**Allow rotate parts**

Specify whether to generate the ROTATE PARTITION statement or the ALTER PARTITION statement when the condition for a rotation is met.

**Values:****YES**

Generate the ROTATE PARTITION statement. Data from the rotating partitions is unloaded before the rotation. You can discard this data or to load it back into the new partitions.

**NO**

Generate the ALTER PARTITION statement and a REORG statement for the affected partitions. For the **Run REORG/REBUILD** option, specify either MANDATORY or ALL RELEVANT to generate the REORG statements. Data from the rotating partitions is loaded back into the table so that you do not have to manually perform the reload. Logical and physical partitions are preserved.

**Default value:**

YES

If your table is not partitioned, specify YES.

**Retain GENERATED ALWAYS:****For ROWID**

Specify whether to retain the GENERATED ALWAYS attribute for ROWID columns.

**Values:****YES**

Retain the GENERATED ALWAYS attribute.

**NO**

Do not retain the GENERATED ALWAYS attribute.

**Default value:**

NO

**For ROW CHANGE TIMESTAMP**

Specify whether to retain the GENERATED ALWAYS attribute for ROW CHANGE TIMESTAMP columns.

**Values:****YES**

Retain the GENERATED ALWAYS attribute.

**NO**

Do not retain the GENERATED ALWAYS attribute.

**Default value:**

NO

**Retain START and RESTART values:****For sequence object:**

Specify whether to retain START and RESTART values for the sequences.

**Values:****YES**

Retain START and RESTART values.

**NO**

Do not retain START and RESTART values.

If you specify NO and ignores are specified for the START or RESTART fields, the ignored fields are not changed. If you specify NO and ignores are not specified for the START and RESTART fields, the values on the target are changed according to the source.

**Default value:**

NO

**IDENTITY START value**

Specify the value to use for the identity column when the table is re-created.

**Values:****ORIGINAL**

Use the START value for the identity column from the Db2 catalog.

## COMPUTED

Compute the START value based upon the identity attributes of the column.

The computed value is based on the existing identity column attributes, such as MAXASSIGNED and the current cache size, at the time that the DDL is produced. Any changes made after the creation of the DDL, to either the identity values or to the data, are not reflected in the DDL and make the DDL obsolete. Object Comparison Tool does not locate an unassigned value from the existing data, and a new RESTART value might need to be provided based upon the underlying data and the application needs.

### Default value:

ORIGINAL

### Mask ignored fields

Specify whether to apply masked values to ignored fields for newly added objects if the field has been masked and ignored.

### Values:

#### YES

Apply the masked values.

#### NO

The original values from the source are applied.

### Default:

NO

This option is not applicable to ignore files that are provided in the **CM Register Options (ADB2CRO)** panel.

## Optional jobs after Reload or Alter:

### Run CHECK DATA

Specify whether to generate a CHECK DATA utility job for all table spaces that are affected by the LOAD utility jobs that are generated by Object Comparison Tool to reload the data.

### Values:

#### YES

Generate a CHECK DATA job.

#### NO

Do not generate a CHECK DATA job.

### Default value:

YES

**Recommendation:** Specify YES if LOAD uses ENFORCE NO.

### Take an image copy

Specify whether to generate COPY utility jobs.

### Values:

#### R (Reload)

Generate a COPY job for all tables that are affected by the generated LOAD jobs to reload the data.

#### A (Alter)

Generate a COPY job for all table spaces, tables, and indexes that are altered with generated ALTER statements.

#### B (Both)

Generate a COPY job for all tables that are affected by the LOAD job and all altered table spaces, tables, and index objects.

#### N (None)

Do not generate any COPY jobs.

**Default value:**

B

**Run REORG/REBUILD**

Specify whether to generate REORG TABLESPACE utility jobs and REBUILD INDEX utility jobs, if needed. These jobs are run after applying the changes from an object comparison to make the target system operational.

**Values:**

**M (Mandatory)**

Generate all REORG and REBUILD jobs that are needed to remove any REORG-pending and REBUILD-pending states and make the data available.

**A (All relevant)**

Generate all REORG and REBUILD jobs that are needed to fully implement the changes. For example, changing PRIQTY is registered when the table space is altered, but the new value is not used until the table space is reorganized.

**N (None)**

Do not generate any REORG and REBUILD jobs. **NONE** is not valid if you specified NO for **Allow rotate parts**.

**Default value:**

A

**Run RUNSTATS**

Specify whether to generate RUNSTATS utility jobs.

**Values:**

**R (Reload)**

Generate a RUNSTATS job for all tables that are affected by the generated LOAD jobs to reload the data.

**A (Alter)**

Generate a RUNSTATS job for all table spaces, tables, and indexes that are altered with generated ALTER statements.

**B (Both)**

Generate a RUNSTATS job for all tables that are affected by the LOAD job and all altered table spaces, tables, and index objects.

**M (Min)**

Generate RUNSTATS jobs for the following conditions:

- If a table space is dropped and recreated, generate RUNSTATS statements for the tables and indexes.
- If a table is dropped and recreated, generate RUNSTATS statements for only the indexes and not the table.
- If an index is created, recreated, or had columns added, generate RUNSTATS statements for the index.
- If the index is created with DEFER YES and REBUILD is generated, the RUNSTATS operation is performed after the REBUILD operation.

**N (None)**

Do not generate any RUNSTATS jobs.

**Default value:**

B

**Run REBIND**

Specify whether to generate a job to rebind the plans and packages that are affected by the changes from an object comparison.

**Values:**

**M (Mandatory)**

Generate a REBIND job for only those plans and packages that were invalidated by the changes.

**A (All relevant)**

Generate a REBIND job for all plans and packages that were affected by the changes, including those plans and packages that were invalidated.

**N (None)**

Do not generate a REBIND job.

**Default value:**

A

**REBIND options**

Indicate whether you want to specify your own BIND options for any plans and packages that are rebound. To use this field, you must specify M (Mandatory) or A (All relevant) in the **Run REBIND** field.

**Values:**

**YES**

Allows you to specify BIND options on the subsequent **REBIND options (ADBPREBO)** panel.

**NO**

Any affected plans or packages are rebound with their existing BIND options (the options that were used during the previous bind or rebind operation).

**Default value:**

NO

If you specify YES, the **REBIND options (ADBPREBO)** panel is displayed after you press Enter:

```

ADBPREBO ----- REBIND options -----
Command ===>

Specify additional REBIND parameters to generate rebinds
for dependent packages.

APREUSE . . . . . (None, Warn, Error)
EXPLAIN . . . . . (Yes, No, Only)
OWNER . . . . . > (Owner of package)
OWNERTYPE . . . . (Role, User)

Additional options: -----
-----
-----
-----

```

Figure 37. **REBIND options (ADBPREBO)** panel

When this panel is displayed, specify any BIND options for dependent packages, and press Enter:

**Note:** Any BIND options that you specify in the **Additional options** field are added to the REBIND statement as is; they are not validated.

**Commands**

You can use the commands listed at the bottom of this panel to specify additional job options:

- BP - Change batch job parameters
- TU - Specify TEMPLATE usage
- UO - Customize utility options
- CO - Change options common to change functions
- ADV - Advanced options
- SIM - Simplified options

Figure 38. Commands on the **Generate Compare Jobs (GOC5)** panel

**Note:** Either ADV or SIM is listed depending on your display mode.

You can enter any of these commands on the command line:

### BP - Change batch job parameters

Allows you to change the parameters for batch utility jobs, such as the job card and space parameters.

When you specify the BP command and press Enter, the **Batch Job Utility Parameters (ADB2UPA)** panel is displayed:

```

DB2 Admin ----- DD1A Batch Job Utility Parameters ----- 11:02
Command ==>

Generate Job Card . ==>      (Yes/No)                DB2 System: DD1A
Job cards:                   DB2 SQL ID: ADM001
  ==> //JD4678SD JOB , 'DB2 UTILITY',
  ==> //          REGION=8M, NOTIFY=USER1,
  ==> //          MSGCLASS=X,
  ==>
  ==>
Generate Job CLASS ==> YES  (Yes/No)    JOB CLASS . . . . . ==>

JOBPARM:
  ==>
  ==>
  ==>
  ==>
CM Batch EXEC statement parameters:
Add SSID parameter . . YES    (Yes/No)
Add PLAN parameter . . YES    (Yes/No)
Additional parameters to add to CM Batch JCL EXEC statement:
  ==>
  ==>
  ==>
ADBTEP2:
Restart . . . . . (Yes/No)
Maxerrors . . . . . 88 (-1 to 99)
BindError . . . . . IGNORE (MAXE, Save or Ignore)
Log DIAG . . . . . YES (Yes/No)
AutoCheck . . . . . YES (Yes/No)
LOAD Summary Report YES (Yes/No)
Auto Rebuild . . . . . YES (Yes/No)
Auto Reorg . . . . . YES (Yes/No)
Advisory Auto Rebuild YES (Yes/No)
Advisory Auto Reorg YES (Yes/No)
LOB/XML IC Unload . . U (Error, Use base data)
Missing IC Unload . . U (Error, Use base data)
Spanned . . . . . (Yes/No)
DB2 Pending Changes options:
  Check at DROP . . . NO (Yes/No)

Space parameters:
Unit name           ==> SYSDA
Space unit . . . . ==> TRK (BLK, TRK, CYL or 4096-32760)
Max Primary . . . . ==> 65535 (In above units or 99999999 or blank)
                                     In KB: 3145680
Max DASD . . . . . ==> 65535 (In above units. Allocations beyond this
                                     are sent to tape) In KB: 3145680
Tape Unit . . . . . ==> TAPE (Unit for tape if size is greater
                                     than Max DASD)
Default space allocation if unable to calculate:
Primary alloc . . . ==> 30 (in above units)
Secondary alloc . . ==> 30 (in above units)

Function-specific parameters:
Unload pct . . . . ==> 0 (0-99 - % increase for converted data set)

```

## Related information:

[Batch job parameters for utility jobs \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

### TU - Specify TEMPLATE usage

Allows you to modify templates for the data sets that are allocated and used by Db2 Object Comparison Tool.

When you specify the TU command and press Enter, the **Specify UTILITY TEMPLATE Usage (ADB25TU3)** panel is displayed:

```
ADB25TU3          DD1A Specify UTILITY TEMPLATE Usage          11:58
Command ==>>>

Line commands:
T - Toggle Use On/Off   C - Clear data   ? - Choose Template for the Keyword
E - Edit Template
Template type           ==>> OC           (UTIL, ALT, MIG, RES, OC)
Generate template statements ==>> NO     (Yes/No)
Sel Keyword           Use Template Comment
-----
                                More:      +
GOCALTR
GOCCREA
GOCDROP
GOCRBND
GOCIFFN
GOCSHVR
GOCCHNG
```

From this panel, you can modify the templates for data sets. The default work data sets and descriptions are shown in the following table:

Table 7. Work data set descriptions

Template keyword	Default data set	Description
GOCALTR	<i>prefix.worklist.DDL.ALTER</i>	Primarily ALTER statements
GOCCREA	<i>prefix.worklist.DDL.CREATE</i>	Primarily CREATE statements
GOCDROP	<i>prefix.worklist.DDL.DROP</i>	Primarily DROP statements
GOCRBND	<i>prefix.worklist.CMD.REBIND</i>	REBIND control statements
GOCIFFN	<i>prefix.worklist.IFF</i>	Internal version file
GOCSHVR	<i>prefix.worklist.SHRVARS</i>	ISPF variables
GOCCHNG	<i>prefix.worklist.CHANGES</i>	Changes from compare

You can specify the following variables in templates:

- The following functional variables:

#### **&GOCPRE**

The prefix for data sets, which you specify on the **Generate Compare Jobs (GOC5)** panel

#### **&GOCWLN**

The statement work list name, which you specify on the **Generate Compare Jobs (GOC5)** panel

- The date and time variables that are supported for the Db2 TEMPLATE utility.
- &USERID

## Related information:

[Associating templates with data sets \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[Syntax and options of the TEMPLATE control statement \(Db2 13 for z/OS\)](#)

## UO - Change utility options

Allows you to specify options for Db2 utilities.

When you specify the UO command and press Enter, the **Change Utilities Options (ADB2UOPS)** panel is displayed:

```
ADB2UOPS ----- DD1A Change Utilities Options ----- 11:15
Select one of the following, then press Enter.

  C - Image copy
  KD - Check data
  M - Modify
  O - Reorg tablespace
  OI - Reorg index
  RB - Rebuild index
  R - Runstats tablespace
  U - Unload
  L - Load

Option
====>
```

On this panel, you can select the Db2 utility for which you want to change the options. When you press Enter, the **Specify Utility Options** panel for the selected utility is displayed, and you can enter the options that you want. Press Enter to save your selections.

**Restriction:** Some utility options are not available for utility jobs that are built by Db2 Object Comparison Tool.

For Db2 Object Comparison Tool to generate utilities with the options that you specified, on the **Generate Compare Jobs (GOC5)** panel, you must set the **Use utility options** to YES. The options that you select are retained and used for any subsequent jobs where **Use utility options** is set to YES.

### Related information:

[Running Db2 utilities from Db2 Admin Tool \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## CO - Change options common to change functions

Allows you to review and change options that are common to change functions in Db2 Administration Tool and Db2 Object Comparison Tool.

When you specify the CO command and press Enter, the **Options for Change Functions (ADB2PCO)** panel is displayed:

```

ADB2PCO n                               Options for Change Functions                               14:19
Command ==>

                                                                 DB2 System: DD1A

Recreate accelerated tables . . . . . YES (Yes/No. Default is Yes)
Restore replication of tables . . . . . YES (Yes/No. Default is Yes)
Reload accelerated tables . . . . . YES (Yes/No. Default is Yes)
Restore acceleration of tables . . . . . YES (Yes/No. Default is Yes)
Remove deleted accelerated tables . . YES (Yes/No. Default is Yes)

Load accelerated tables LOCKMODE . . . NONE (Default is TABLESET)
Load accelerated tables DETECTCHANGES DATA (Default is DATA)
Unload altered tables . . . . . NO (Yes/No/Des. Default is YES)
Preserve all data . . . . . YES (Yes/No. Default is YES)

Enable WSL authorization switching . . NO (Yes/No. Default is No)
Object processing order . . . . . H (T - Object type, H - DB hierarchy.
                                     Default is H)
Statement validation exit name . . . . (Name of EXEC used to validate
                                     statements in WSL Validate)

Allow PBR2 to PBR changes . . . . . NO (Yes/No. Default is No)
DB2 release number . . . . . 1215 (Use VVRM format)
DB2 function level . . . . . 504 (E.g. 100, 500, 501, 5nn)
GRANT processing order . . . . . C (C - CREATE prefix for GRANT
                                   P - POSTUTIL prefix for GRANT
                                   Default is C )

Auto view regenerate . . . . . NO (Yes/No. Default is No)
Ignore source column order . . . . . NO (Yes/No. Default is No)
Generate hidden ROWID columns . . . . NO (Always, Only, No. Default is No)

```

**ADV - Advanced options**

Toggles the display mode to show all compare job options.

**SIM - Simplified options**

Toggles the display mode to show only the basic compare job options.

**Related information**

[Video - Db2 Object Comparison Tool: Simplified job options](#)

## Compare reporting options

If **Change reporting options** = YES on the **Generate Compare Jobs (GOC5)** panel, you can change the compare report options on a subsequent panel (**Specify Compare Reporting Options (GOC5R0)** panel) before running the comparison.

You can specify the following reporting options:

```

GOC5R0 ----- Specify Compare Reporting Options ----- 12:20

Report options for Compare:
Only changed objects . . . YES (Yes/No)
Ignore fields:
  User specified . . . . . YES (Yes/No)
  System generated . . . . YES (Yes/No)
  Object specific . . . . . YES (Yes/No)
Translation masks . . . . . YES (Yes/No)
Summary report . . . . . YES (Yes/No)
Object count report . . . . YES (Yes/No)
Conversion report . . . . . YES (Yes/No)

```

Figure 39. **Specify Compare Reporting Options (GOC5R0)** panel

**Only changed objects**

Specify whether the detailed report is to include only those objects that have changed.

**Ignore fields:**

**User specified**

Specify whether the report is to include the names of user-specified ignore fields.

**System generated**

Specify whether the report is to include the names of system ignore fields.

**Object Specific**

Specify whether the report is to include the names of fields that are ignored for specific objects.

**Related information:**

[“4. Specifying ignores” on page 80](#)

**Translation masks**

Specify whether the report is to include the translation masks that are used by the compare job.

**Related information:**

[“Translation masks” on page 73](#)

**Summary report**

Specify whether the report is to include a summary, which consists of one line per object.

**Object count report**

Specify whether the report is to include statistics of compared and changed objects.

**Conversion report**

Specify whether to report expected conversion problems for tables when a change is run.

The following example shows the corresponding strings for the parameters that are passed to step T03COMP PGM=GOC2CMP if you specify YES for the fields on panel GOC5RO. Specifying Yes for both **User specified** and **System generated** results in REPIGALL being used as the parameter.

```

Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
Worklist name . . . . . : AAAAAAA (also used as middle qualifier in DSNs)
Co ----- Specify Compare Reporting Options ----- 12:18
|
| Report options for Compare:
| Only changed objects . . : REPCHG
Ch| Ignore fields:
|   User specified . . . . : REPIGUSR
|   Object Specific . . . . : Yes      (Yes/No)
Da|   System generated . . . . : REPIGSYS
|   Translation masks . . . : REPMASK
|   Summary report . . . . : REPSUM
Op|   Object count report . . : REPCOUNT
|   Conversion report . . . : REPCONV
|
|-----|

```

Figure 40. Example of the **Specify Compare Reporting Options (GOC5RO)** panel with REPIGALL used as the PARM options.

## Generating a compare batch job to make changes by using Change Management

Registering changes in Change Management (CM) simplifies the process of recording and tracking the changes that you make to your Db2 objects.

**Before you begin**

Change Management must be enabled on the system and be either optional or required for your SQL ID. You enable Change Management during the customization of Db2 Admin Tool.

This procedure is a subtask of “5. Generating a compare job” on page 84. You must have identified the source and target objects and optionally any masks and ignores and have the **Generate Compare Jobs (GOC5)** panel displayed.

## About this task

You can register the change on multiple target locations. In addition, you can specify an optional target profile as you register the multi-target change.

## Procedure

To generate a compare batch job to make changes by using Change Management:

1. On the **Generate Compare Jobs (GOC5)** panel, specify values for the compare job options as follows, and press Enter:
  - Set **Generate apply jobs** to YES.
  - Specify values for the other options as needed. See “Compare job options” on page 87.

### Tips:

- To view only the minimal number of options that you need to set to run a comparison, use the simple display mode. You can toggle between simple and advance mode by using the SIM and ADV commands. For more information about these display modes, see “Simple and advanced display modes” on page 89.
- Set **Save compare results** to YES if you want to analyze data about the comparison, ignore changes, or increase the efficiency of subsequent comparisons. The saved compare results contain information about objects that were part of the comparison, including detected differences, changes to make, and how those changes are to be implemented. You can save the compare results only for tables, indexes, global variables, and distinct data types.

After you press Enter, one of the following panels is displayed:

- If CM is mandatory for your SQL ID, the **CM Register Options (ADB2CRO)** panel is displayed.
  - If CM is optional for your SQL ID, the **Change Management Prompt (ADB2CMRO)** panel is displayed:
2. If the **Change Management Prompt (ADB2CMRO)** panel is displayed, Specify YES to process the compare change through Change Management, and press Enter.

```

Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
  Worklist name . . . . . PQ76055N (also used as middle qualifier in DSNs)
Co. -----
| DB2 Admin ----- DSN8 Change Management Prompt ----- 09:17 |
| Change Management is optional for SQLID:  VNRJJP                |
Ch| Do you wish to use Change Management for this function: YES (Yes/No)
Da|
|-----|

```

Figure 41. **Change Management Prompt (ADB2CMRO)** panel

3. On the **CM Register Options (ADB2CRO)** panel, specify the following information:
  - An owner and a name for the change. The default owner is the current SQL ID. The name of the change cannot contain an apostrophe (or single quotation mark).
  - Whether you want to register the change on multiple target locations. If you specify YES, you can also optionally specify a target profile.

- Optionally, a comment for the change, whether to replace existing changes, an ignore for the change, and a mask for the change.

```

ADB2CRO n ----- CM - Register Options ----- 11:27
Command ==>

Commands: NEXT                                DB2 System: DD1A
                                                DB2 SQL ID: ADM001

Specify the following values to register a change:

Owner . . . . . ADM001                > (Optional, Default is ADM001)
Name . . . . . change1                >
Comment . . . . .                      >
Multi-target change . YES              (Yes/No, Default is No)
  Target name . . . . DB2X_FILE        > (Optional, ? to lookup)
  Group name . . . . .                  > (Optional, ? to lookup)

Replace existing change . .             ('/' to replace, Default is BLANK)

Specify the owner and name values to use for this change (? to lookup):
      Owner      Name
Ignore . . . . . >
Mask . . . . . >

```

Figure 42. **CM Register Options (ADB2CRO) panel**

- Issue the NEXT command, and press Enter.

The change is registered as a normal change.

- Complete the requested input on any subsequent panels that are displayed.

One or more of the following panels might be displayed depending on the compare job options that you selected. For more information about one of these panels, see the related option or panel description.

Panel	Related option on Generate Compare Jobs (GOC5) panel or panel description
<b>Save Compare Results (ADB2C22) panel</b>	<a href="#">“Save compare results” on page 94</a>
<b>Specify Compare Reporting Options (GOC5RO) panel</b>	<a href="#">“Compare reporting options” on page 110</a>
<b>Specify Data Set Name for Apply Jobs (GOC5AJ) panel</b>	<a href="#">“Generate apply jobs” on page 97</a>
<b>Specify Work Statement List Data Set (ADB2WLDA) panel</b>	<a href="#">“As work statement list” on page 98</a>
<b>Specify Work Statement List (ADB27WLD) panel</b>	<a href="#">“As work statement list” on page 98</a>
<b>Specify Work Statement List Data Set (GOC5WL) panel</b>	<a href="#">“As work statement list” on page 98</a>
<b>Specify Job Parameters (ADB2W1R) panel</b>	<a href="#">“As work statement list” on page 98</a>
<b>REBIND options (ADBPREBO) panel</b>	<a href="#">“REBIND options” on page 106</a>
<b>DB2 Object Compare Warning (GOCGMPW) panel</b>	<a href="#">“Generate apply jobs” on page 97</a>

- On the **Specify Register Mode (GOC5RM) panel** specify one of the following actions to take for any pending changes to the objects on the target system that are affected by this change, and press Enter:

**Cancel**

Do not register the change if pending changes exist.

**Prereq**

Make the pending changes for the affected objects prerequisite changes for this change.

## Supersede

Make this change a prerequisite change for the pending changes.

```
Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
Worklist name . . . . . PQ76055N (also used as middle qualifier in DSNs)
Co. -----
| Compare ----- Specify Register Mode ----- 09:23
| Pending changes action . . . (Cancel, Prereq, Supersede)
Ch|
Da|
|-----
```

Figure 43. **Specify Register Mode (GOC5RM)** panel

7. If you requested a batch job (the **Generate online** option is set to NO), edit the generated JCL job as needed and submit it to run the comparison. Otherwise (if **Generate online** is set to YES), the comparison process is run online.

For information about the Object Comparison Tool parameters in the generated JCL job, see [“Parameters in the generated compare batch job”](#) on page 117.

## Results

The change to apply the compare changes is registered. After the compare batch job is run, you can use Db2 Admin Tool to analyze and run the change. These actions apply the changes from the comparison.

### Related information

[Analyzing a change \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[Running a change \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## Generating a compare batch job for a multi-target import

A *multi-target import* is the process of importing changes from a compare job to objects on multiple target environments. When you generate a batch job for this import, register the change with Change Management (CM) so that you can later import the change to objects on multiple target environments.

### Before you begin

This procedure is a subtask of “5. Generating a compare job” on page 84. You must have identified the source and target objects and optionally any masks and ignores and have the **Generate Compare Jobs (GOC5)** panel displayed.

### About this task

As part of this task of importing changes from a compare job to objects on multiple target environments, you can specify masks for the compare job on one or more of the following panels:

#### Specify Compare Masks (GOC3) panel

Specify masking on this panel if your compare source and target object use different naming conventions.

#### CM Register Options (ADB2CRO) panel

Specify masking on this panel if your multi-target change uses different naming conventions than your compare target.

## CM - Update Associated Target panel (ADBPCMTU) panel

Specify masking on this panel if your multiple target systems use different naming conventions than your multi-target change.

The masking that you specify on one panel does not override the masking that you specify on another panel. All specified masks are applied.

## Procedure

To generate a compare batch job for a multi-target import:

1. On the **Generate Compare Jobs (GOC5)** panel, specify values for the compare job options as follows, and press Enter:

- Set **Generate Online** to NO.
- Specify values for the other options as needed. See [“Compare job options” on page 87](#).

### Tips:

- To view only the minimal number of options that you need to set to run a comparison, use the simple display mode. You can toggle between simple and advance mode by using the SIM and ADV commands. For more information about these display modes, see [“Simple and advanced display modes” on page 89](#).
- Set **Save compare results** to YES if you want to analyze data about the comparison, ignore changes, or increase the efficiency of subsequent comparisons. The saved compare results contain information about objects that were part of the comparison, including detected differences, changes to make, and how those changes are to be implemented. You can save the compare results only for tables, indexes, global variables, and distinct data types.

After you press Enter, one of the following panels is displayed:

- If CM is mandatory for your SQL ID, the **CM Register Options (ADB2CRO)** panel is displayed.
  - If CM is optional for your SQL ID, the **Change Management Prompt (ADB2CMRO)** panel is displayed:
2. If the **Change Management Prompt (ADB2CMRO)** panel is displayed, specify YES, and press Enter.
  3. On the **CM Register Options (ADB2CRO)** panel, specify a name for the change and set **Multi-target change** to YES. You can also optionally specify a mask.

```
ADB2CRO n ----- CM - Register Options ----- 16:25
Command ==>

Commands: NEXT                                DB2 System: DD1A
                                                DB2 SQL ID: ADM001

Specify the following values to register a change:

Owner . . . . . ADM001                        > (Optional, Default is ADM001)
Name . . . . . TEST2                          >
Comment . . . . .                               >
Multi-target change . YES                     (Yes/No, Default is NO)
  Target name . . . . PSVTEST                  > (Optional, ? to lookup)
  Group name . . . . .                          > (Optional, ? to lookup)

Replace existing change . .                    ('/' to replace, Default is BLANK)

Specify the owner and name values to use for this change (? to lookup):
                                Owner          Name
Ignore . . . . .                   >
Mask . . . . .                       >
```

4. Issue the NEXT command, and press Enter.
5. If the **Insert a Target (ADBPC911)** panel is displayed, complete the fields to add a target, and press Enter.

For help on defining targets for Change Management, see [Setting up the targets \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

- On the **Associate Targets (ADBPCMT)** panel, use the line commands to add and edit targets as needed.

```

DB2 Admin ----- CM - Associate Targets ----- Row 1 to 1 of 1
Command ==> Scroll ==> CSR

Details for multi-target change: PEDRO.TEST1          DB2 System: DD1A
                                                       DB2 SQL ID: ADM001

Commands: NEXT
Line commands:
  U - Update  D - Delete  AT - Add targets  AG - Add targets from group
  I - Interpret ? - Show all line
commands

      Target
Sel Name  DB2 Location  Change  Change  Status
      *      *          *      name     *
----->----->----->----->----->----->----->
      PSVTEST  DBAD
***** END OF DB2 DATA *****

```

To update an existing target, specify the U line command next to the target. Then, on the **CM - Update Associated Target panel (ADBPCMTU)** panel, under **Target Overrides:**, specify new values for the change owner, change name, mask owner, and mask name:

```

ADBPCMTU n ----- CM - Update Associated Target ----- 10:05
Command ==>

Press Enter to confirm changes.

Name . . . . . : PSVTEST
DB2 location . . . . . : DBAD
Target Defaults:
  Mask owner . . . . . : USER02
  Mask name . . . . . : MASKUSR2 >
Target Overrides:
  Change owner . . . . . PSV01 >
  Change name . . . . . TEST2 >
  Mask owner . . . . . USER02 >
  Mask name . . . . . MASKUSR2 >

```

Figure 44. **CM - Update Associated Target panel (ADBPCMTU)** panel

Then press Enter to save your changes. If you exit (PF3), the values are not saved.

- On the **Associate Targets (ADBPCMT)** panel, issue the NEXT command, and press Enter.
- Complete the requested input on any subsequent panels that are displayed.

One or more of the following panels might be displayed depending on the compare job options that you selected. For more information about one of these panels, see the related option or panel description.

Panel	Related option on Generate Compare Jobs (GOC5) panel or panel description
<b>Save Compare Results (ADB2C22)</b> panel	<a href="#">“Save compare results” on page 94</a>
<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<a href="#">“Compare reporting options” on page 110</a>
<b>Specify Data Set Name for Apply Jobs (GOC5AJ)</b> panel	<a href="#">“Generate apply jobs” on page 97</a>
<b>Specify Work Statement List Data Set (ADB2WLDA)</b> panel	<a href="#">“As work statement list” on page 98</a>
<b>Specify Work Statement List (ADB27WLD)</b> panel	<a href="#">“As work statement list” on page 98</a>

Panel	Related option on Generate Compare Jobs (GOC5) panel or panel description
<b>Specify Work Statement List Data Set (GOC5WL)</b> panel	<a href="#">“As work statement list” on page 98</a>
<b>Specify Job Parameters (ADB2W1R)</b> panel	<a href="#">“As work statement list” on page 98</a>
<b>REBIND options (ADBPREBO)</b> panel	<a href="#">“REBIND options” on page 106</a>
<b>DB2 Object Compare Warning (GOCGMPW)</b> panel	<a href="#">“Generate apply jobs” on page 97</a>

9. On the **Specify Register Mode (GOC5RM)** panel specify one of the following actions to take for any pending changes to the objects on the target system that are affected by this change, and press Enter:

**Cancel**

Do not register the change if pending changes exist.

**Prereq**

Make the pending changes for the affected objects prerequisite changes for this change.

**Supersede**

Make this change a prerequisite change for the pending changes.

```

Compare ----- Generate Compare Jobs -----
Option ==>

Specify the following for DB2 Object Comparison Tool:

Worklist information:
Worklist name . . . . . PQ76055N (also used as middle qualifier in DSNs)
Co.-----
| Compare ----- Specify Register Mode ----- 09:23 |
| Pending changes action . . . (Cancel, Prereq, Supersede) |
|-----|
Ch|
Da|
|-----|

```

Figure 45. **Specify Register Mode (GOC5RM)** panel

10. If you requested a batch job (the **Generate online** option is set to NO), edit the generated JCL job as needed and submit it to run the comparison. Otherwise (if **Generate online** is set to YES), the comparison process is run online.

For information about the Object Comparison Tool parameters in the generated JCL job, see [“Parameters in the generated compare batch job” on page 117](#).

## Parameters in the generated compare batch job

When you generate a compare batch job, the resulting JCL includes Object Comparison Tool parameters. Those parameters correspond to the following panel options:

Parameter in the JCL job	Panel	Corresponding option
ACCLOCKMODE	<b>Options for Change Functions (ADB2PCO)</b> panel	<b>Load accelerated tables lock mode</b> See <a href="#">“CO - Change options common to change functions” on page 109</a> .

Parameter in the JCL job	Panel	Corresponding option
ALTPART	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Allow rotate parts</b> See <a href="#">Allow rotate parts</a>
APPCONT	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Content of apply jobs(s)</b> See <a href="#">“Content of apply job(s)” on page 101.</a>
AUTHSQL	<b>ALTER - Build Analyze and Apply Job (ADBPALT)</b> panel	<b>Authorization Switch ID</b>
AUTHSW	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Enable auth-switching</b> See <a href="#">“Enable auth-switching” on page 93.</a>
CMDDL	None	None CMDDL does not correspond to a panel option. CMDDL specifies whether to comment out the ADMIN ALTER IMPLICIT statements.
CMDELTA	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Generate Apply Job = CHANGE</b> See <a href="#">Change.</a>
CMRACT	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Save compare results</b> See <a href="#">“Save compare results” on page 94.</a>
CMRADEL	<b>Save Compare Results (ADB2C22)</b> panel	<b>Eligible for auto-delete</b>
CMPRCOMM	<b>Save Compare Results (ADB2C22)</b> panel	<b>Comment</b>
CMPRNAME	<b>Save Compare Results (ADB2C22)</b> panel	<b>Name</b>
CMROWN	<b>Save Compare Results (ADB2C22)</b> panel	<b>Owner</b>
DACVE	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Stop on conversion error</b> See <a href="#">“Stop on conversion error ” on page 102.</a>
DISOPTRE	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Disable REORG optimization</b> See <a href="#">“Disable REORG optimization” on page 94.</a>
DROP_FKS_NOT_IN_SOURCE	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Drop FKs not in source</b> See <a href="#">“Drop FKs not in source” on page 91.</a>
DROPEXOBJ	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Allow implicit drop of excluded objects</b> See <a href="#">“Allow implicit drop of excluded objects ” on page 93.</a>

Parameter in the JCL job	Panel	Corresponding option
ENACCAT	Options for Change Functions (ADB2PCO) panel	<b>Restore acceleration of tables</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.
ENREPAT	Options for Change Functions (ADB2PCO) panel	<b>Restore replication of tables</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.
GENROWID	Options for Change Functions (ADB2PCO) panel	<b>Generate hidden ROWID columns</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.
GRANTORD	Options for Change Functions (ADB2PCO) panel	<b>Table GRANT processing order</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.
GRTSQLID	Generate Compare Jobs (GOC5) panel	<b>Object Grantor</b> See <a href="#">“Object Grantor”</a> on page 92.
ICSPECNAME	Specify Compare Ignores (GOC4) panel	<b>Ignore Changes Specification: Name</b> See <a href="#">“Modifying ignore changes specifications”</a> on page 152.
ICSPECOWN	Specify Compare Ignores (GOC4) panel	<b>Ignore Changes Specification: Owner</b> See <a href="#">“Modifying ignore changes specifications”</a> on page 152.
IDENTSVL	Generate Compare Jobs (GOC5) panel	<b>IDENTITY START value</b> See <a href="#">“IDENTITY START value”</a> on page 103.
IGNCOLORD	Options for Change Functions (ADB2PCO) panel	<b>Ignore source column order</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.
KEEPCOL	Generate Compare Jobs (GOC5) panel	<b>Suppress DROP of columns</b> See <a href="#">“Suppress DROP of columns”</a> on page 92.
KEEPTGT	Generate Compare Jobs (GOC5) panel	<b>Suppress DROP of objects</b> See <a href="#">“Suppress DROP of objects”</a> on page 91.
MASKIGN	Generate Compare Jobs (GOC5) panel	<b>Mask ignored fields</b> See <a href="#">“Mask ignored fields”</a> on page 104.
NONEWCOL	Generate Compare Jobs (GOC5) panel	<b>Suppress adding columns</b> See <a href="#">“Suppress adding columns”</a> on page 92.
PBR2TOPBR	Options for Change Functions (ADB2PCO) panel	<b>Allow PBR2 to PBR changes</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.

Parameter in the JCL job	Panel	Corresponding option
PROCORD	<b>Options for Change Functions (ADB2PCO)</b> panel	<b>Object Processing order</b> See “ <a href="#">CO - Change options common to change functions</a> ” on page 109.
REBIND_APLJOB	None	None  REBIND_APLJOB does not correspond to a panel option. REBIND_APPLJOB specifies whether the REBIND parameters need a trailing hyphen in the apply job format in the resulting JCL when not using a WSL for the change.
REBIND_APREUSE	<b>REBIND options (ADBPREBO)</b> panel	<b>APREUSE</b> See <a href="#">Figure 37</a> on page 106.
REBIND_EXPLAIN	<b>REBIND options (ADBPREBO)</b> panel	<b>EXPLAIN</b> See <a href="#">Figure 37</a> on page 106.
REBIND_OWNER	<b>REBIND options (ADBPREBO)</b> panel	<b>OWNER</b> See <a href="#">Figure 37</a> on page 106.
REBIND_OWNERTYPE	<b>REBIND options (ADBPREBO)</b> panel	<b>OWNERTYPE</b> See <a href="#">Figure 37</a> on page 106.
REBIND_ADD_OPTS	<b>REBIND options (ADBPREBO)</b> panel	<b>Additional options</b> See <a href="#">Figure 37</a> on page 106.
REBINDA	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Run REBIND=All</b> relevant See <a href="#">All relevant</a> .
REBINDM	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Run REBIND=Mandatory</b> See <a href="#">Mandatory</a> .
REBINDN	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Run REBIND=None</b> See <a href="#">None</a> .
RECOVER	<b>Generate Analyze Job (ADB2C11A)</b> panel	<b>Data to recover</b> This parameter is for Change Management use only. See <a href="#">Analyzing a change (IBM Db2 Administration Tool for z/OS 13.1.0)</a> .
RECREAT	<b>Options for Change Functions (ADB2PCO)</b> panel	<b>Recreate accelerated tables</b> See “ <a href="#">CO - Change options common to change functions</a> ” on page 109.
RELOADAT	<b>Options for Change Functions (ADB2PCO)</b> panel	<b>Reload accelerated tables</b> See “ <a href="#">CO - Change options common to change functions</a> ” on page 109.

Parameter in the JCL job	Panel	Corresponding option
REMOVEAT	<b>Options for Change Functions (ADB2PCO)</b> panel	<b>Remove deleted accelerated tables</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.
REORGA	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Run REORG/REBUILD=All</b> relevant See <a href="#">All relevant</a> .
REORGM	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Run REORG/REBUILD=Mandatory</b> See <a href="#">Mandatory</a> .
REPALL	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	All possible reporting options are set to yes, except REPCHG. See <a href="#">“Reporting options:”</a> on page 94.
REPCHG	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Only changed objects</b> See <a href="#">“Only changed objects”</a> on page 110.
REPCONV	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Conversion report</b> See <a href="#">“Conversion report”</a> on page 111.
REPCOUNT	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Object count report</b> See <a href="#">“Object count report”</a> on page 111.
REPIGALL	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	All <b>Ignore fields</b> reporting options are set to YES. See <a href="#">Ignore fields</a> .
REPIGOSI	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Ignore fields: Object Specific</b> See <a href="#">Ignore fields: Object Specific</a> .
REPIGSYS	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Ignore fields: System generated</b> See <a href="#">Ignore fields: System generated</a> .
REPIGUSR	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Ignore fields: User specified</b> See <a href="#">Ignore fields: User specified</a> .
REPMASK	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Translation masks</b> See <a href="#">“Translation masks”</a> on page 111.
REPSUM	<b>Specify Compare Reporting Options (GOC5RO)</b> panel	<b>Summary report</b> See <a href="#">“Summary report”</a> on page 111.
RIDALWYS	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Retain GENERATED ALWAYS: For ROWID</b> See <a href="#">Retain GENERATED ALWAYS: For ROWID</a> .
RPTEXOBJ	<b>CM - Manage Exclude Specifications (ADBPC7)</b> panel	Excluded objects See <a href="#">Creating and managing exclude specifications (IBM Db2 Administration Tool for z/OS 13.1.0)</a> .

Parameter in the JCL job	Panel	Corresponding option
RPTXSPEC	<b>CM - Manage Exclude Specifications (ADBPC7)</b> panel	Exclude specifications See <a href="#">Creating and managing exclude specifications (IBM Db2 Administration Tool for z/OS 13.1.0)</a> .
SCOPEWARN	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Scope Warning Messages</b> See <a href="#">“Scope Warning Messages”</a> on page 94.
SEQSRVL	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Retain START and RESTART values for sequence object:</b> See <a href="#">Retain START and RESTART values for sequence object</a> .
SRCIESPECNAME	<b>Specify Compare Source (GOC1)</b> panel	<b>Exclude Specification: Name</b> See <a href="#">“1. Specifying source objects”</a> on page 56.
SRCIESPECOWN	<b>Specify Compare Source (GOC1)</b> panel	<b>Exclude Specification: Owner</b> See <a href="#">“1. Specifying source objects”</a> on page 56.
TGTIESPECNAME	<b>Specify Compare Target (GOC1)</b> panel	<b>Exclude Specification: Name</b> See <a href="#">“2. Specifying target objects”</a> on page 68.
TGTIESPECOWN	<b>Specify Compare Target (GOC1)</b> panel	<b>Exclude Specification: Owner</b> See <a href="#">“2. Specifying target objects”</a> on page 68.
TMSALWYS	<b>Generate Compare Jobs (GOC5)</b> panel	<b>Retain GENERATED ALWAYS: For ROW CHANGE</b> See <a href="#">Retain GENERATED ALWAYS: For ROW CHANGE</a> .
UNLDALTB	<b>Options for Change Functions (ADB2PCO)</b> panel	<b>Unload Altered tables</b> See <a href="#">“CO - Change options common to change functions”</a> on page 109.

## Chapter 5. Checking the compare report

After you run a comparison, check the compare report to see the differences between the source and target objects.

If you want to change the target objects to match the source objects, first ensure that the compare report contains only the changes that you want to apply. You might need to create additional masks and ignore fields and repeat the comparison process until the report contains only the changes that you want.

### Procedure

To check the compare report:

1. Navigate to the compare report in the REPORT data set by using one of the following facilities:

#### SDSF

In SDSF, view the data sets for your job:

```
Display Filter View Print Options Search Help
-----
ISFPCU41 US DISPLAY ALL CLASSES                LINE 1-2 (2)
COMMAND INPUT ===>                            SCROLL ===> PAGE
NP  JOBNAME JobID   Owner   Prty Queue      C   Pos SAff ASys Status
    TS6462  T0243255 TS6462   15 EXECUTION  C   7422 RS22  RS22
?   TS6462D J0243934 TS6462    1 PRINT       A
```

Then, select the REPORT DD name for step T02COMP:

```
Display Filter View Print Options Search Help
-----
SFPCU41 DATA SET DISPLAY - JOB TS6462D (J0243934) LINE 1-14 (14)
OMMAND INPUT ===>                            SCROLL ===> PAGE
P  DDNAME StepName ProcStep DSID Owner   C Dest      Rec-Cnt Page
   JESMSG LG JES2      2 TS6462 H LOCAL    21
   JESJCL   JES2      3 TS6462 H LOCAL   153
   JESYSMSG JES2      4 TS6462 H LOCAL  414
   REPORT  S01DDL    106 TS6462 H LOCAL   33
   SYSPRINT S01DDL    107 TS6462 H LOCAL    3
   REPORT  T01DDL    109 TS6462 H LOCAL   33
   SYSPRINT T01DDL    110 TS6462 H LOCAL    3
   S REPORT T02COMP    112 TS6462 H LOCAL  197
   ADBMSG  T02COMP    113 TS6462 H LOCAL    2
   SYSTSPRT T02COMP    114 TS6462 H LOCAL    8
   SYSPRINT T02COMP    115 TS6462 H LOCAL   33
   ADBDIAG T02COMP    117 TS6462 H LOCAL   13
   CONVRPT T02COMP    118 TS6462 H LOCAL   27
   SYSOUT  T02COMP    119 TS6462 H LOCAL  170
```

#### Saved compare results

Use the RPT command on the **Manage Saved Compare Results (ADBPMCR)** panel. For specific instructions, see Chapter 7, “Managing saved compare results,” on page 147.

2. Read through the findings in the report.

For a description of the report and examples, see [“Compare report format” on page 124](#).

3. Optional: Read through the summary conversion report in the CONVRPT data set to determine if any conversions will occur when changing the target to match the source.

For more information about summary conversion reports, see [“Sample summary conversion report” on page 138](#).

### What to do next

If you want to change the target objects to match the source objects, take one of the following actions:

- If the report contains only those changes that you want, [Apply the changes](#).

- If the report contains unnecessary changes, modify masks or ignores as needed and run the comparison again. You can also ignore changes from the saved compare results and then run the comparison again.

**Related tasks**

[“3. Specifying compare masks” on page 71](#)

[“4. Specifying ignores” on page 80](#)

[“Ignoring changes” on page 151](#)

**Related information**

[z/OS SDSF User's Guide \(z/OS 3.1.0\)](#)

## Compare report format

The information that is contained in the compare report can vary based on the options selected on the **Specify Compare Reporting Options (GOC5RO)** panel.

**Note:** The comparison process only reports the changes that are necessary to upgrade target objects to match source objects. No actual changes are made. For information about implementing the changes, see [“Running a work statement list to apply changes” on page 157](#).

The report can contain any of the following information:

- [“Mask information” on page 124](#)
- [“Ignore field information” on page 125](#)
- [“Source and target information” on page 125](#)
- [“Compare results” on page 125](#)
- [“Summary” on page 126](#)
- [“Count report” on page 127](#)

The report might also include the following messages:

- (E) Error message
- (W) Warning message
- (I) Informational message

These messages can contain return codes, which provide additional context based on your situation.

### Mask information

If you specified that you want translation masks included in the report (by setting the **Translation masks** option to YES on the **Specify Compare Reporting Options (GOC5RO)** panel), the report includes a listing of translation masks, as shown in the following example:

```
TRANSLATION MASKS
=====

OWNER      : AAA*                , BBB*
OWNER      : TESTSYS            , PRODOWN
GRANTEE    : TESTX              , PRODOWN
AUTHID     : VND0JK2            , VNDR230
TBNAME     : TAB1*              , XXTAB*
NAME       : VND0JK2            , VNDR230
DBNAME     : DB01                , PRODDB
DBNAME     : RRR8D81A            , TTT8D81A
SGNAME     : TESTG              , PRODG
BPNAME     : BP1                 , BP4
TSBPNAME   : BP0                 , BP1
IXBPNAME   : BP0                 , BP2

Processed top down. First mask that fits a name of a given type will be used

BPNAME will cover TSBPNAME and IXBPNAME
SGNAME will cover TSSGNAME and IXSGNAME
```

NAME will cover all NAME types except COLNAME  
AUTHID will cover SQLID, OWNER, SCHEMA and GRANTOR/GRANTEE

## Ignore field information

If you specified that you want certain ignore fields included in the report (by using the **Ignore fields:** options on **Specify Compare Reporting Options (GOC5RO)** panel), the report includes a listing of the selected ignores, as shown in the following example:

```
FIELDS IGNORED WHEN COMPARING SOURCE AND TARGET OBJECTS  
=====
```

Only user requested ignore fields are reported

```
SYSINDEXES      : BPOOL(U)  
SYSINDEXPART   : PQTY(U), SQTY(U), STORTYPE(U), STORNAME(U), VCATNAME(U),  
                FREEPAGE(U), PCTFREE(U), SECQTYI(U)  
SYSTABLEPART   : PQTY(U), SQTY(U), STORTYPE(U), STORNAME(U), VCATNAME(U),  
                FREEPAGE(U), PCTFREE(U), SECQTYI(U)  
SYSTABLESPACE  : BPOOL(U), MAXROWS(U)
```

## Source and target information

The report includes information about the extraction of source and target objects, as shown in the following example:

```
OBJECT COMPARISON REPORT  
=====
```

```
Source: VIEW ADDED, FULL REPORT  
        Extracted from location *FROM DDL FILE* at 2006-06-10 09:16 by UNKNOWN  
  
Target: TABLESPACE TTT8S81D FROM CATLG  
        Extracted from DSN8 at 2006-06-10 09:16 by VNDR230
```

These lines report the following information:

- Input for the extraction of the source and target objects.  
If the objects were extracted from a Db2 catalog, this value is the Db2 subsystem ID. If the objects were extracted from a file with saved DDL statements, this value is \*FROM DDL FILE\*.
- When the extract was performed
- Who performed the extract  
This value is listed as UNKNOWN if the user ID is not known.
- Any free-form text that was added when the source or target version file was created through the ISPF full-screen interface.

If long names are used, authorization IDs or names can span lines. Object Comparison Tool tries not to split an authorization ID or a name across lines, if possible.

## Compare results

After the source and target extraction information, the report includes the results of the comparison. If **Only changed objects** = YES on **Specify Compare Reporting Options (GOC5RO)** panel, the report contains only those objects that are changed, deleted, dropped, or dropped and re-created. Otherwise, the report contains all objects that were compared.

The order of items in the report depends on the value of the **Object processing order** field on the **Options for Change Functions (ADB2PCO)** panel:

- If **Object processing order** = T (Object type), the report displays results for objects sorted by type. For example, the report might display all databases, followed by all table spaces, followed by all tables.

- If **Object processing order** = H (Database hierarchy), which is the default value, the report displays results for all object types grouped by database. In this hierarchical format, each database is followed by objects in that database. For example, table spaces in a database are listed after the database, tables in a table space are listed after the table space, and indexes over each table are listed after the table. This order has some exceptions. For example, temporal tables and history tables follow all databases, because they need to be processed after all table spaces are processed.

The comparison results include one or more of the following items:

### Added objects

The objects that are not found in the target are reported as added objects. For example:

```
View VNDR230.VDEPTS not found on target
New View VNDR230.VDEPTS will be added
Authorisations for View VNDR230.VDEPTS will be copied from source
```

### Dropped objects

The objects that are not found in the source are reported as dropped objects. For example:

```
View VNDR230.VDEPMG1 not found on source
View VNDR230.VDEPMG1 will be dropped
```

### Compared objects

For compared objects, the following sequence of information is reported:

- Object identification

The object type and object names of source and target objects are listed. The object names are listed with no masks applied.

- Differences

If differences are found, they are reported individually. The report indicates how the target object will be changed to match the source object by one of the following strings:

**(A)**

An ALTER operation can be used.

**(D)**

The object needs to be dropped and re-created.

For example:

```
Compare tablespace source(DB33971.TS33971A) and
target(DB33971.TS33971A)
(A)Field Numparts changed from 3 to
4
Tablespace will be altered
```

**Tip:** If the report includes unexpected changes to bind options for trigger packages, you might need to rebind some packages. For detailed information, see [“Troubleshooting: The Compare report shows changes to bind options for trigger packages”](#) on page 220.

## Summary

If you specified that you want a summary included in the report (by setting the **Summary report** option to YES on the **Specify Compare Reporting Options (GOC5RO)** panel), this section is included, as shown in the following example:

```
COMPARISON SUMMARY REPORT
=====

Obtyp Source Object          Target Object          Result
Object type
-----
S      RRR8D81A.DSN8S81D      TTT8D81A.DSN8S81D      No change
Tablespace
```

T	VNDR230.DEPT	VNDR230.DEPT	No change	Table
X	VNDR230.XDEPT1	VNDR230.XDEPT1	No change	Index
X	VNDR230.XDEPT2	VNDR230.XDEPT2	No change	Index
X	VNDR230.XDEPT3	VNDR230.XDEPT3	No change	Index
R	RDD	RDD	No change	
Relation				
R	RDE	RDE	No change	
Relation				
V	VNDR230.VASTRDE1	VNDR230.VASTRDE1	No change	View
V	VNDR230.VASTRDE2	VNDR230.VASTRDE2	No change	View
V	VNDR230.VDEPMG1	VNDR230.VDEPMG1	No change	View
V	VNDR230.VDEPT	VNDR230.VDEPT	No change	View
V	VNDR230.VDEPTS		Added	View
V	VNDR230.VEMPDPT1	VNDR230.VEMPDPT1	No change	View
V	VNDR230.VHDEPT	VNDR230.VHDEPT	No change	View
V	VNDR230.VPHONE	VNDR230.VPHONE	No change	View

This section of the report summarizes the actions that will be taken to modify the object or an indication that no change to the object was detected. It contains one line for each object that was compared and the result of the comparison. If long names are used, they continue on the next line in the report, in the same column and with the same indentation.

## Count report

If you specified that you want a count report included (by setting the **Object count report** option to YES on the **Specify Compare Reporting Options (GOC5RO)** panel), this section lists the number of objects that were processed per object type, as shown in the following example:

COMPARISON COUNTS REPORT							
=====							
Object type		On source	On target	Compared	Added	Dropped	Altered
Not Added	Recreated						
-----							
Tablespaces		1	1	1	0	0	
0	0	0					
Tables		1	1	1	0	0	
0	0	0					
Indexes		3	3	3	0	0	
0	0	0					
Views		8	7	7	1	0	
0	0	0					
Relations		2	2	2	0	0	
0	0	0					

It groups all objects by type and reports the number of objects on the source and on the target. The count report also lists the number of objects compared, added or not added (on source only), and dropped (if on target only). You can also see how many objects from the compared objects were altered and how many were dropped and recreated.

### Related concepts

[“Batch compare program” on page 179](#)

### Related reference

[“Compare job options” on page 87](#)

When you generate a compare batch job, you can specify a number of options to control the behavior of the comparison operation and job. These options are listed on the **Generate Compare Jobs (GOC5)** panel.

[“Compare reporting options” on page 110](#)

If **Change reporting options** = YES on the **Generate Compare Jobs (GOC5)** panel, you can change the compare report options on a subsequent panel (**Specify Compare Reporting Options (GOC5RO)** panel) before running the comparison.

## Sample compare report 1

The following sample compare report includes only those objects that were changed, added, or dropped (in the OBJECT COMPARISON REPORT section). It contains all sections of the compare report, and all ignore fields are listed (in the FIELDS IGNORED WHEN COMPARING SOURCE AND TARGET OBJECTS section).

```
-----
GOC2CMP - Compare DB2 Objects                                     2006-06-09 19:01
-----

Database 2 Object Comparison Tool
5697-L40 (C) Copyright IBM Corporation 2001, 2006.
All rights reserved. Licensed materials - property of IBM.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP schedule contract with IBM Corp.

Parameters for this run:

Suppress DROP of objects : No
Suppress DROP of columns : No
Suppress adding columns  : No

TRANSLATION MASKS
=====

DBNAME      : RRR8D81A                                     , TTT8D81A

FIELDS IGNORED WHEN COMPARING SOURCE AND TARGET OBJECTS
=====

SYSAXRELS      : AUXRELOBID(S), IBMREQD(S)
SYSCHECKS     : DBID(S), OBID(S), TIMESTAMP(S), RBA(S), IBMREQD(S)
SYSCOLAUTH     : TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), IBMREQD(S),
                CONTOKEN(S), GRANTEDTS(S)
SYSCOLUMNS   : COLCARD(S), HIGH2KEY(S), LOW2KEY(S), IBMREQD(S), STATSTIME(S),
                COLCARDF(S), CREATEDTS(S), ALTEREDTS(S)
SYSDATABASE    : DBID(S), IBMREQD(S), CREATEDBY(S), TIMESTAMP(S), CREATEDTS(S),
                ALTEREDTS(S), BPOOL(U), INDEXBP(U)
SYSDATATYPES  : CREATEDBY(S), DATATYPEID(S), CREATEDTS(S), IBMREQD(S)
SYSDBAUTH     : TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), GRANTEETYPE(S),
                IBMREQD(S), GRANTEDTS(S)
SYSFIELDS     : IBMREQD(S)
SYSFKEYS      : IBMREQD(S)
SYSINDEXES    : CLUSTERED(S), DBID(S), OBID(S), ISOBID(S), INDEXSPACE(S),
                FIRSTKEYCARD(S), FULLKEYCARD(S), NLEAF(S), NLEVELS(S),
                SPACE(S), IBMREQD(S), CLUSTERRATIO(S), CREATEDBY(S),
                IOFACTOR(S), PREFETCHFACTOR(S), STATSTIME(S),
                FIRSTKEYCARDF(S), FULLKEYCARDF(S), CREATEDTS(S), ALTEREDTS(S),
                COPYLRN(S), CLUSTERRATIOF(S), SPACEF(S), BPOOL(U)
SYSINDEXPART  : CARD(S), FAROFFPOS(S), LEAFDIST(S), NEAROFFPOS(S), IBMREQD(S),
                SPACE(S), STATSTIME(S), FAROFFPOSF(S), NEAROFFPOSF(S),
                CARDF(S), ALTEREDTS(S), SPACEF(S), DSNUM(S), EXTENTS(S),
                PSEUDO_DEL_ENTRIES(S), LEAFNEAR(S), LEAFFAR(S), CREATEDTS(S),
                PQTY(U), SQTY(U), STORTYPE(U), STORNAME(U), VCATNAME(U),
                FREEPAGE(U), PCTFREE(U), SECQTYI(U)
SYSKEYCOLUSE  : IBMREQD(S)
SYSKEYS       : IBMREQD(S)
SYSPACKDEP   : IBMREQD(S)
SYSPLANDEP   : IBMREQD(S)
SYSPARMS     : ROUTINEID(S), DATATYPEID(S), CAST_FUNCTION_ID(S), IBMREQD(S)
SYSRELS      : IBMREQD(S), RELOBID1(S), RELOBID2(S), TIMESTAMP(S)
SYSRESAUTH   : TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), IBMREQD(S),
                GRANTEDTS(S)
SYSROUTINEAUTH : GRANTEDTS(S), IBMREQD(S)
SYSROUTINES  : CREATEDBY(S), ROUTINEID(S), CREATEDTS(S), ALTEREDTS(S),
                IBMREQD(S), PARM1(S), PARM2(S), PARM3(S), PARM4(S), PARM5(S),
                PARM6(S), PARM7(S), PARM8(S), PARM9(S), PARM10(S), PARM11(S),
                PARM12(S), PARM13(S), PARM14(S), PARM15(S), PARM16(S),
```

```

                PARM17(S), PARM18(S), PARM19(S), PARM20(S), PARM21(S),
                PARM22(S), PARM23(S), PARM24(S), PARM25(S), PARM26(S),
                PARM27(S), PARM28(S), PARM29(S), PARM30(S)
SYSSCHEMAAUTH  : GRANTEDTS(S), IBMREQD(S)
SYSSEQUENCES   : NAME(S), SEQUENCEID(S), CREATEDBY(S), CREATEDTS(S),
                ALTEREDTS(S), IBMREQD(S)
SYSSEQUENCEAUTH : CONTOKEN(S), GRANTEDTS(S), IBMREQD(S)
SYSSEQUENCESDEP : BSEQUENCEID(S), IBMREQD(S)
SYSSTOGRROUP   : VPASSWORD(S), SPACE(S), SPCDATE(S), IBMREQD(S), CREATEDBY(S),
                STATSTIME(S), CREATEDTS(S), ALTEREDTS(S), SPACEF(S)
SYSSYNONYMS    : IBMREQD(S), CREATEDBY(S), CREATEDTS(S)
SYSTABAUTH     : TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), IBMREQD(S),
                GRANTEDTS(S)
SYSTABCONST    : CREATEDTS(S), IBMREQD(S)
SYSTABLEPART   : CARD(S), FARINDREF(S), NEARINDREF(S), PERCACTIVE(S),
                PERCDROP(S), IBMREQD(S), CHECKRID(S), SPACE(S), PAGESAVE(S),
                STATSTIME(S), CHECKRID5B(S), EPOCH(S), CARDF(S), ALTEREDTS(S),
                SPACEF(S), DSNUM(S), EXTENTS(S), LIMITKEY_INTERNAL(S),
                CREATEDTS(S)
SYSTABLES      : DBID(S), OBID(S), CLUSTERRID(S), CARD(S), NPAGES(S),
                PCTPAGES(S), IBMREQD(S), PARENTS(S), CHILDREN(S), KEYOBID(S),
                CHECKRID(S), CREATEDBY(U), CREATEDTS(S), ALTEREDTS(S),
                RBA1(S), RBA2(S), PCTROWCOMP(S), STATSTIME(S), CARDF(S),
                CHECKRID5B(S), NPAGESF(S), SPACEF(S), AVGWLEN(S),
                RELCREATED(S)
SYSTABLESPACE  : DBID(S), OBID(S), PSID(S), NTABLES(S), NACTIVE(S), SPACE(S),
                IBMREQD(S), ROOTNAME(S), ROOTCREATOR(U), CREATEDBY(S),
                STATSTIME(S), CREATEDTS(S), ALTEREDTS(S), NACTIVEF(S),
                SPACEF(S), BPOOL(U)
SYSTRIGGERS    : DBID(S), OBID(S), CREATEDBY(S), CREATEDTS(S), IBMREQD(S)
SYSVIEWDEP    : IBMREQD(S)
SYSVIEWS      : IBMREQD(S), RELCREATED(S), REFRESH_TIME(S), SIGNATURE(S)
SYSVOLUMES    : IBMREQD(S)

```

(S) System ignore. Set automatically by compare  
 Also set for fields only used by newer versions of DB2  
 (U) User ignore. Requested by user input  
 (U) is reported for fields that are both System and User ignores

-----  
 GOC2CMP - Compare DB2 Objects 2006-06-09 19:01  
 -----

OBJECT COMPARISON REPORT  
 =====

Only changed, added and deleted objects will be reported

```

Source: VIEW ADDED
       Extracted from location *FROM DDL FILE* at 2006-06-09 18:57 by UNKNOWN

Target: TABLESPACE TTT8S81D FROM CAT
       Extracted from DSN8 at 2006-06-09 18:57 by VNDR230

Target system is DB2 Release 810

```

View VNDR230.VDEPTS not found on target  
 New View VNDR230.VDEPTS will be added  
 Authorisations for View VNDR230.VDEPTS will be copied from source

COMPARISON SUMMARY REPORT  
 =====

Obtyp	Source Object	Target Object	Result	Object type
S	RRR8D81A.DSN8S81D	TTT8D81A.DSN8S81D	No change	Tablespace
T	VNDR230.DEPT	VNDR230.DEPT	No change	Table
X	VNDR230.XDEPT1	VNDR230.XDEPT1	No change	Index
X	VNDR230.XDEPT2	VNDR230.XDEPT2	No change	Index
X	VNDR230.XDEPT3	VNDR230.XDEPT3	No change	Index
R	RDD	RDD	No change	Relation
R	RDE	RDE	No change	Relation
V	VNDR230.VASTRDE1	VNDR230.VASTRDE1	No change	View
V	VNDR230.VASTRDE2	VNDR230.VASTRDE2	No change	View
V	VNDR230.VDEPMG1	VNDR230.VDEPMG1	No change	View
V	VNDR230.VDEPT	VNDR230.VDEPT	No change	View
V	VNDR230.VDEPTS		Added	View
V	VNDR230.VEMPDPT1	VNDR230.VEMPDPT1	No change	View



TRANSLATION MASKS

DBNAME : RRR8D81A , TTT8D81A

FIELDS IGNORED WHEN COMPARING SOURCE AND TARGET OBJECTS

Only user requested ignore fields are reported

SYSINDEXES : BPOOL(U)  
SYSINDEXPART : PQTY(U), SQTY(U), STORTYPE(U), STORNAME(U), VCATNAME(U),  
FREEPAGE(U), PCTFREE(U), SECQTYI(U)  
SYSTABLEPART : PQTY(U), SQTY(U), STORTYPE(U), STORNAME(U), VCATNAME(U),  
FREEPAGE(U), PCTFREE(U), SECQTYI(U)  
SYSTABLESPACE : BPOOL(U), MAXROWS(U)

-----  
GOC2CMP - Compare DB2 Objects

2006-06-10 09:20  
-----

OBJECT COMPARISON REPORT

Source: VIEW ADDED, FULL REPORT  
Extracted from location \*FROM DDL FILE\* at 2006-06-10 09:16 by UNKNOWN

Target: TABLESPACE TTT8S81D FROM CATLG  
Extracted from DSN8 at 2006-06-10 09:16 by VNDR230

Target system is DB2 Release 810

Compare tablespace source(RRR8D81A.DSN8S81D) and target(TTT8D81A.DSN8S81D)

No changes to Tablespace  
Grant(target): Grantor=VNDR230 Grantee:PUBLIC (Kept)

Compare table source(VNDR230.DEPT) and target(VNDR230.DEPT)

No changes to Table  
Grant(target): Grantor=VNDR230 Grantee:PUBLIC\* (Kept)

Compare index source(VNDR230.XDEPT1) and target(VNDR230.XDEPT1)

No changes to Index

Compare index source(VNDR230.XDEPT2) and target(VNDR230.XDEPT2)

No changes to Index

Compare index source(VNDR230.XDEPT3) and target(VNDR230.XDEPT3)

No changes to Index

View VNDR230.VASTRDE1 not found on source

View VNDR230.VASTRDE1 will be dropped

View VNDR230.VASTRDE2 not found on source

View VNDR230.VASTRDE2 will be dropped

View VNDR230.VDEPMG1 not found on source

View VNDR230.VDEPMG1 will be dropped

Compare View source(VNDR230.VDEPT) and target(VNDR230.VDEPT)

No changes to View  
Grant(target): Grantor=VNDR230 Grantee:PUBLIC\* (Kept)

View VNDR230.VDEPTS not found on target

New View VNDR230.VDEPTS will be added  
Authorisations for View VNDR230.VDEPTS will be copied from source

View VNDR230.VEMPDPT1 not found on source

View VNDR230.VEMPDPT1 will be dropped

Compare View source(VNDR230.VHDEPT) and target(VNDR230.VHDEPT)

No changes to View

Grant(target): Grantor=VNDR230 Grantee:PUBLIC\* (Kept)

View VNDR230.VPHONE not found on source  
View VNDR230.VPHONE will be dropped

Compare Referential Constraint source(RDD) and target(RDD)  
No changes to Referential constraint

Compare Referential Constraint source(RDE) and target(RDE)  
No changes to Referential constraint

COMPARISON SUMMARY REPORT  
=====

Obtyp	Source Object	Target Object	Result	Object type
S	RRR8D81A.DSN8S81D	TTT8D81A.DSN8S81D	No change	Tablespace
T	VNDR230.DEPT	VNDR230.DEPT	No change	Table
X	VNDR230.XDEPT1	VNDR230.XDEPT1	No change	Index
X	VNDR230.XDEPT2	VNDR230.XDEPT2	No change	Index
X	VNDR230.XDEPT3	VNDR230.XDEPT3	No change	Index
R	RDD	RDD	No change	Relation
R	RDE	RDE	No change	Relation
V		VNDR230.VASTRDE1	Dropped	View
V		VNDR230.VASTRDE2	Dropped	View
V		VNDR230.VDEPMG1	Dropped	View
V	VNDR230.VDEPT	VNDR230.VDEPT	No change	View
V	VNDR230.VDEPTS		Added	View
V		VNDR230.VEMPDPT1	Dropped	View
V	VNDR230.VHDEPT	VNDR230.VHDEPT	No change	View
V		VNDR230.VPHONE	Dropped	View

COMPARISON COUNTS REPORT  
=====

Object type	On source	On target	Compared	Added	Dropped	Altered	Not Added
Recreated							
Schemas	0	0	0	0	0	0	0
User Def Types	0	0	0	0	0	0	0
Sequences	0	0	0	0	0	0	0
Databases	0	0	0	0	0	0	0
Tablespaces	1	1	1	0	0	0	0
Tables	1	1	1	0	0	0	0
Indexes	3	3	3	0	0	0	0
Aliases	0	0	0	0	0	0	0
Storage groups	0	0	0	0	0	0	0
Synonyms	0	0	0	0	0	0	0
Functions	0	0	0	0	0	0	0
Stored procedures	0	0	0	0	0	0	0
Triggers	0	0	0	0	0	0	0
Views	3	7	2	1	5	0	0
Relations	2	2	2	0	0	0	0

### Sample compare report 3

The following sample compare report includes only the objects that were changed, added, or dropped (in the OBJECT COMPARISON REPORT section). In this case, additional masks were specified, and only

system-generated ignore fields are listed (in the FIELDS IGNORED WHEN COMPARING SOURCE AND TARGET OBJECTS section).

-----  
GOC2CMP - Compare DB2 Objects

2006-06-10 10:00  
-----

Database 2 Object Comparison Tool  
5697-L40 (C) Copyright IBM Corporation 2001, 2006.  
All rights reserved. Licensed materials - property of IBM.  
US Government Users Restricted Rights - Use, duplication or disclosure  
restricted by GSA ADP schedule contract with IBM Corp.

TRANSLATION MASKS  
=====

OWNER	:	AAA*	,	BBB*
OWNER	:	TESTSYS	,	PRODOWN
GRANTEE	:	TESTX	,	PRODOWN
AUTHID	:	VND0JK2	,	VNDR230
TBNAME	:	TAB1*	,	XXTAB*
NAME	:	VND0JK2	,	VNDR230
DBNAME	:	DB01	,	PROddb
DBNAME	:	RRR8D81A	,	TTT8D81A
SGNAME	:	TESTG	,	PRODG
BPNAME	:	BP1	,	BP4
TSBPNAME	:	BP0	,	BP1
IXBPNAME	:	BP0	,	BP2

Processed top down. First mask that fits a name of a given type will be used

BPNAME will cover TSBPNAME and IXBPNAME  
SGNAME will cover TSSGNAME and IXSGNAME  
NAME will cover all NAME types except COLNAME  
AUTHID will cover SQLID, OWNER, SCHEMA and GRANTOR/GRANTEE

FIELDS IGNORED WHEN COMPARING SOURCE AND TARGET OBJECTS  
=====

Only system generated ignore fields are reported  
System ignore fields also found in the user ignore input will not be reported

SYSAXRELS	:	AUXRELOBID(S), IBMREQD(S)
SYSCHECKS	:	DBID(S), OBID(S), TIMESTAMP(S), RBA(S), IBMREQD(S)
SYSCOLAUTH	:	TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), IBMREQD(S), CONTOKEN(S), GRANTEDTS(S)
SYSCOLUMNS	:	COLCARD(S), HIGH2KEY(S), LOW2KEY(S), IBMREQD(S), STATSTIME(S), COLCARD(S), CREATEDTS(S), ALTEREDTS(S)
SYSDATABASE	:	DBID(S), IBMREQD(S), CREATEDBY(S), TIMESTAMP(S), CREATEDTS(S), ALTEREDTS(S)
SYSDATATYPES	:	CREATEDBY(S), DATATYPEID(S), CREATEDTS(S), IBMREQD(S)
SYSDBAUTH	:	TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), GRANTEETYPE(S), IBMREQD(S), GRANTEDTS(S)
SYSFIELDS	:	IBMREQD(S)
SYSFOREIGNKEYS	:	IBMREQD(S)
SYSINDEXES	:	CLUSTERED(S), DBID(S), OBID(S), ISOBID(S), INDEXSPACE(S), FIRSTKEYCARD(S), FULLKEYCARD(S), NLEAF(S), NLEVELS(S), SPACE(S), IBMREQD(S), CLUSTERRATIO(S), CREATEDBY(S), IOFACTOR(S), PREFETCHFACTOR(S), STATSTIME(S), FIRSTKEYCARD(S), FULLKEYCARD(S), CREATEDTS(S), ALTEREDTS(S), COPYLRN(S), CLUSTERRATIO(S), SPACE(S)
SYSINDEXPART	:	CARD(S), FAROFFPOS(S), LEAFDIST(S), NEAROFFPOS(S), IBMREQD(S), SPACE(S), STATSTIME(S), FAROFFPOS(S), NEAROFFPOS(S), CARD(S), ALTEREDTS(S), SPACE(S), DSNUM(S), EXTENTS(S), PSEUDO_DEL_ENTRIES(S), LEAFNEAR(S), LEAFFAR(S), CREATEDTS(S)
SYSKEYCOLUSE	:	IBMREQD(S)
SYSKEYS	:	IBMREQD(S)
SYSPACKDEP	:	IBMREQD(S)
SYSPLANDEP	:	IBMREQD(S)
SYSPARMS	:	ROUTINEID(S), DATATYPEID(S), CAST_FUNCTION_ID(S), IBMREQD(S)
SYSRELS	:	IBMREQD(S), RELOBID1(S), RELOBID2(S), TIMESTAMP(S)
SYSRESAUTH	:	TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), IBMREQD(S), GRANTEDTS(S)
SYSROUTINEAUTH	:	GRANTEDTS(S), IBMREQD(S)
SYSROUTINES	:	CREATEDBY(S), ROUTINEID(S), CREATEDTS(S), ALTEREDTS(S),

```

                IBMREQD(S), PARM1(S), PARM2(S), PARM3(S), PARM4(S), PARM5(S),
                PARM6(S), PARM7(S), PARM8(S), PARM9(S), PARM10(S), PARM11(S),
                PARM12(S), PARM13(S), PARM14(S), PARM15(S), PARM16(S),
                PARM17(S), PARM18(S), PARM19(S), PARM20(S), PARM21(S),
                PARM22(S), PARM23(S), PARM24(S), PARM25(S), PARM26(S),
                PARM27(S), PARM28(S), PARM29(S), PARM30(S)
SYSSCHEMAAUTH : GRANTEDTS(S), IBMREQD(S)
SYSSEQUENCES : NAME(S), SEQUENCEID(S), CREATEDBY(S), CREATEDTS(S),
                ALTEREDTS(S), IBMREQD(S)
SYSSEQUENCEAUTH : CONTOKEN(S), GRANTEDTS(S), IBMREQD(S)
SYSSEQUENCESDEP : BSEQUENCEID(S), IBMREQD(S)
SYSSTOGRUP : VPASSWORD(S), SPACE(S), SPCDATE(S), IBMREQD(S), CREATEDBY(S),
                STATSTIME(S), CREATEDTS(S), ALTEREDTS(S), SPACEF(S)
SYSSYNONYMS : IBMREQD(S), CREATEDBY(S), CREATEDTS(S)
SYSTABAUTH : TIMESTAMP(S), DATEGRANTED(S), TIMEGRANTED(S), IBMREQD(S),
                GRANTEDTS(S)
SYSTABCONST : CREATEDTS(S), IBMREQD(S)
SYSTABLEPART : CARD(S), FARINDREF(S), NEARINDREF(S), PERCACTIVE(S),
                PERCDROP(S), IBMREQD(S), CHECKRID(S), SPACE(S), PAGESAVE(S),
                STATSTIME(S), CHECKRID5B(S), EPOCH(S), CARDF(S), ALTEREDTS(S),
                SPACEF(S), DSNUM(S), EXTENTS(S), LIMITKEY_INTERNAL(S),
                CREATEDTS(S)
SYSTABLES : DBID(S), OBID(S), CLUSTERRID(S), CARD(S), NPAGES(S),
                PCTPAGES(S), IBMREQD(S), PARENTS(S), CHILDREN(S), KEYOBID(S),
                CHECKRID(S), CREATEDBY(S), CREATEDTS(S), ALTEREDTS(S),
                RBA1(S), RBA2(S), PCTROWCOMP(S), STATSTIME(S), CARDF(S),
                CHECKRID5B(S), NPAGESF(S), SPACEF(S), AVGWLEN(S),
                RELCREATED(S)
SYSTABLESPACE : DBID(S), OBID(S), PSID(S), NTABLES(S), NACTIVE(S), SPACE(S),
                IBMREQD(S), ROOTNAME(S), ROOTCREATOR(S), CREATEDBY(S),
                STATSTIME(S), CREATEDTS(S), ALTEREDTS(S), NACTIVEF(S),
                SPACEF(S)
SYSTRIGGERS : DBID(S), OBID(S), CREATEDBY(S), CREATEDTS(S), IBMREQD(S)
SYSVIEWDEP : IBMREQD(S)
SYSVIEWS : IBMREQD(S), RELCREATED(S), REFRESH_TIME(S), SIGNATURE(S)
SYSVOLUMES : IBMREQD(S)

```

-----  
GOC2CMP - Compare DB2 Objects

2006-06-10 10:00  
-----

OBJECT COMPARISON REPORT

=====

Only changed, added and deleted objects will be reported

Source: VIEW ADDED, CHANGED ONLY REPT  
Extracted from location \*FROM DDL FILE\* at 2006-06-10 09:56 by UNKNOWN

Target: TABLESPACE TTT8S81D FROM CATLG  
Extracted from DSN8 at 2006-06-10 09:56 by VNDR230

Target system is DB2 Release 810

View VNDR230.VDEPTS not found on target  
New View VNDR230.VDEPTS will be added  
Authorisations for View VNDR230.VDEPTS will be copied from source

COMPARISON SUMMARY REPORT

=====

Obtyp	Source Object	Target Object	Result	Object type
S	RRR8D81A.DSN8S81D	TTT8D81A.DSN8S81D	No change	Tablespace
T	VNDR230.DEPT	VNDR230.DEPT	No change	Table
X	VNDR230.XDEPT1	VNDR230.XDEPT1	No change	Index
X	VNDR230.XDEPT2	VNDR230.XDEPT2	No change	Index
X	VNDR230.XDEPT3	VNDR230.XDEPT3	No change	Index
R	RDD	RDD	No change	Relation
R	RDE	RDE	No change	Relation
V	VNDR230.VASTRDE1	VNDR230.VASTRDE1	No change	View
V	VNDR230.VASTRDE2	VNDR230.VASTRDE2	No change	View
V	VNDR230.VDEPMG1	VNDR230.VDEPMG1	No change	View
V	VNDR230.VDEPT	VNDR230.VDEPT	No change	View

V	VNDR230.VDEPTS		Added	View
V	VNDR230.VEMPDPT1	VNDR230.VEMPDPT1	No change	View
V	VNDR230.VHDEPT	VNDR230.VHDEPT	No change	View
V	VNDR230.VPHONE	VNDR230.VPHONE	No change	View

-----  
**COMPARISON COUNTS REPORT**  
 =====

Object type	On source	On target	Compared	Added	Dropped	Altered	Not
Added Recreated							
----- Tablespace 0 0	1	1	1	0	0	0	0
Tables 0 0	1	1	1	0	0	0	0
Indexes 0 0	3	3	3	0	0	0	0
Views 0 0	8	7	7	1	0	0	0
Relations 0 0	2	2	2	0	0	0	0
----- -----							

## Sample compare report 4

The following sample compare report shows a subset of the report when LOB objects are converted from explicit to implicit. The OBJECT COMPARISON REPORT section includes messages about the converted objects. The COMPARISON SUMMARY REPORT section lists all objects and the end result of the action performed.

**OBJECT COMPARISON REPORT**  
 =====

Tablespace DB33971.TL971APN not found on target  
 New LOB Tablespace DB33971.TL971APN will be added

Compare tablespace source(DB33971.TL971AP1) and target(DB33971.TL971AP1)  
 No changes to Tablespace

Compare tablespace source(DB33971.TL971AP2) and target(DB33971.TL971AP2)  
 No changes to Tablespace

Compare tablespace source(DB33971.TL971AP3) and target(DB33971.TL971AP3)  
 No changes to Tablespace

Tablespace DB33971.TL971BP1 not found on target  
 New LOB Tablespace DB33971.TL971BP1 will be added

Compare tablespace source(DB33971.TS33971A) and target(DB33971.TS33971A)  
 (A)Field Numparts changed from 3 to 4  
 Tablespace will be altered

Tablespace DB33971.TS33971B not found on target  
 New Tablespace DB33971.TS33971B will be added

Compare table source(VNDR1.TB33971A) and target(VNDR1.TB33971A)  
 (D)Column CLOB2 added

(A)Partition(s) added to the target table

Auxiliary table VNDR1.TB971AP1 processed

Auxiliary table VNDR1.TB971AP2 processed

Auxiliary table VNDR1.TB971AP3 processed

Auxiliary table VNDR1.TB971APN added

ADB7163W The number of auxiliary tables associated with the source table might not be consistent with the number of LOB columns in the source table and with the number of partitions in the table space. Implicit LOB objects are used when the base table is re-created. After changes are applied, ensure that one auxiliary table exists for each LOB column in each partition.

Table VNDR1.TB33971A is partitioned and will be dropped by dropping tablespace DB33971.TS33971A

Table will be recreated

Table data will not be converted

Not eligible for FORMAT INTERNAL processing

Table VNDR1.TB33971B not found on target

Auxiliary table VNDR1.TB971BP1 processed  
 ADB7150W Source contains incomplete set of explicit LOB objects  
 therefore all LOB objects for this base table will be created  
 implicitly.  
 New Table VNDR1.TB33971B will be added

Compare index source(VNDR1.IX33971A) and  
 target(VNDR1.IX33971A)  
 Index VNDR1.IX33971A will be dropped by dropping the  
 tablespace  
 Index will be recreated because the base table will be dropped and  
 recreated

Index VNDR1.IX33971B not found on  
 target  
 New Index VNDR1.IX33971B will be  
 added

Index VNDR1.IX971APN not found on  
 target  
 This is an index on auxiliary  
 table.  
 The index is will not be created because the auxiliary table has been converted from explicit to  
 implicit.  
 The respective index will be created implicitly by  
 DB2.

Compare auxiliary index source(VNDR1.IX971AP1) and  
 target(VNDR1.IX971AP1)  
 Index VNDR1.IX971AP1 will be  
 dropped  
 Index will not be recreated because the auxiliary table has been converted from explicit to  
 implicit

Compare auxiliary index source(VNDR1.IX971AP2) and  
 target(VNDR1.IX971AP2)  
 Index VNDR1.IX971AP2 will be  
 dropped  
 Index will not be recreated because the auxiliary table has been converted from explicit to  
 implicit

Compare auxiliary index source(VNDR1.IX971AP3) and  
 target(VNDR1.IX971AP3)  
 Index VNDR1.IX971AP3 will be  
 dropped  
 Index will not be recreated because the auxiliary table has been converted from explicit to  
 implicit

Index VNDR1.IX971BP1 not found on  
 target  
 This is an index on auxiliary  
 table.  
 The index is will not be created because the auxiliary table has been converted from explicit to  
 implicit.  
 The respective index will be created implicitly by DB2.

COMPARISON SUMMARY  
 REPORT  
 =====

Obtyp	Source Object	Target Object	Result	Object
D	DB33971	DB33971	No change	
S	DB33971.TL971APN		Not added	
T	VNDR1.TB971APN		Not added	
X	VNDR1.IX971APN		Not added	

```

S      DB33971.TL971AP1          DB33971.TL971AP1          No change
Tablespace
T      VNDR1.TB971AP1          VNDR1.TB971AP1          Dropped
Table
X      VNDR1.IX971AP1          VNDR1.IX971AP1          Dropped
Index
S      DB33971.TL971AP2          DB33971.TL971AP2          No change
Tablespace
T      VNDR1.TB971AP2          VNDR1.TB971AP2          Dropped
Table
X      VNDR1.IX971AP2          VNDR1.IX971AP2          Dropped
Index
S      DB33971.TL971AP3          DB33971.TL971AP3          No change
Tablespace
T      VNDR1.TB971AP3          VNDR1.TB971AP3          Dropped
Table
X      VNDR1.IX971AP3          VNDR1.IX971AP3          Dropped
Index
S      DB33971.TL971BP1          DB33971.TL971BP1          Not added
Tablespace
T      VNDR1.TB971BP1          VNDR1.TB971BP1          Not added
Table
X      VNDR1.IX971BP1          VNDR1.IX971BP1          Not added
Index
S      DB33971.TS33971A          DB33971.TS33971A          Altered
Tablespace
T      VNDR1.TB33971A          VNDR1.TB33971A          Dropped/created
Table
X      VNDR1.IX33971A          VNDR1.IX33971A          Dropped/created
Index
S      DB33971.TS33971B          DB33971.TS33971B          Added
Tablespace
T      VNDR1.TB33971B          VNDR1.TB33971B          Added
Table
X      VNDR1.IX33971B          VNDR1.IX33971B          Added
Index
-----
-----

```

COMPARISON COUNTS  
REPORT

=====

Object type	On source	On target	Compared	Added	Dropped	Altered	Not added
Recreated							
-----							
Schemas	0	0	0	0	0	0	0
0							
User Def Types	0	0	0	0	0	0	0
0							
Sequences	0	0	0	0	0	0	0
0							
Databases	1	1	1	0	0	0	0
0							
Tablespaces	7	4	4	1	0	1	
2							
Tables	7	4	4	1	0	0	
2							
Indexes	7	4	4	1	0	0	
2							
Aliases	0	0	0	0	0	0	0
0							
Storage groups	0	0	0	0	0	0	0
0							
Synonyms	0	0	0	0	0	0	0
0							
Functions	0	0	0	0	0	0	0
0							
Stored procedures	0	0	0	0	0	0	0
0							
Triggers	0	0	0	0	0	0	0
0							
Views	0	0	0	0	0	0	0
0							
Column masks	0	0	0	0	0	0	0
0							

Row permissions	0	0	0	0	0	0
0	0					
Relations	0	0	0	0	0	0
0	0					

## Sample compare report 5

The following sample excerpt from a compare report shows the ignore field listing. In this case, only those fields that are ignored for specific objects are listed, because the **Object Specific** option was set to YES on the **Specify Compare Reporting Options (GOC5RO)** panel.

```
>FIELDS IGNORED WHEN COMPARING SPECIFIC OBJECTS
=====
Specified Object Name          Compared Object Name
-----
DB47985.TS47985A              DB47985.TS47985A
(f)SYSTABLESPACE:PARTITIONS,BPOOL
DB47985.TS47985B              DB47985.TS47985B
(f)SYSINDEXPART:PARTITION,LIMITKEY
(f)SYSTABLEPART:PARTITION,LIMITKEY,LIMITKEY_INTERNAL,LOGICAL_PART
(f)SYSTABLES:PARTKEYCOLNUM
(f)SYSTABLESPACE:PARTITIONS
(f)SYSCOLUMNS:PARTKEY_COLSEQ,PARTKEY_ORDERING
(f)SYSAXURELS:PARTITION
DB47985.TS479*5B              DB47985.TS47985B
(f)SYSTABLESPACE:PARTITIONS
DB47985.TS47985C              DB47985.TS47985C
(f)SYSTABLESPACE:BPPOOL
DB47985.TS47985D              No Match
(f)SYSTABLESPACE:BPPOOL
DB47985                        DB47985
(f)SYSDATABASE:INDEXBP
DB47985.TS479*5B              DB47985.TS47985B
(g)PBG_NUMPARTS
DB47985.TS47985*              DB47985.TS47985A
                                DB47985.TS47985B
                                DB47985.TS47985C
(f)SYSTABLESPACE:TYPE,ENCODING_SCHEME,SBCS_CCSID,DBCS_CCSID,MAXROWS,
    LOCKPART,LOG,CURRENT_VERSION,CREATOR_TYPE,INSTANCE,CLONE
DB4798501234567890>.TS47985E9012345678> No Match
(f)SYSTABLESPACE:PARTITIONS
-
(g) Indicates the name of the GENERIC group that was used.
(f) Indicates table name:field list.
```

## Sample summary conversion report

You can use the *summary conversion report* to determine whether conversions will occur when a change is run. This information can help you discover any potential problems.

A summary conversion report is generated if the **Conversion report** option is set to YES on the **Specify Compare Reporting Options (GOC5RO)** panel). This report is printed to a separate data set that is identified by the DD statement CONVRPT.

The following sample summary conversion report in the following figure shows the truncations and conversions that will take place when the change runs.

```
>***** TOP OF DATA *****
-----
GOC2CMP - Expected Conversion Errors
-----

Database 2 Object Comparison Tool
5697-L40 Copyright IBM Corporation 2001, 2009.
All rights reserved. Licensed materials - property of IBM.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP schedule contract with IBM Corp.
-----
```

## GOC2CMP - Expected Conversion Errors

Source:  
Extracted from location \*FROM DDL FILE\* at 2009-06-23 13:52 by VNDRG

Target:  
Extracted from DSN9 at 2009-06-23 13:52 by VNDRG

Target system is DB2 Release  
915

Conversion report generated in ANALYZE mode

### CONVERSION REPORT SUMMARY =====

Compare table source(VNDRG.SRC257TB) and target(VNDRG.TAR257TB)

Column name	From type	To type
EMPNO	INTEGER	SMALLINT
NAME	CHAR(30)	CHAR(25)

Compare table source(VNDRG.SRC257TB1) and target(VNDRG.TAR257TB1)

Column name	From type	To type
SALARY	SMALLINT	DECIMAL(5,2)

\*\*\*\*\* BOTTOM OF DATA \*\*\*\*\*

## Possible conversion errors

Conversion errors might occur when Object Comparison Tool applies changes to the target due to truncation during conversion of data types.

The following conversion errors might occur:

- INTEGER to SMALLINT
- INTEGER to DECIMAL
- SMALLINT to DECIMAL
- FLOAT to SMALLINT
- FLOAT to INTEGER
- FLOAT to BIGINT
- FLOAT to DECIMAL
- DECIMAL to SMALLINT
- DECIMAL to INTEGER
- DECIMAL to DECIMAL
- DECIMAL to DATE
- DECIMAL to TIME
- DECIMAL to TIMESTAMP
- CHAR to CHAR
- CHAR to BINARY
- CHAR to VARCHAR
- CHAR to DATE
- CHAR to TIME
- CHAR to TIMESTAMP
- CHAR to SMALLINT
- CHAR to INTEGER
- CHAR to DECIMAL
- CHAR to LONGVAR

BINARY to BINARY  
VARBINARY to VARBINARY  
VARCHAR to CHAR  
VARCHAR to VARCHAR  
VARCHAR to VARBINARY  
VARCHAR to TIME  
VARCHAR to TIMESTAMP  
VARCHAR to SMALLINT  
VARCHAR to INTEGER  
VARCHAR to DECIMAL  
VARCHAR to LONGVAR  
VARCHAR to DATE  
  
LONGVAR to CHAR  
LONGVAR to VARCHAR  
LONGVAR to DATE  
LONGVAR to TIME  
LONGVAR to TIMESTAMP  
LONGVAR to LONGVAR  
  
GRAPHIC to GRAPHIC  
GRAPHIC to VARGRAPHIC  
GRAPHIC to LONGVAR  
  
VARGRAPHIC to GRAPHIC  
VARGRAPHIC to LONGVAR  
VARGRAPHIC to VARGRAPHIC  
  
LONGVARG to GRAPHIC  
LONGVAR to VARGRAPHIC  
LONGVAR to LONGVAR  
  
DECFLOAT to SMALLINT  
DECFLOAT to INTEGER  
DECFLOAT to FLOAT  
DECFLOAT to DECIMAL  
DECFLOAT to BIGINT  
  
DATE to CHAR  
DATE to VARCHAR  
  
TIME to CHAR  
TIME to VARCHAR  
  
TIMESTAMP to CHAR  
TIMESTAMP to VARCHAR

---

## Chapter 6. Excluding objects from the compare process

To be more selective about which objects are included in a comparison, you can specify that certain objects be excluded from the compare process. To do so, you must define an *exclude specification*.

### About this task

An *exclude specification* is a defined list of objects that Object Comparison Tool is to exclude from the source, target, or both. For example, you might want to use exclude specifications in the following scenarios:

- Object Comparison Tool does not add source authorizations to the target if the source authorization does not exist at the target. Instead, a warning is issued. If, however, you exclude authorizations from the source, no message is issued.
- When comparing objects, if an object exists in the target only, the resulting action is to drop the object from the target. However, if the object is excluded, it is not processed and the object is not dropped at the target. The object is retained. The same action occurs with authorizations. During the compare process, when you exclude an authorization, the authorization that already exists at the target is retained.
- When you use the Db2 Administration Tool GEN function, any authorizations in the exclude specification are excluded from DDL generation.

An exclude specification can include authorizations. You can exclude authorizations independent from the object with which they are associated.

The exclude specification on an object does not cascade to its dependent objects. You must list all objects to exclude. For example, the exclusion of a table does not mean that its dependent objects, such as indexes and views, are also excluded. Any objects to be excluded must be explicitly included in an exclude specification. The exceptions are history tables, schemas, and archive tables:

- If a temporal table is excluded, its history table is also excluded. If the history table is specified in an exclude specification, both the temporal and history table are excluded.
- If a schema is excluded, all UDFs, UDTs, procedures, triggers, and sequences that are part of the schema are excluded.
- When an archive-enabled table is excluded, its corresponding archive table is also excluded. Similarly, if an archive table is excluded, its corresponding archive-enabled table is also excluded.

You can set the duration of an exclude specification. After the date is passed, the exclude specification is eligible for deletion. You can later change an autodelete date by modifying the exclude specification.

Excluded objects can still be dropped implicitly as a result of a comparison, depending on the setting of the **Allow implicit drop of excluded objects** field on the **Generate Compare Jobs (GOC5)** panel. An *implicit drop* occurs when the action of dropping an object results in the drop of a dependent object. For example, if a comparison results in a table space being dropped at the target, the table, index, and other objects dependent on the table space should also be dropped. If **Allow implicit drop of excluded objects** is set to YES, excluded objects can be dropped as needed and are re-created according to the target definition. If this option is set to NO, if an excluded object needs to be dropped (perhaps because its parent object was dropped or because its dependent object was dropped), Object Comparison Tool stops processing the compare, and a severe message is issued.

**Restriction:** You cannot manage exclude specifications when you are analyzing or running a change in Change Management.

## Procedure

To exclude objects from a comparison:

1. Create an exclude specification by using one of the following methods:
  - Create one based on saved compare results. See [“Creating exclude specifications from saved compare results”](#) on page 142
  - Create one during the compare process. See [“Creating or editing exclude specifications during a comparison”](#) on page 144.
  - Define one in Change Management (CM). See [Creating and managing exclude specifications \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).
2. Reference the exclude specification when specifying the source or target. See [“1. Specifying source objects”](#) on page 56, [“2. Specifying target objects”](#) on page 68, or both.

## Creating exclude specifications from saved compare results

Your saved compare results might include objects that you do not want to include in future comparisons. You can select these objects from the saved compare results and add them to an exclude specification. You can exclude them from the source, target, or both.

**Restriction:** Because compare results can be saved for only the following object types, this method of creating exclude specifications (from the saved compare results) applies to only these objects:

- tables
- indexes
- global variables
- distinct data types

## Procedure

To create exclude specifications from saved compare results:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option MR, and press Enter.
2. On the **Manage Saved Compare Results (ADBPMCR)** panel, specify any filtering criteria for the saved compare results that you want displayed, and press Enter.
3. On the **Manage Saved Compare Results (ADBPCR)** panel, specify the S line command next to the result that you want to select, and press Enter.
4. On the **Compare Results (ADBPCRS)** panel, specify one of the following line commands next to the object that you want to exclude and complete the resulting panel:

Option	Description
EX	Exclude On the resulting <b>Create Exclude Specification (ADBPCEX)</b> panel, specify the requested information about for the exclude specification, and press Enter:

Option	Description
	<pre> ADBPCEX n ----- Create Exclude Specification ----- 14:53 Command ==&gt;  Source Exclude Specification:   Owner . . . . . JSMITH      &gt; (Optional, default is JSMITH, ? to lookup)   Name  . . . . . SRCIESPEC01 &gt; (Required, ? to lookup)   Comment . . . . .                               &gt;   Eligible for auto-delete . . . 30 (no of days, blank for no auto-delete)  Target Exclude Specification:   Owner . . . . . JSMITH      &gt; (Optional, default is JSMITH, ? to lookup)   Name  . . . . . TGTIESPEC01 &gt; (Required, ? to lookup)   Comment . . . . .                               &gt;   Eligible for auto-delete . . . 30 (number of days, blank for no auto-delete) </pre>
<p><b>EXS</b></p>	<p>Exclude from source</p> <p>The <b>CM - Exclude Objects (ADBPC7L)</b> panel displays a list of source objects from the compare operation. Use the line commands to edit the list of objects that you want to exclude. See <a href="#">Tips</a>.</p> <pre> ADBPC7L n ----- Exclude Objects ----- Row 1 to 3 of 3 Command ==&gt;                               Scroll ==&gt; CSR  Exclude specification lines for "JSMITH"."SRCIESPEC01" Commands: CANCEL Line commands:   D - Delete  E - Edit  I - Insert  R - Repeat  ? - Show all line commands  Sel T  Qual      Name                Column/      Auth    *  *          *                Constint  Grantee    Level      Message -----&gt;-----&gt;-----&gt;-----&gt;-----&gt;    TB VNRG      TB23367    SA DB23367  TS23367B    SA DB23367  TS23367B </pre>
<p><b>EXT</b></p>	<p>Exclude from target</p> <p>The <b>CM - Exclude Objects (ADBPC7L)</b> panel displays a list of target objects from the compare operation. Use the line commands to edit the list of objects that you want to exclude. See <a href="#">Tips</a>.</p>

**Tips:** When adding and editing objects in an exclude specification on the **CM - Exclude Objects (ADBPC7L)** panel, use the following guidance:

- You can use the wildcard character (\*) when specifying object names.
- You can insert or repeat multiple rows by issuing the corresponding line command followed by the number of rows (up to 99) that you want to insert or repeat. For example, issue I4 to insert 4 new rows, and R6 to repeat the selected row 6 times.
- In the **T** column, specify one of the following two-character codes for the object type or authorization type:
  - AL - Alias
  - CA - Column authorizations
  - DB - Database
  - DT - Distinct type
  - FU - User-defined function

- GV - Global variable
- IX - Index
- RL - Referential constraint
- SC - Schema
- SG - Storage group
- SP - Stored procedure
- SQ - Sequence
- SY - Synonym
- TA - Table authorization
- TB - Table
- TG - Trigger
- TS - Table space
- VA - View authorization
- VW - View

5. Exit back to the **DB2 Object Comparison Tool Menu (GOCMENU)** panel.

## Creating or editing exclude specifications during a comparison

You can create exclude specifications to omit objects from the compare process. If an exclude specification is not already defined, you can create one when selecting the source or target objects. You can also edit an existing exclude specification. Excluded objects are treated as though they do not exist in the source, target, or both.

### Before you begin

This procedure assumes that you are in the process of specifying source or target objects and the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel is displayed.

### Procedure

To create or edit exclude specifications during a comparison:

1. On the **Specify Compare Source (GOC1)** panel or the **Specify Compare Target (GOC1)** panel, complete the fields under **Exclude Specifications:** as follows:

#### Name

Specify the name for the exclude specification. If you want to edit an existing exclude specification, specify the existing name. If you want to create an exclude specification, specify a unique name; the specification will be created.

#### Owner

Optionally specify the specification owner. If this field is left blank, your authorization ID is used as the owner.

#### Edit Objects

Specify YES.

2. Ensure that the option that you want is specified in the **Option** field (according to the task you are completing: “1. Specifying source objects” on page 56 or “2. Specifying target objects” on page 68), and press Enter.
3. Complete the one of the following procedures:

Option	Description
<b>To create an exclude specification</b>	a. On the <b>Insert Exclude Specification (ADB2C22)</b> panel, optionally specify a comment and an <b>Eligible for auto-delete</b> value, and press Enter.

Option	Description
	<p>The following message confirms that the new specification was added:</p> <pre data-bbox="565 243 1472 296">INSERT stmt executed</pre> <p>b. Press exit (PF3).</p> <p>c. On the <b>CM - Exclude Objects (ADBPC7L)</b> panel, specify the objects that you want to include in the exclude specification. Type object names and other information, and use line commands to edit the list of objects. See <a href="#">Tips</a>.</p> <p>d. Press exit (PF3).</p>
<b>To edit an existing exclude specification</b>	<p>a. On the <b>CM - Exclude Objects (ADBPC7L)</b> panel, edit the list of objects that you want to include in the exclude specification. Type over the object names and other information, and use line commands to modify the list. See <a href="#">Tips</a>.</p> <p>b. Press exit (PF3).</p>

**Tips:** When adding and editing objects in an exclude specification on the **CM - Exclude Objects (ADBPC7L)** panel, use the following guidance:

- You can use the wildcard character (\*) when specifying object names.
- You can insert or repeat multiple rows by issuing the corresponding line command followed by the number of rows (up to 99) that you want to insert or repeat. For example, issue I4 to insert 4 new rows, and R6 to repeat the selected row 6 times.
- In the **T** column, specify one of the following two-character codes for the object type or authorization type:
  - AL - Alias
  - CA - Column authorizations
  - DB - Database
  - DT - Distinct type
  - FU - User-defined function
  - GV - Global variable
  - IX - Index
  - RL - Referential constraint
  - SC - Schema
  - SG - Storage group
  - SP - Stored procedure
  - SQ - Sequence
  - SY - Synonym
  - TA - Table authorization
  - TB - Table
  - TG - Trigger
  - TS - Table space
  - VA - View authorization
  - VW - View

### What to do next

Return to [“1. Specifying source objects”](#) on page 56 or [“2. Specifying target objects”](#) on page 68.



## Chapter 7. Managing saved compare results

You can view and modify characteristics of your saved compare results.

### Procedure

To manage saved compare results:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option MR, and press Enter.
2. On the **Manage Saved Compare Results (ADBPMCR)** panel, specify criteria for the saved compare results that you want to view or modify, and press Enter:

```
ADBPMCR ----- Manage Saved Compare Results ----- 16:36
Option ==>

Compare results information:
Owner . . . . . >
Name . . . . . NEW* >

Enter additional selection criteria:
Created within . . . . . 2012 - 2013
Altered within . . . . .
Eligible for auto-delete:
  Within . . . . .
  Next . . . . .
```

3. On the **Manage Saved Compare Results (ADBPCR)** panel, use the line commands to view more detail for, modify, or delete saved compare results:

```
ADBPCR ----- Manage Saved Compare Results -----Row 1 to 14 of 30
Command ==> Scroll ==> PAGE
Line commands:
U - Update DEL - Delete S - Show summary I - Details on results
RPT - Compare Report ? - Show all line
commands

Sel Owner      Name      Eligible for Comment
-----> auto-delete
S  OWN1      NEW1      2012-12-31  My first compare result
   OWN2      NEW2
   OWN3      NEW3
   OWN4      NEW4
   OWN5      NEW5
   OWN6      NEW6
   OWN7      NEW7
```

Figure 46. **Manage Saved Compare Results (ADBPCR)** panel

For example, consider the following two commonly used line commands:

#### S

Use this line command to view the results of a particular compare operation.

The **Compare Results (ADBPCRS)** panel shows the summary of that compare operation:

```

ADBPCRS ----- Compare Results ----- Row 1 to 14 of 30
Command ==> Scroll ==> PAGE

Compare results for "OWN1"."NEW1"
Commands: RPT VERSION SRCEX TGTEX SHOWSOURCE AUTH
Line commands:
EX - Exclude EXS - Exclude from source EXT - Exclude from target
EXA - Exclude Authorizations EXC Exclude Constraints
? - Show all line commands

Sel 0 Target Target Additional Info Compare Action I
    * * * * * * * *
----->-----
    DB DB23367 DB23367 No changes ?
    TS DB23367 TS23367A Altered
    TS Added
    TB VNDRG TB23367 Dropped/created
    IX VNDRG IX23367 Dropped/createdY
    RL VNDRG TCHILD TGTFPARENT Dropped/created

```

Figure 47. **Compare Results (ADBPCRS)** panel

For information about this panel, see [“Compare Results \(ADBPCRS\) panel” on page 149.](#)

## RPT

Use this line command to view a report for a particular compare operation.

The **Compare Report (ADBPCRR)** panel is displayed:

```

ADBPCRR ----- Compare Report ----- 08:22
Command ==> Scroll ==> PAGE

Compare report for "OWN1"."NEW1"

Compare database source(DB23367) and target(DB23367)
No changes to database

Compare tablespace source(DB23367.TS23367A) and target(DB23367.TS23367A)
(A)Field BUFFERPOOL changed from BP0 to BP1
Tablespace will be altered

Tablespace DB23367.TS23367B not found on target
New Tablespace DB23367.TS23367B will be added
Authorizations for Tablespace DB23367.TS23367B will be copied from
source

Compare table source(VNDRG.TB23367) and target(VNDRG.TB23367)
Column LAHMANID
(A)Type changed from VARCHAR(12) to VARCHAR(20)
(D)Nulls/default changed from NOT NULL to WITH DEFAULT NULL
Column DATE_YYYY
A)Type changed from SMALLINT to INTEGER
(D)Column TEAM_NAME added
Table VNDRG.TB23367 will be dropped
Table will be recreated
Table data will not be converted
Not eligible for FORMAT INTERNAL processing

Compare index source(VNDRG.IX23367) and target(VNDRG.IX23367)
Index VNDRG.IX23367 will be dropped by dropping the table
Index will be recreated because the base table will be dropped and recreated

```

Figure 48. **Compare Report (ADBPCRR)** panel

## Compare Results (ADBPCRS) panel

The **Compare Results (ADBPCRS)** panel shows a summary of the selected compare operation. Open this panel by specifying line command S on the **Manage Saved Compare Results (ADBPCR)** panel.

```

ADBPCRS ----- Compare Results ----- Row 1 to 14 of 30
Command ===>                               Scroll ===> PAGE

Compare results for "OWN1"."NEW1"
Commands: RPT VERSION SRCEX TGTEX SHOWSOURCE AUTH
Line commands:
EX - Exclude EXS - Exclude from source EXT - Exclude from target
EXA - Exclude Authorizations EXC Exclude Constraints
? - Show all line commands

Sel  0  Target  Target  Additional Info  Compare  I
     *  *      *                               *        *
----->-----
DB          DB23367
TS DB23367  TS23367A
TS
TB VNDRG    TB23367
IX VNDRG    IX23367
RL VNDRG    TCHILD          TGTFPARENT
                                No changes  ?
                                Altered
                                Added
                                Dropped/created
                                Dropped/createdY
                                Dropped/created

```

Figure 49. **Compare Results (ADBPCRS)** panel

This panel includes the following columns:

### Sel

An input field for entering a line command.

### Object

The type of object that was compared. This type can be one of the following values:

#### AL

Alias

#### AR

Auxiliary table

#### CL

Clone table

#### DB

Database

#### DT

User-defined type

#### FU

Function

#### GV

Global variable

#### IX

Index

#### MK

Column mask

#### PK

Rebind package

#### PL

Rebind plan

#### PM

Row permission

<b>RL</b>	Referential constraint
<b>SC</b>	Schema
<b>SG</b>	Storage group
<b>SP</b>	Stored procedure
<b>SQ</b>	Sequence
<b>SY</b>	Synonym
<b>TB</b>	Table
<b>TG</b>	Trigger
<b>TS</b>	Table space
<b>VW</b>	View

**Target Schema**

The target object owner.

**Target Name**

The target object name.

**Additional Information**

The referential constraint name if the change updates a referential constraint.

**Compare Action**

The type of change to the object.

**Implicit**

An indication of whether the change includes an implicit drop or an implicit drop and re-create.

Use the following commands to view additional information:

**SHOWSOURCE**

Displays the source object that is associated with the target in the **Additional Information** column.

**VERSION**

Displays the following fields:

**Version**

Shows the version number of native stored procedures and PL/SQL functions. This column is displayed in place of the **Compare Action** column.

**Active version**

Indicates an active version of specific native stored procedures and PL/SQL functions. This column is displayed in place of the **I** column.

---

## Chapter 8. Ignoring changes

When you compare objects, the report might include object changes that you do not want. You can designate those object changes as *ignore changes*, or changes to ignore in subsequent compare processing. Specifying such changes is called an *ignore changes specification* and requires saved compare results. Within the saved compare results, you select the object changes that you want to ignore. The selected object changes are reported as part of the compare process, but no SQL statements are generated for the changes.

### About this task

Only changes that report differences between a source object and a target object can be ignored. Added and dropped objects can be excluded from compare processing but not ignored.

Ignore changes for tables, global variables, indexes, and user-defined distinct types (UDT) are supported. Changes to temporal tables, history tables, materialized query tables (MQT), and hash organization cannot be ignored.

You must specify each object change that you want ignored. Related object changes are not automatically ignored.

**Tip:** Use caution when selecting object changes to be ignored. Many objects and fields in the Db2 catalog records are interdependent. When one change is ignored, another change might be invalid if it is not also ignored. For example, if a change to the number of table space partitions is ignored, other object changes related to partitioning must also be ignored. In this case, changes to the number of partitions in the table and adding or deleting a limit key must also be ignored. LOB columns and LOB objects are another example of an interdependency. If adding a LOB column to a table is ignored, adding an explicit auxiliary table for the column must also be ignored, and its explicit LOB table space and index be excluded from compare.

### Procedure

To ignore changes:

1. Create an [ignore changes specification](#).
2. Update the specification as needed during the compare process. See [“Modifying ignore changes specifications”](#) on page 152.

---

## Creating ignore changes specifications

An *ignore changes specification* designates the changes that you want ignored in subsequent compare processing.

### Before you begin

You must have saved compare results from which you want to select object changes to ignore.

### Procedure

To create an ignore change specification:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option MR (Manage saved compare results), and press Enter.
2. On the **Manage Saved Compare Results (ADBPMCR)** panel, specify criteria to find the saved compare result that you want, and press Enter.
3. On the **Manage Saved Compare Results (ADBPCR)** panel, issue the RPT command., and press Enter.
4. On the **Compare Report (ADBPCRR)** panel, issue the I line command next to the compare changes that you want to ignore.

```

ADBPCRR ----- DD1A Compare Report ----- 08:22
Command ==>>>                               Scroll ==>> PAGE

Compare analysis report for "OWN1"."NEW1"
Commands: CONTINUE IGNOREALL RESETALL
Line commands:
  I - Ignore change R - Reset

Sel S Report

----->
  Compare database source(DB23367) and target(DB23367)
    No changes to database

  Compare tablespace source(DB23367.TS23367A) and target(DB23367.TS23367A)
    (A)Field BUFFERPOOL changed from BP0 to BP1
    Tablespace will be altered

  Tablespace DB23367.TS23367B not found on target
  New Tablespace DB23367.TS23367B will be added
  Authorizations for Tablespace DB23367.TS23367B will be copied from
source

  Compare table source(VNDRG.TB23367) and target(VNDRG.TB23367)
    Column LAHMANID
  I      (A)Type changed from VARCHAR(12) to VARCHAR(20)
        (D)Nulls/default changed from NOT NULL to WITH DEFAULT NULL
    Column DATE_YYYY
  I      (A)Type changed from SMALLINT to INTEGER
        (D)Column TEAM_NAME added
    Table VNDRG.TB23367 will be dropped
    Table will be recreated
    Table data will not be converted
    Not eligible for FORMAT INTERNAL processing

  Compare index source(VNDRG.IX23367) and target(VNDRG.IX23367)
    Index VNDRG.IX23367 will be dropped by dropping the table
    Index will be recreated because the base table will be dropped and recreated

```

Figure 50. Compare Report panel (ADBPCRR)

When you press Enter, the status column for the change is updated to I.

5. Issue the CONTINUE command, and press Enter.
6. On the **Create Ignore Specification (ADB2C22)** panel, type an owner name and specification name. The ignore changes specification is created.

## What to do next

You can now reference this ignore changes specification when you compare objects.

### Related information

[Managing ignore changes \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## Modifying ignore changes specifications

You can update an *ignore changes specification*, which is a list of object changes that are to be ignored during the compare process.

### Procedure

To modify ignore changes specifications:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option 4, and press Enter.
2. On the **Specify Compare Ignores (GOC4)** panel, specify one of the following sets of information:

Edit preference	Fields to specify
<b>To edit the specification as a list:</b>	Under <b>Ignore Changes Specification</b> , specify an owner and name for the existing ignore changes specification and specify YES for <b>Edit Ignore Changes Specification</b> .
<b>To edit the compare result with the ignore changes marked:</b>	Under <b>Saved Compare Results</b> , specify an owner and name for the saved result, and specify YES for <b>Display using a saved compare result</b> .

```

Compare ----- Specify Compare Ignores -----
Option ==>

Ignore Fields Specification:
Owner . . . . . > (? to look up)
Name . . . . . > (? to look up)
Data Set:
  Data Set Name . .
Options:
  Edit Ignore Fields Specification . . . NO (Yes/No)

Ignore Changes Specification:
Owner . . . . . OWN1 > (? to look up)
Name . . . . . NEW1 > (? to look up)
Edit Ignore Changes Specification . . . YES (Yes/No)
Display using a saved compare result . . NO (Yes/No)
  Saved Compare Results:
    Owner . . . . . OWN1 > (? to look up)
    Name . . . . . ISPEC01 > (? to look up)

```

Figure 51. **Specify Compare Ignores (GOC4)** panel

- If you specified YES for **Edit Ignore Changes Specification**, modify the list as needed on the **Ignored Changes List (ADBPCICL)** panel:

```

ADBPCICL ----- Ignored Changes List ----- Row 1 to 14 of 30
Command ==> Scroll ==> PAGE

Ignored changes for "OWN1"."ISPEC01"
Line commands:
D - Delete

Sel T Target Target Attribute Additional Info
Qual Name
-----> -----> ----->
      DB23367 TS23367A BUFFERPOOL
TB VNRG TB23367 Data type LAHMANID
TB VNRG TB23367 Data type DATE_YYYY

```

Figure 52. **Ignored Changes List (ADBPCICL)** panel

You can add object changes or delete an object change that is listed. When you add object changes, those changes are ignored. Ignored changes are not applied to the target objects.

To add an object change, add the object type and name in the blank line after the column headings and before the listed change objects. Use the wildcard character (\*) in the **Target Qualifier** or **Target Name** column to indicate that all changes for matching objects are to be ignored. For example, if you specify new\*, objects that meet the wildcard specification new\* are still processed; however, all changes for these objects are ignored. If the qualifier or name does not include a wildcard character, the wildcard character (\*) is appended to the qualifier or name. If the **Target Qualifier** or **Target Name** column is blank, an asterisk (\*) is substituted.

- If you specified YES for **Display using a saved compare result**, use the line commands to modify the ignore changes as needed on the **Compare Report (ADBPCRR)** panel. See step “4” on page 151 in [“Creating ignore changes specifications”](#) on page 151.

**Related tasks**

[“Creating ignore changes specifications” on page 151](#)

An *ignore changes specification* designates the changes that you want ignored in subsequent compare processing.

---

## Chapter 9. Applying changes to target objects

After you run a comparison, you can apply the changes in the compare report to the target object or objects.

### About this task

Object Comparison Tool supports changes to implicit LOB and XML table spaces. When tables have implicit LOB or XML table spaces defined, Object Comparison Tool generates multiple image copies, which requires that a template be used for the SYSCOPY data set. You can define your own SYSCOPY template or Object Comparison Tool can use the default. The default template is:

```
DSN(&US. .&SSID. .&DB. .&SN. .&UQ)
```

The default template for clones is:

```
DSN(&US. .&SSID. .&DB. .&SN. .CLONE. &UQ)
```

For more information about templates and how to associate them with certain data sets, see [Associating templates with data sets \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

### Procedure

To apply changes to the target objects:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, select option **5 - Generate compare job**, and press Enter.
2. On the **Generate Compare Jobs (GOC5)** panel, set the **Generate Apply Job** field to Yes and set any other fields as needed, and press Enter.
3. On the **Specify Data Set Name for Apply Jobs (GOC5AJ)** panel, enter the name of the data set where you want the apply jobs generated.
4. Edit the generated compare job as needed, and submit the job.
5. Check the output to confirm that the job completed successfully.
6. Run the generated apply job to make the changes to your target objects or if you requested a work statement list (WSL), [Run the WSL to apply the changes](#).

---

## Apply jobs

If you requested multiple apply jobs when running Compare ( **Generate apply jobs** = YES and **Generate one job** = NO or PROCESS on the **Generate Compare Jobs (GOC5)** panel), Object Comparison Tool produces a number jobs with predefined names. The specific jobs generated depends on the value of the **Generate one job** option.

If **Generate one job** = NO, Object Comparison Tool produces the following apply jobs:

### T10Unnnn

UNLOAD jobs, which contain the following steps:

- Step 1 issues a Db2 command to place the table space in read-only status.
- Step 2 deletes the SYSREC and SYSPUNCH data sets, if they already exist.
- Step 3 unloads the given table.
- Additional steps create Db2 LOAD utility control statements for the unloaded data.

The generated UNLOAD jobs can be run in parallel. The space parameter for the SYSREC data set in step 3 is derived from the RUNSTATS statistics in the catalog and from the high-used-RBA value of the data set from the table space. If the target version files are not created from a Db2 catalog, evaluate

and possibly correct the space parameter, because no space data is available and default sizes are used.

**Note:** Version files that are created outside the scope of the Db2 catalog, such as those created from DDL, do not represent the same information that is found in the Db2 catalog.

#### **T20DROP**

DROP job. This job includes statements that must be run before objects are created. It can include DROP, RENAME, and ALTER statements. This job also includes DD statements that reference all unload data sets to ensure that all UNLOAD jobs have run before the objects are dropped.

#### **T30CREAT**

CREATE job. This job creates the objects and their authorizations. It can also contain ALTER SQL statements.

If an inline utility needs to be run between DDL statements, the T30CREAT job is split into multiple jobs, named T30Cnnnn.

#### **T40STOP**

STOP job. This job stops page sets. It can contain other SQL and utility statements.

#### **T50ALTER**

ALTER job. This job can contain CREATE and ALTER SQL statements.

If an inline utility needs to be run between DDL statements, the T50ALTER job is split into multiple jobs, named T50Annnn.

#### **T60START/T61START**

START jobs. These jobs start page sets. The T61START job is for clone objects.

#### **T70Rnnnn**

RELOAD jobs.

#### **T71Rnnnn**

REORG jobs. These REORG jobs remove REORG-pending conditions. All REORG statements are combined into a single job if SHRLEVEL CHANGE is specified and the mapping table name is provided.

#### **T72REBLD**

REBUILD jobs.

#### **T80Rnnnn**

REORG jobs. These REORG jobs fully implement the effects of the changes (for example, space parameter changes). All REORG statements are combined into a single job if SHRLEVEL CHANGE is specified and the mapping table name is provided.

#### **T81REBLD**

REBUILD job. This job rebuilds indexes.

#### **T85REFR**

REFRESH job. This job refreshes tables.

#### **T89POSTI**

Jobs to grant authorizations on tables and to reload accelerators.

#### **T90RB**

Rebind jobs.

#### **Utilities**

After the LOAD jobs have run, optional jobs are created to run CHECK (T71CHECK), COPY (T73IMC), and RUNSTATS (T87RUNST). You can run these jobs in parallel.

If **Generate one job** = PROCESS, Object Comparison Tool produces the following apply jobs:

#### **T10U0001**

UNLOAD job.

#### **T20DROP**

DROP job.

**T30CREAT**

CREATE job.

**T40STOP**

STOP job.

**T50ALTER**

Alter job.

**T60START**

START job.

**T70R0001**

Reload job.

**T89POSTI**

Table GRANTs and reload accelerator job.

**T71R0001**

REORG job.

**T72REBLD**

REBUILD job.

**T73CHECK**

CHECK job.

**T74IMC**

IMAGECOPY job.

**T80R0001**

REORG job.

**T81B0001**

REBUILD job.

**T87RUNST**

RUNSTATS job.

**Note:** If the number of steps in an UNLOAD, reload, or REORG job (T10U0001, T70R0001, or T80R0001) exceed a maximum of 255, a second job corresponding to each process (T10U0002, T70R0002, or T80R0002) is generated.

## Running a work statement list to apply changes

---

Apply changes are placed in a work statement list (WSL) if you specified **Generate apply jobs** = YES and **As work statement list** = YES on the **Generate Compare Jobs (GOC5)** panel. You must use Db2 Admin Tool panels to run the work statement list.

### About this task

When you run a WSL, you can generate single or multiple apply jobs for all operations. When no UNLOAD, RELOAD, or REORG operations are required, or when a single UNLOAD, a single RELOAD, or a single REORG operation is required, only a single job is generated for the WSL. When multiple UNLOAD, RELOAD, or REORG operations are required, and you want to generate a single apply job for all operations rather than a separate job for each operation, you must specify that a single job be generated when you run the WSL. If you specify one job, table GRANT statements are generated by Object Comparison Tool after tables, related indexes, and foreign keys are created, and after all rows are reloaded.

**Note:** In the LOAD step, the TEMPLATE statement for the SYSREC data set is used when you select HPU for the unload process. Because Object Comparison Tool does not know if HPU will be used as the unload method before the WSL is run, the template might seem unneeded when the WSL is generated. However, the TEMPLATE SYSREC statement is used if you select HPU as the unload process when you build the job in Db2 Admin Tool.

## Procedure

Use Db2 Admin Tool. See [Running a WSL\(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

## Related reference

[“Compare job options” on page 87](#)

When you generate a compare batch job, you can specify a number of options to control the behavior of the comparison operation and job. These options are listed on the **Generate Compare Jobs (GOC5)** panel.

## Related information

[Work statement lists \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## Chapter 10. Managing and restoring dialogs

In Object Comparison Tool, a *dialog* is a set of user selections for compare batch jobs. You can restore, rename, and delete any previously saved dialogs.

### About this task

When you restore a dialog, it is restored in the display mode that was used when the dialog was saved. For example, if the dialog was saved in advanced mode, it is restored in advanced mode. If a dialog was saved before the display mode feature was introduced in Object Comparison Tool, the dialog is restored in advanced mode.

### Procedure

To manage and restore dialogs:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option **M**, and press Enter.
2. On the **Saved Dialogs (ADB2SD)** panel, specify the prefix for the data set with the saved dialogs, and press Enter.

```
Compare ----- Saved Dialogs ----- 14:05
Option ==>

Enter the prefix for saved dialog data sets:
Prefix ==> NBRON.ADMIN.SAVEDLGS.APAR.B37
```

Figure 53. **Saved Dialogs (ADB2SD)** panel

The **Manage/Restore Dialogs (ADB2SDM)** panel lists all the saved dialogs in the specified dialog data set:

```
ADB2SDM n ----- Manage/Restore Dialogs ----- Row 1 of 2
Command ==> Scroll ==> PAGE

Line commands: D - Delete R -Rename S - Select ? - Show all line commands

S Name      Description                               Created      Id
*          *                               *           *
-----
TEST01     SOURCE IS DDL W/ADD.COLUMN                2003/07/09  NBRON
PREVTEST   DDL&DB2 W/STOGRUP                        2003/09/09  NBRON
```

Figure 54. **Manage/Restore Dialogs (ADB2SDM)** panel

3. Use the line commands to take the following actions as needed:

#### To restore a dialog:

Issue the S line command next to the dialog. The selections for the corresponding dialog are restored, and you can immediately run the batch job.

**To rename a dialog:**

Issue the R line command next to the dialog. On the subsequent **Member Rename (ADB2SDR)** panel, enter the new dialog name, and press Enter.

**To delete the dialog:**

Issue the D line command next to the dialog. The dialog is immediately deleted when you press Enter.

## Saving dialogs

---

In Object Comparison Tool, you can save the current compare batch job selections, including the options on the **Generate Compare Jobs (GOC5)** panel and its subordinate panels, for later retrieval. This set of user selections is called a *dialog*. You can later restore this dialog or use this dialog to compare multiple sources and targets.

**About this task**

The display mode that is used when you save a dialog is the same display mode that is used when the dialog is restored. For example, if you are using simple display mode when you save the dialog, it is restored in simple display mode.

**Procedure**

To save a dialog:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option **S**, and press Enter.
2. On the **Save Dialog (ADB2SDS)** panel, specify the following information, and press Enter to save the dialog:

```
ADB2SDS ----- Save Dialog ----- 10:35
Enter/verify the following:
Prefix    ===> NBRON.ADMIN.SAVEDLGS.APAR.B37
Name      ===>
Description ===>
```

**Prefix**

Specify a unique qualified name for a collection of saved dialogs. This name is used as a prefix for one or more data sets in which saved dialogs are stored.

**Name**

Specify a name that identifies the dialog within the collection of dialogs that is identified by the prefix. If you use a duplicate dialog name within the prefix, the existing dialog is replaced. (A confirmation is displayed to confirm that you want to replace the dialog.) Otherwise, a new member is created.

**Description**

Optionally, enter a description of the dialog.

**What to do next**

If you later want to retrieve this dialog, complete the steps in [Chapter 10, “Managing and restoring dialogs,”](#) on page 159.

**Related tasks**

[“Comparing multiple sources and targets”](#) on page 165

You can compare sources and targets from one or more saved dialogs. For example, you can compare multiple saved targets to one saved source. This functionality is called *MultiCompare*.

**Related information**

[“Simple and advanced display modes” on page 89](#)



---

# Chapter 11. Converting version files to the latest Db2 version

To save time during compare processing, you can convert an older version file to the latest Db2 version that is supported by Db2 Object Comparison Tool by using the Version File Conversion utility. For example, if a version file was created for Db2 11 and you are running Db2 12, you can convert the version file for Db2 12.

## About this task

A *version file* contains object information that is extracted by Db2 Object Comparison Tool during the compare process; this file is then used during the comparison.

Each version of Db2 Admin Tool and Object Comparison Tool supports a range of version file levels. If an unconverted version file that is within the supported range is passed to Object Comparison Tool, the tool can process the version file but requires extra conversion time. The original version file is unchanged by Object Comparison Tool

Alternatively, the Version File Conversion utility converts a version file within the supported range to the current level before it is processed. A converted version file can be used by Object Comparison Tool immediately, saving the time that the conversion would require. This utility converts the version file catalog records to the latest supported version of Db2 and permanently upgrades the file to the current supported level.

You can use the Version File Conversion utility to convert version files that are stored in sequential data sets, PDS data sets, and PDSE data sets and base versions that are stored using Db2 Admin Tool.

## Procedure

To convert version files to the latest Db2 version, run the Version File Conversion utility by using a batch job. Sample JCL is provided in the SAMP library ADBVFCO member. Modify this sample as follows:

- Tailor the JCL to your installation before use.
- Specify input to the Version File Conversion utility by using the VFLIST DD statement.
- For a sequential to sequential data set conversion, specify the existing version file with the VFOLD keyword and the new version file with the VFNEW keyword. For example:

```
VFOLD='C386799.DT27760.C.VERSION(SOURCE0) ', VFNEW='C386799.DT27760.C.SOURCE0N' ;  
VFOLD='Z9.ORDER.VERSION(MAY29) ', 'Z9.ORDER.VERSION(JUNE17) ' ;
```

- Ensure that any sequential data sets named are allocated and cataloged first. If VFOLD or VFNEW are PDS data sets, the member names must be included. The VFNEW PDS member name can be a new name; however, VFOLD and VFNEW cannot have the same name. The data set must not contain any sequence numbers in columns 72 - 80.
- Use one of the following two ways to convert a base version file: by VOWNER and VNAME or by VID. Change Management option 4.1 lists all base version files that are stored in Db2 Admin Tool. VOWNER, VNAME, and VID are listed for each base version. A base version file can be described by VOWNER=*owner\_name*, VNAME=*version\_file\_name*; or by VID=*nnn*. The converted base version replaces the original version only if no errors are detected.
- To convert a base version file and store it in a sequential, PDS, or PDSE data set, specify the base version file by using VOWNER and VNAME or by using VID and specify the output data set by using VFNEW. Object Comparison Tool converts the base version file to the current level and writes it to the data set specified by VFNEW. The original base version file remains unchanged.

## **Results**

The Version File Conversion utility processes each version file sequentially. If an error is detected, an error message is issued and processing is halted for the version file in error. The next version file is then processed.

## Chapter 12. Comparing multiple sources and targets

You can compare sources and targets from one or more saved dialogs. For example, you can compare multiple saved targets to one saved source. This functionality is called *MultiCompare*.

### Before you begin

You must have saved dialogs from previous comparisons. See [“Saving dialogs” on page 160](#).

### Procedure

To compare multiple sources and targets:

1. On the **DB2 Object Comparison Tool Menu (GOCMENU)** panel, specify option **MC**, and press Enter.
2. On the **Saved Dialogs (GOCMC1)** panel, specify the prefix of the data set where the dialogs are saved and press Enter.

```
Compare ----- Saved Dialogs ----- 14:05
Option ==>

Enter the prefix for saved dialog data sets:
Prefix ==> RAJESHR.ADMIN.SAVEDLGS
```

Figure 55. *Saved Dialogs (GOCMC1)* panel

3. On the **Manage Dialogs MultiCompare (GOCMC)** panel, select the dialogs that you want to include in the comparison by using one of the following commands:

#### RUN

Runs the compare process for all included dialogs [those with a Y in the **I (Indicator)** column].

Before issuing RUN, use the IC (include) and XC (exclude) line commands to ensure that the dialogs you want are included.

#### RUNALL

Runs the compare process for all the listed dialogs except the excluded dialogs [those with a N in the **I (Indicator)** column].

Before issuing RUNALL, use the IC (include) and XC (exclude) line commands to ensure that the dialogs that you do not want to use are excluded.

**Attention:** RUN and RUNALL will fail if any of the selected dialogs have an incomplete status.

```
Compare ----- Manage Dialogs MultiCompare -----
Command ==>                               Scroll ==> PAGE

Commands:   RUN   RUNALL

Line commands: S - Select      D - Delete      I - Interpret
               RL - Repeat/change location  Rnn - Repeat 'nn' times
               MS - Modify Source           MT - Modify Target
               IC - Include in Compare      XC - eXclude from Compare

S   Name      Location      Description Created   Id      I Status
----->-----
TARGET1 STPLEX4A_DSN8      AQ238S1 ON   2004/08/13 RAJESHR Y Incomplete
```

Figure 56. *Manage Dialogs MultiCompare (GOCMC)* panel

For information about this panel, see [“Manage Dialogs MultiCompare \(GOCMC\) panel” on page 167](#).

4. On the **MultiCompare / Select Dialog (GOC2MCMC)** panel, specify whether you want to compare one source to multiple targets.

```
Compare -----MultiCompare/Select Dialog ----- 14:18
Selecting this option enables the compare process to use the source of the
following dialog to be compared against all the targets selected.

Compare one source to multiple targets? ==> (Yes/No)
Dialog Name                          ==> <Dialog Name>
```

Figure 57. **MultiCompare / Select Dialog (GOC2MCMC)** panel

- To compare one source to multiple targets, specify Y and the dialog name that contains the source that you want to use, and press Enter. The source in the specified dialog is used in each compare process instead of the source in the saved dialogs.
  - To run Compare with the source and target that are defined in each saved dialog, specify N, and press Enter. The source that is specified for each of the saved dialogs is used.
5. On the **Generate Compare Jobs (GOC5)** panel, specify the compare job options, and press Enter.

Consider the following restrictions and behavior:

- The **Generate online** option is automatically set to No. Generating a work statement list online is not supported with MultiCompare.
- If the **Single compare job** option is Yes, the number of steps in the job exceeds 255, and the **Member name** value exceeds six characters, the member name is truncated to allow the addition of a numeric suffix.
- If the **Single compare job** option is No and the **Member name** value exceeds five characters, the member name is truncated to allow the addition of a numeric suffix. This suffix indicates the compare job for each target.
- The **As work statement list** option must be set to Yes for MultiCompare.
- The option to replace work statement lists is not supported when generating apply jobs. Work statement lists are automatically appended. To replace work statement lists for more than one dialog, you must run the jobs individually.
- The work statement list name is derived as Work List Namennn, where nnn uniquely identifies the work statement list for each target.

For descriptions of the options on the **Generate Compare Jobs (GOC5)** panel, see [“Compare job options”](#) on page 87.

6. Complete the options on any subsequent pop-up windows.
7. Edit the generated JCL job as needed and submit it to run the comparison.

## Manage Dialogs MultiCompare (GOCMC) panel

The **Manage Dialogs MultiCompare (GOCMC)** panel displays a list of saved dialogs. You can select from this list dialogs that you want to use in a comparison.

```
Compare ----- Manage Dialogs MultiCompare -----
Command ==>                                         Scroll ==> PAGE

Commands:  RUN  RUNALL

Line commands: S - Select      D - Delete      I - Interpret
               RL - Repeat/change location  Rnn - Repeat 'nn' times
               MS - Modify Source          MT - Modify Target
               IC - Include in Compare     XC - eXclude from Compare

S   Name      Location      Description Created   Id       I Status
----->-----
TARGET1 STPLEX4A_DSN8      AQ238S1 ON  2004/08/13 RAJESHR Y Incomplete
```

Figure 58. **Manage Dialogs MultiCompare (GOCMC)** panel

This panel lists the following information for each dialog:

**Name**

The name of the dialog.

**Location**

The location where the target points if Db2.

**Description**

The description of the saved dialog.

**Created**

The date that the dialog was created.

**Id**

The user ID of the person who created the dialog.

**I (Indicator)**

An indication (Y or N) of whether the dialog is to be included in the compare when the RUN or RUNALL commands are issued. This field is set by using the IC and XC line commands.

**Status**

The status of the dialog. A status of INCOMPLETE means that not all source and target definitions are provided for that dialog and the dialog will not be included in the compare even if the **I (Indicator)** field is set to Y.

From this panel, you can perform the following actions on a dialog by using the listed line command:

**S - Select**

Selects the dialog to be included in the comparison.

**D - Delete**

Deletes the dialog from the library.

**I - Interpret**

Displays the source and target details for the dialog, for example:

```

Compare ----- Interpret Dialog ----- Row 1 to 11 of 11
                                         Scroll ==> PAGE

      Type                Name
-----
Source Mask data set      None specified
        Ignore data set   Using defaults
        Version data set  D3410.VER.DSN
        Location          STPLEX4A_DSN7
        Table space       VNRJPD.VNRJPTS
        Table space       AHXFLWDB.AHX3UJWU
        Table space       ADBD4BAS.ADBS4BAS
        Table space       ADBDCH3.ADBSCH3
Target Version data set  D3410.TGT.VER
        Location          STPLEX4A_DSN7
        Table space       VNRJPD.CQ289TS

```

Figure 59. **Interpret Dialog (GOCMI)** panel

This panel indicates whether the source and target definitions are from DDL or the Db2 catalog. If the source or target is from the Db2 catalog, this panel lists all objects in the definition and their type.

**RL - Repeat/change location**

Replicates a dialog and changes the location of the target. When you specify RL, the **Distributed DB2 Systems (ADB2DDF)** panel displays the remote Db2 subsystems that are available from the current Db2 subsystem:

```

DB2 Admin ----- Distributed DB2 Systems -----
Command ==>                                         Scroll ==> PAGE

Select by typing '+'
Select the location you wish to use:                DB2 System: DD1A
                                                    DB2 SQL ID: ADM001

Line commands:
  S - Use DDF to access remote catalog  CO - Connect to remote subsystem
  DIS - Display threads for remote system

Select Location
-----
  STPLEX4A_DSN7
  SQLVM6
  STLEC1

```

Figure 60. **Distributed DB2 Systems (ADB2DDF)** panel

Select the new location or locations for the dialog by entering a plus sign (+) in the **Select** field. You can select multiple locations at one time. Press End. The **Repeat Dialog/Change Location (GOCMRL)** panel displays the new dialog with the new location:

```

Compare----- Repeat Dialog/Change Location --- Row 1 to 1 of 1
Command ==>                                         Scroll ==> PAGE

Commands: NEXT

Specify output compare version file:
  Data set prefix: J148286.OC                      (Prefix of target version files)
  Data set suffix: TGTVF                          (Optional suffix)

S Target Location  Dialog  Dialog Description
* * *
-----
  STPLEX4A_DSN8   TEST1

```

Figure 61. **Repeat Dialog/Change Location (GOCMRL)** panel

You can select and edit the dialog name and description. When you are finished editing, issue the NEXT command to return to the **Manage Dialogs MultiCompare (GOCMC)** panel.

**Rnn - Repeat *nn* times**

Replicates a dialog multiple times. The **Repeat Dialog/Change Location (GOCMRL)** panel displays the replicated dialogs when this command is invoked. You can edit the dialog target location, name, and description. When you are finished with your edits, issue the NEXT command to return to the **Manage Dialogs MultiCompare (GOCMC)** panel.

**MS - Modify Source**

Modifies the source details. The **Specify Compare Source (GOC1)** panel is displayed.

**MT - Modify Target**

Modify the target details. The **Specify Compare Target (GOC1)** panel is displayed.

**IC - Include in Compare**

Selects the dialog to include in the compare process. Upon selection, the **I (Indicator)** field is set to Y. The dialog status must be Complete for it to be included in the compare process; otherwise, an error occurs.

**XC - Exclude from Compare**

Selects a dialog to exclude from the compare process when the RUNALL command is issued. You can also use this command to reverse the IC command and change the **I (Indicator)** field from Y to N.

**Related tasks**

[“Comparing multiple sources and targets” on page 165](#)

You can compare sources and targets from one or more saved dialogs. For example, you can compare multiple saved targets to one saved source. This functionality is called *MultiCompare*.

[“Saving dialogs” on page 160](#)



---

## Chapter 13. Batch DDL file extraction program

The DDL file extraction program interprets a source file of DDL statements that define Db2 objects. The program generates a *version file*, which contains records that are similar in format to those in the Db2 catalog that defines the same objects.

To effectively compare the input DDL objects to different versions of the same objects, you can use the version file as input to the batch Compare program.

**Restriction:** Version files are compressed internally and should not be created with DFSMS compression, because GEN and the DDL reader opens them for update, which is not allowed for DFSMS compressed data sets. Db2 Administration Tool or Db2 Object Comparison Tool jobs will receive S213-C8 abends if the version file data sets are defined with DFSMS compression.

The batch DDL file extraction program is run and a report is produced when you specify DDL files on the **Specify Source DDL File (GOC11)** panel or the **Specify Target DDL File (GOC11)** panel and run a compare job.

The source of the DDL statements can be:

- A sequential file that contains SQL statements
- An extract from a Db2 catalog of some set of Db2 objects and dependencies

### Related concepts

[“Components of the comparison process” on page 17](#)

Db2 Object Comparison Tool compares objects by reading the Db2 catalog or DDL files. Object Comparison Tool produces comparison reports and then optionally generates either JCL jobs or work statement list (WSL) tasks with changes for the target objects.

[“Batch compare program” on page 179](#)

### Related tasks

[“Specifying a DDL file for the source or target definition” on page 57](#)

Db2 Object Comparison Tool can use a file that contains data definition language (DDL) for the definitions of the source or target objects. Object Comparison Tool processes everything in the DDL file; objects are not selected based on type or name. If your DDL defines a single table, only that table is used.

### Related reference

[“Supported SQL statements for DDL file extraction” on page 171](#)

The DDL file extraction program supports a subset of the SQL statements that are supported by Db2 for z/OS.

[“Batch DDL file extraction program report format” on page 174](#)

The report that the batch DDL file extraction program produces begins with a header and the IBM copyright statement. The copyright statement is followed by a line that indicates the version of Db2 startup parameters that are used when the extraction program is processing statements from the input stream.

---

## Supported SQL statements for DDL file extraction

The DDL file extraction program supports a subset of the SQL statements that are supported by Db2 for z/OS.

DDL statements that are submitted for processing by the DDL file extraction program must be in the format that is supported by SPUFI or DSNTEP2:

- Input must be in columns 1-72.
- Phrases can span records. For example, column 1 of an input record immediately follows column 72 of the previous record.
- Comments can be included and are indicated by two consecutive dashes (--).

- The generated statement terminator was the question mark ( ? ) for releases earlier than Db2 Admin Tool Version 11.1 and is the grave accent ( ` ) for Db2 Admin Tool Version 11.1 and later releases.

**Restriction:** The DDL reader does not communicate with Db2. Therefore, the DDL reader is unable to get the defaults that are established by the user for table space buffer pool, compression and index buffer pool, and pad index. The defaults that are used are those used before DB2 9.

The following SQL statements are supported:

- ALTER DATABASE
- ALTER FUNCTION
- ALTER INDEX
- ALTER PROCEDURE
- ALTER SEQUENCE
- ALTER STOGROUP
- ALTER TABLE

**ALTER TABLE restrictions:**

- The ALTER statement is not supported for auxiliary tables.
- Constraint names are not compared (and differences not reported) because constraint names can be either explicitly specified or, if they are not explicitly specified, generated by Db2. If the constraint names are generated by Db2, the constraint names could be different between source and target, even if the DDL for the object might be the same for source and target.
- ALTER TABLE ROTATE PARTITION statements have the following restrictions:
  - The maximum number of ALTER TABLE statements that can be processed to rotate partitions is  $n-1$ , where  $n$  is the number of partitions.
  - If a rotate operation occurred and new partitions have also been added, the rotate operation is not detected.
  - If a rotate operation occurred and the limit keys were altered, the rotate operation might not be detected.

- ALTER TABLESPACE
- COMMENT ON

**COMMENT ON restriction:** The COMMENT ON statement is not supported for auxiliary tables.

- COMMIT
- CREATE ALIAS
- CREATE AUX TABLE
- CREATE DATABASE
- CREATE DISTINCT TYPE
- CREATE FUNCTION
- CREATE INDEX
- CREATE PROCEDURE
- CREATE SEQUENCE
- CREATE STOGROUP
- CREATE SYNONYM
- CREATE TABLE

**CREATE TABLE restrictions:**

- For CREATE TABLE LIKE statements, the DDL must also include the definition of the table in the LIKE clause.

– Constraint names are not compared (and differences not reported) because constraint names can be either explicitly specified or, if they are not explicitly specified, generated by Db2. If the constraint names are generated by Db2, the constraint names could be different between source and target, even if the DDL for the object might be the same for source and target.

- CREATE TABLESPACE
- CREATE TRIGGER
- CREATE VARIABLE

If a dependent object such as a procedure (native stored procedure), PL/SQL function, trigger, view, column mask, or row permission in the data set references a global variable, the CREATE VARIABLE statement should be included in the DDL data set.

- CREATE VIEW
- DROP ALIAS
- DROP DATABASE
- DROP DISTINCT TYPE
- DROP INDEX
- DROP SEQUENCE
- DROP SPECIFIC FUNCTION
- DROP STORED PROCEDURE
- DROP SYNONYM
- DROP TABLE
- DROP TABLESPACE
- DROP TRIGGER
- DROP VARIABLE
- DROP VIEW
- GRANT collection privileges
- GRANT database privileges
- GRANT distinct type or JAR privileges

**GRANT JAR restriction:** The GRANT USAGE ON JAR statement is not supported in change management or in Db2 Object Comparison Tool.

- GRANT function or procedure privileges
- GRANT package privileges
- GRANT plan privileges
- GRANT schema privileges
- GRANT sequence privileges
- GRANT system privileges
- GRANT table or view privileges
- GRANT use privileges
- GRANT variables
- **GRANT restriction:** For objects that exist on both the source and the target, Db2 Object Comparison Tool compares and reports the authorization differences, but does not propagate the differences from the source to the target. Db2 Object Comparison Tool does not propagate the differences to avoid corrupting the target authorizations. During the apply job, the GRANT statements from the source are ignored and the GRANT statements from the target are read.
- LABEL ON

**LABEL ON restriction:** The LABEL ON statement is not supported for auxiliary tables.

- RENAME INDEX

The DDL of the index must be included in the source DDL along with the RENAME INDEX statement.

**RENAME INDEX restriction:** Rename of an implicit index is not supported.

- RENAME TABLE
- SET CURRENT PATH
- SET CURRENT SQLID

## Batch DDL file extraction program report format

The report that the batch DDL file extraction program produces begins with a header and the IBM copyright statement. The copyright statement is followed by a line that indicates the version of Db2 startup parameters that are used when the extraction program is processing statements from the input stream.

Next, if the first statement in the input stream is not a SET CURRENT SQLID statement, the program indicates the authorization ID under which the input statements are being processed. This authorization ID serves as the owner of objects that are created and as the default schema name when a schema name is required but not specified. The authorization ID remains in effect until it is changed by a subsequent SET CURRENT SQLID statement.

Finally, a statistical summary of the process is produced that indicates the number of:

- DDL input records in the input stream
- Unique DDL statements within those records
- Catalog records written to an intermediate data set
- Catalog records written to the final output data set

The following figure shows sample output.

```

>-----
GOC2DTC - Create Version File from DDL File                                2006-06-09 18:57
-----

      DB2 Object Comparison Tool
      5697-L40 (C) Copyright IBM Corporation 2001, 2007.
      All rights reserved. Licensed materials - property of IBM.
      US Government Users Restricted Rights - Use, duplication or disclosure
      restricted by GSA ADP schedule contract with IBM Corp.

Using DB2 DECP Version 8(new function mode) startup parameters for SSID DSN8
Processing under auth_id of current task, VNDR230, until changed by SET CURRENT SQLID statement.

-----
GOC2DTC - Create Version File from DDL File                                2006-06-09 18:57
-----

GOC2DTC - Summary
      Number of DDL input records           :    369
      Number of DDL statements              :     41
      Number of Catalog records intermediate :     59
      Number of Catalog records written     :     59

GOC2DTC - Successful completion

```

Figure 62. CREATE VERSION report from DDL file

---

## Chapter 14. Batch Db2 catalog extraction program

The batch Db2 catalog extraction report is produced when you set target DDL file definitions on the Specify Compare Target panel and run a compare job.

This report is generated by using the Db2 Administration Tool ADB2GEN program.

The use of ADB2GEN in the compare process is controlled by two program parameters, which are set in the JCL:

- WRTCAT (write catalog records, that is create a version file).

**Restriction:** Version files are compressed internally and should not be created with DFSMS compression because GEN and the DDL reader opens them for update, which is not allowed for DFSMS compressed data sets. Db2 Administration Tool or Db2 Object Comparison Tool jobs will receive S213-C8 abends if the version file data sets are defined with DFSMS compression.

**Restriction:** If LOB objects are involved, a new version file layout is created. This new version file is not compatible with old version files containing LOBs. The old version files with LOBs must be regenerated.

- NOGEN (do not create DDL for extracted objects)

You specify options and object extract requests in exactly the same manner as in ADB2GEN. However, when you extract objects for Db2 Object Comparison Tool, you generate all parameter and request input by using the ISPF panels.

### Related tasks

[“2. Specifying target objects” on page 68](#)

After you specify the comparison source, the next step is to specify the target. The *target* is the object or objects that you want to compare to the source.

### Related reference

[“Batch Db2 catalog extraction program report” on page 175](#)

The report that the batch Db2 catalog extraction program produces begins with a header and the IBM copyright statement. The copyright statement is followed by a line that indicates the version of Db2 startup parameters that are used when the extraction program is processing statements from the input stream.

---

## Batch Db2 catalog extraction program report

The report that the batch Db2 catalog extraction program produces begins with a header and the IBM copyright statement. The copyright statement is followed by a line that indicates the version of Db2 startup parameters that are used when the extraction program is processing statements from the input stream.

The report contains four parts:

- Header and IBM copyright statement
- Db2 system ID and version, followed by a summary of the parameters
- A summary of object extract requests and related messages
- A count of the number of catalog records written

The following figure shows sample output.

```

>-----
ADB2GEN - Create DDL from catalog info                                2006-06-09 18:57
-----
Database 2 Administration Tool
5697-L90 (C) Copyright IBM Corporation 1998, 2006.
All rights reserved. Licensed materials - property of IBM.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP schedule contract with IBM Corp.
-----
ADB2GEN - Create DDL from catalog info                                2006-06-09 18:57
-----
Input prepared by Sqliid VNDR230 on DSN8 (DB2 version 810) for use on DB2 version 810 system
Object definitions extracted from DSN8 (DB2 version 810)

Parameters for this run :

Create Database(s)      : No   Create Tablespace(s)   : Yes   Create Table(s)       : Yes
Create View(s)         : Yes   Create Index(es)       : Yes   Create Synonym(s)     : Yes
Create Alias(es)       : Yes   Create Label(s)        : Yes   Create Comment(s)     : Yes
Create Triggers        : Yes   Create Foreign key(s)  : Yes   also for refs not gen'd : Yes
Create User def. Types : No   Create Functions       : No   Create Stored Procedures: No
Create Sequences       : No

Copy Stogroup Grant(s) : Yes
Copy Database Grant(s) : Yes   Copy Tablespace Grant(s): Yes   Copy Table Grant(s)      : Yes
Copy View Grant(s)     : Yes   Copy authorisations on referenced schema(s) : No
Copy U.def type Grant(s): No   Copy Function Grant(s)  : No   Copy Procedure Grant(s) : No
Copy Sequence Grant(s) : No

Insert COMMIT statement after every definition
RE will generate all parameters even if they take default values
-----
ADB2GEN - Create DDL from catalog info  TABLESPACE TTT8S81D FROM CAT  2006-06-09 18:57
-----

Generating DDL for Tablespace DSN8S81D In Database TTT8D81A

-----
ADB2GEN - Create DDL from catalog info  TABLESPACE TTT8S81D FROM CAT  2006-06-09 18:57
-----

ADB2GEN - Summary of catalog records written

Number of catalog records written      :    118

ADB2GEN - Ended normally

```

Figure 63. CREATE VERSION report from Db2 catalog

ADB2GEN gets the DECIMAL=COMMA/PERIOD (and other Db2 parameters) from a DSNHDECP module which ADB2GEN looks for in the STEPLIB data sets. The values that ADB2GEN finds in this module might not match what Db2 is currently using, or match the values that were used to store data in catalog rows; if the modules don't match, ADB2GEN might produce incorrect DDL.

You can determine the DSNHDECP parameters that ADB2GEN is using by referring to the DSNHDECP parameter section of the ADB2GEN output listing. An example is highlighted in the following figure.

```

>-----
ADB2GEN - Create DDL from catalog info                                2006-11-29 13:50
-----

Database 2 Administration Tool
5697-L90 (C) Copyright IBM Corporation 1998, 2006.
All rights reserved. Licensed materials - property of IBM.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP schedule contract with IBM Corp.

-----
ADB2GEN - Create DDL from catalog info                                2006-11-29 13:50
-----

Input prepared by Sqliid SINNOTT on DB8A (DB2 version 810) for use on DB2 version 810 system
Object definitions extracted from DB8A (DB2 version 810)

DB2 DSNHDECP values for this run :          DB2 Version, Release and Mod Level : 810
Default CCSID for EBCDIC SBCS : 00037      Decimal point option : '.'      Default
CCSID for EBCDIC Mixed : 00002            Subsystem ID : DB8A      Default CCSID
for EBCDIC DBCS : 00002                  Graphic for DBCS data : No      Default CCSID
for ASCII SBCS : 00437                  Date format : ISO      Default CCSID for
ASCII Mixed : 00002                    Time format : ISO      Default CCSID for ASCII
DBCS : 00002                            Default encoding scheme : EBCDIC Default CCSID for UNICODE SBCS :
00367          DB2 Version 8 New Function Mode : Yes Default CCSID for UNICODE Mixed :
01208          Default CCSID for UNICODE DBCS : 01200

Parameters for this run :

Create Database(s)      : Yes   Create Tablespace(s)  : Yes   Create Table(s)       : Yes
Create View(s)         : Yes   Create Index(es)      : No    Create Synonym(s)     : No
Create Alias(es)      : No    Create Label(s)       : No    Create Comment(s)     : No
Create Triggers        : No    Create Foreign key(s) : No    also for refs not gen'd : No
Create User def. Types : No   Create Functions      : No    Create Stored Procedures: No
Create Sequences       : No

Copy Stogroup Grant(s) : Yes
Copy Database Grant(s) : Yes   Copy Tablespace Grant(s): Yes   Copy Table Grant(s)    : No
Copy View Grant(s)     : No    Copy authorisations on referenced schema(s) : No
Copy U.def type Grant(s): No   Copy Function Grant(s) : No    Copy Procedure Grant(s) : No
Copy Sequence Grant(s) : No

Insert COMMIT statement after every definition
RE will generate all parameters even if they take default values
-----
ADB2GEN - Create DDL from catalog info                                2006-11-29 13:50
-----

```

Figure 64. DSNHDECP values



---

## Chapter 15. Batch compare program

The batch compare program is run when you specify **Generate online** = NO on the **Generate Compare Jobs (GOC5)** panel. This program compares two sets of Db2 objects, reports the differences, and writes the changes to a file. This changes file is used to generate updates to upgrade target objects to match the source objects.

For more information about setting batch compare options, see [“5. Generating a compare job” on page 84](#).

The batch compare program processes two version files, one that represents the (new) source version of the objects to be compared and one that represents the (old) target version.

The batch compare program performs the following tasks:

- Applies any masks to the prefix of the source version file
- Sorts the two version files
- Compares the two version files, applying masks to all relevant names and authorization IDs before comparison and ignoring any differences that are specified in the ignore file

In addition, you can create a list of objects to be excluded from a compare process by using exclude specification. The list can be created manually or based on results from a compare results stored in a Db2 table. For more information about creating and using exclude specifications, see [Chapter 6, “Excluding objects from the compare process,” on page 141](#).

For batch compare report examples, see [“Compare report format ” on page 124](#).

### Related concepts

[“Components of the comparison process ” on page 17](#)

Db2 Object Comparison Tool compares objects by reading the Db2 catalog or DDL files. Object Comparison Tool produces comparison reports and then optionally generates either JCL jobs or work statement list (WSL) tasks with changes for the target objects.

[“Batch DDL file extraction program ” on page 171](#)

The DDL file extraction program interprets a source file of DDL statements that define Db2 objects. The program generates a *version file*, which contains records that are similar in format to those in the Db2 catalog that defines the same objects.

### Related tasks

[“5. Generating a compare job” on page 84](#)

A *compare job* performs the requested comparison.

[“Specifying a version scope for the source or target definition” on page 67](#)

### Related reference

[“Compare report format ” on page 124](#)

The information that is contained in the compare report can vary based on the options selected on the **Specify Compare Reporting Options (GOC5RO)** panel.

---

## Compare version files

Db2 Object Comparison Tool operates on sorted version files. The version file record prefix is the sort key. Masks are applied to the prefix of the source version file before the file is sorted. The result is that objects in the source and target version files are in the same sequence.

The following results can occur when the batch compare program attempts to match object names:

- An object was not found in the source version file.

In this case, the object is registered for deletion unless the option to keep target objects was specified through the Suppress DROP of objects field in the **Generate Compare Jobs (GOC5)** panel.

- An object was not found in the target version file.

In this case, the object definition is saved to create the object at a later stage. Masks are applied to the relevant fields before the object definition DDL is built.

- An object was found in the source and target version files.

In this case, masks are applied to the source version Db2catalog records that describe the source object.

The objects are compared; only fields for which ignore has not been specified (explicitly or by default) are included. The possible results of the comparison are:

- Objects are identical
- Upgrade can be performed by altering the target object
- Upgrade requires drop and re-create of the target object

The differences are reported, and the actions that are required to upgrade the target version to the source version (if any) are written to the CHANGES file for use at a later stage.

## Special considerations for comparing Db2objects

---

You can perform most comparisons field by field, comparing the catalog records that represent the objects. However, special considerations are needed in some situations.

These situations are described in the following sections:

- [“Constraint names” on page 180](#)
- [“DROP statements in the source DDL” on page 181](#)
- [“Functions” on page 181](#)
- [“Implicit and explicit objects” on page 181](#)
- [“Materialized query tables” on page 181](#)
- [“Native SQL procedures” on page 182](#)
- [“Object authorizations” on page 182](#)
- [“Online schema evolution” on page 182](#)
- [“Partitioned tables” on page 182](#)
- [“Renamed objects” on page 183](#)
- [“Special considerations for comparing Db2objects” on page 180](#)
- [“Table columns” on page 183](#)
- [“Table drop/re-create without data conversion” on page 184](#)
- [Table 9 on page 185](#)
- [“Triggers” on page 185](#)
- [“Views” on page 185](#)

### Constraint names

Constraint names are not compared (and differences not reported) because constraint names can be explicitly specified or, if they are not explicitly specified, be generated by Db2. If the constraint names are generated by Db2, the constraint names could be different between source and target, even if the DDL for the object might be the same for source and target.

Differences in constraint name are not compared, because this comparison would cause an unnecessary drop and re-create of constraints that are logically correct. Dropping and recreating constraints would put the table space in CHECK PENDING, that is, out of service. Because the objects have no real differences, only differences in the constraint names, putting the table space out of service might not be necessary.

## DROP statements in the source DDL

All DROP statements in the source DDL are copied to the DDL that is produced during the compare process. The effect of the drop statements is the same as dropping the objects on the target before running the compare job. Data from the dropped tables is saved by generated unload utilities making it possible for you to recover data from the dropped tables manually. In addition, the corresponding RUNSTATS, IMAGECOPY, and CHECK DATA utilities are not generated even if they are requested on the **Generate Compare Jobs (GOC5)** panel.

All implicitly dropped objects are found when the target catalog is available. However, if the target catalog is specified in the DDL, the DROP impact might be incorrectly reported. Data in the dropped objects that is missing from the DROP impact report is not saved by generated unload utilities. It is important that you save the dropped objects if at least one DROP statement is in the source DDL and the target catalog is unavailable. DROP statements in the target DDL are ignored. The statement sequence CREATE/DROP for the same object is invalid, the result is unpredictable.

## Functions

Functions are compared based on the function signature, meaning that the function-specific name is treated as an attribute of the function, and a comparison is performed. If specific names are different, the target function definition is upgraded with the source-specific name. If you do not want the function definition upgraded, SYSROUTINES.SPECIFICNAME should be ignored.

If SQL PL functions, including non-inline SQL scalar function and SQL table function, are included in the compared objects, use the compare option **Bypass SQL PL functions** to control how Db2 Object Comparison Tool should process the objects. When the **Bypass SQL PL functions** option is specified as NO and when the non-inline SQL scalar functions or the SQL table functions are detected, Db2 Object Comparison Tool terminates processing. Otherwise, Db2 Object Comparison Tool skips the non-inline SQL scalar functions and the SQL table functions. Db2 Object Comparison Tool then continues processing the other objects and generates the APPLY job or work statement list.

**Note:** Examine the APPLY job or work statement list to verify that the content is complete.

## Implicit and explicit objects

Db2 Object Comparison Tool compares implicit objects from the source with implicit objects from the target and explicit objects from source with explicit objects from target. All objects from DDL source are explicit objects. If no explicit counterparts are found on the target, those objects are processed as new explicit objects which need to be added to the target. If no explicit counterparts for explicit target objects are found on the source, those objects are dropped from target.

## Materialized query tables

Comparisons involving materialized query tables (MQTs) do not compare columns. Instead, only the table type is compared.

For example:

1. If the target is defined as:

```
CREATE TABLE <schema>.<mqt_name> AS (  
  SELECT * FROM SYSIBM.SYSDUMMY1 )  
DATA INITIALLY DEFERRED REFRESH DEFERRED IN <dbname>.<ts_name>;
```

2. And the source is a same-named, different columned table (it does not matter if the source had 20 more columns):

```
CREATE TABLE <schema>.<table_name> (AAAAAAD CHAR(2))  
IN <dbname>.<ts_name>;
```

3. The compare output shows:

```
SDSF OUTPUT DISPLAY XXXXXXXX Jnnnnnnn DSID 110 LINE 49 COLUMNS 02- 81
COMMAND INPUT ===> SCROLL ===> CSR
Compare table source(<schema>.<mqt_name>) and target(<schema>.<tb_name>)
(A)Table type changed from Materialized Query Table to normal table
Table will be altered

GOC2CMP - Ended normally
```

4. And the result is:

```
--#ADMIN PROCESS CREATE
ALTER TABLE <schema>.<mqt_name>
DROP MATERIALIZED QUERY;
COMMIT;
```

## Native SQL procedures

Db2 Object Comparison Tool compares active and inactive versions of a native SQL procedure by comparing the options and the native SQL procedure bodies. The native SQL procedure bodies are compared the same way as the trigger bodies are compared.

## Object authorizations

Db2 Object Comparison Tool handles object authorizations differently, depending on the object location:

- For objects that exist on both the source and the target, Db2 Object Comparison Tool compares and reports the authorization differences, but does not propagate the differences from the source to the target. Db2 Object Comparison Tool does not propagate the differences to avoid corrupting the target authorizations. During the apply job, the GRANT statements from the source are ignored and the GRANT statements from the target are read.
- For objects that exist only on the source and that are added to the target during the apply job, the source authorization is applied to the target objects.

## Online schema evolution

The following DB2 8 online schema evolution functionality is not exploited by Db2 Object Comparison Tool. This means that none of the related ALTER statements are generated when applying these types of changes:

- Alter of Identity column attributes.
- Add partitioning key. This function is intended for adding partitioning information for a table in a partitioned table space if the definition of the table is incomplete.

## Partitioned tables

Tables in partitioned table spaces can be dropped only by dropping the table space. If a table in a partitioned table space has changes that require the table to be dropped and re-created, the partitioned table space is dropped and re-created as well, even if the table space comparison shows no differences.

Db2 Object Comparison Tool can accept differences in the number of partitions by ignoring the field SYSTABLESPACE.PARTITIONS. In this case, no comparisons are performed at a partition level, and all partition characteristics are taken from the target.

If the table space is not part of the comparison (that is, the comparison is performed at the table level), the following conditions apply when a partitioned table needs to be dropped and re-created:

- If the target is a DDL file, the table space cannot be dropped and re-created because the table space definition is not available.
- If the target contains tables from the Db2 catalog, the table space definition from the catalog is stored in the version file. Unless otherwise indicated, the fact that a table is partitioned is derived from the stored table space definition. In any other case, the table space definition is used only for the purpose of re-creating the table space.

## Pending Changes

Pending changes are included in version files created from catalog records. No pending changes can be included in version files from DDL. Pending changes are merged into the changed objects before the objects are compared. If pending changes are ignored, the source and target pending changes are not merged into the changed objects. If the source DDL contains an ALTER statement with DROP PENDING CHANGES, the ALTER statement is passed to the target, and pending changes in the target version file are ignored.

## Renamed objects

When comparing objects, Db2 Object Comparison Tool considers whether a database, table space, table, index, or column was renamed in the source system. You can tell Db2 Object Comparison Tool when a rename has occurred in the source system by using rename specifications. Enter rename specifications the same way that you enter compare masks. For more information about entering rename specifications, see “3. Specifying compare masks” on page 71. When you specify that an object or column was renamed in the source, Db2 Object Comparison Tool compares the existing object in the target with the renamed object in the source. When the target object is updated, the data in the target system is preserved. For example, you have the following source and target objects:

```
Source = CREATE TABLE USERA.T2 (COLA, COLB, COLY, COLZ)
Target = CREATE TABLE USERA.T2 (COLA, COLB, COLY, COLZ)
```

If you rename the source table T2 to T1 and COLY to COLX, the source and target objects are now different.

```
Source = CREATE TABLE USERA.T1 (COLA, COLB, COLX, COLZ)
Target = CREATE TABLE USERA.T2 (COLA, COLB, COLY, COLZ)
```

Using the following RENAME specifications (see “3. Specifying compare masks” on page 71 for syntax examples and supported object types), the table is renamed during the compare process to T1 and COLY is renamed to COLX:

```
RENAMETB:USERA.T2,USERA.T1
RENAMECOL:USERA.T1.COLY,COLX
```

**Note:** The new table name (T1) is referenced in the RENAMECOL statement because the RENAMETB statement occurs before the RENAMECOL statement. If the RENAMECOL statement was issued first, you would reference the original table name in the RENAMECOL statement.

The following steps are generated on the target system:

- Unload the table T2 data
- Drop table T2 and create table T1
- Load the COLY data from table T2 data into COLX in table T1

### Restrictions:

- It is not always possible for Db2 Object Comparison Tool to uniquely relate a column to a specific table because there is no connection to Db2 at the time the compare process is run (the object definitions also might originate from DDL). This situation occurs when a view references two tables and has an unqualified reference to a column. Db2 Object Comparison Tool checks if a rename might be the reason for the difference and indicates this reason in the report. If differences exist, the final outcome is not affected and the view is changed accordingly.
- Renaming an implicit index is not supported.
- Renaming an auxiliary table is not supported.

## Table columns

Table columns are matched based on column name. If column positions are different, the table is dropped and re-created to reflect the source sequence of columns.

Column names that are not found in the source file are considered dropped and are removed from the target table unless suppress drop of columns is specified in the **Generate Compare Jobs (GOC5)** panel.

Column names that are not found in the target file are considered new and are added to the target table. If the source and target tables are identical except for one or more appended columns, the target table is altered to add the new columns if the column attributes are acceptable. Otherwise, the table is dropped and re-created.

## Table drop/re-create without data conversion

Under certain conditions Db2 Object Comparison Tool can determine that the step that occurs between the unload and load steps to convert the data is not necessary. Performance can improve when the conversion step is omitted from the batch job.

In general, Db2 Object Comparison Tool does not generate a conversion step when the following table modifications are made:

- The table is renamed.
- Columns in the table are:
  - Moved
  - Renamed
  - Deleted
  - Inserted with an attribute of WITH DEFAULT or NULLS
- Only the attributes of the column are changed.

The data types and lengths are changed according to the matrix in the following table:

*Table 8. Matrix for data type and length changes that do not require data conversion. The following table describes the matrix used by Db2 Object Comparison Tool to determine whether the data conversion step between unload and load can be skipped.*

	From data type								To data type							
	SMALL INT	INT	DEC	FLOAT	CHAR	VAR CHAR	LVAR CHAR	DATE	TIME	TIME STAMP	RID	BIG INT	DEC FLOAT (16)	DEC FLOAT (34)	BIN ARY	VAR BIN ARY
SMALLINT	Y	Y	A	Y	-	-	-	-	-	-	-	Y	Y	Y	-	-
INT	-	Y	A	Y	-	-	-	-	-	-	-	Y	Y	Y	-	-
DEC	A	A	A	Y	-	-	-	-	-	-	-	Y	Y	Y	-	-
FLOAT(1-21)	-	-	-	Y	-	-	-	-	-	-	-	Y	Y	Y	-	-
FLOAT(22-53)	-	-	-	Y	-	-	-	-	-	-	-	Y	Y	Y	-	-
CHAR	A	A	A	-	Y	Y	Y	-	-	-	-	-	-	-	Y	-
VARCHAR	A	A	A	-	Y	Y	Y	-	-	-	-	-	-	-	-	Y
LVARCHAR	-	-	-	-	Y	Y	Y	-	-	-	-	-	-	-	-	-
DATE	-	-	-	-	-	-	-	Y	-	-	-	-	-	-	-	-
TIME	-	-	-	-	-	-	-	-	Y	-	-	-	-	-	-	-
TIMESTAMP	-	-	-	-	-	-	-	Y	Y	Y	-	-	-	-	-	-
RID	-	-	-	-	-	-	-	-	-	-	Y	-	-	-	-	-
BIGINT	Y	Y	Y	Y	-	-	-	-	-	-	-	Y	-	-	-	-
DECFLOAT(16)	Y	Y	Y	Y	-	-	-	-	-	-	-	-	Y	Y	-	-
DECFLOAT(34)	Y	Y	Y	Y	-	-	-	-	-	-	-	-	Y	Y	-	-
BINARY	-	-	-	-	Y	-	-	-	-	-	-	-	-	-	Y	-
VARBINARY	-	-	-	-	-	Y	-	-	-	-	-	-	-	-	-	Y

**Notes:**

- **Y = YES**, data conversion is always skipped.
- **A = ACTION**, data conversion is normally performed, with truncation, if necessary. If the new column can accommodate the data, data conversion is skipped for the following conversion types:
  - smallint to decimal
  - integer to decimal
  - decimal to smallint
  - decimal to integer
  - decimal to decimal

However, if the scale of the decimal type is changed, the data conversion is performed.

When the data conversion step is skipped, a converted unload data set is not created.

## Triggers

Triggers are represented as character strings that contain CREATE TRIGGER statements. To apply masks to the trigger definition, triggers are parsed and the language elements are identified. Masks are applied to the source trigger elements where masks are applicable, and the triggers are compared element by element.

The only exception to this process is that to successfully compare an unqualified name to a qualified name, the compare program attempts to determine implicit qualifiers for unqualified names. If the trigger has changed, the change is reported.

The sequence in which triggers are created is important because they are run in the same sequence by Db2. To maintain the correct sequence, all triggers for a table are processed at the same time.

The manner in which the batch compare program processes triggers depends on the value that you entered in the Suppress DROP of target field in the **Generate Compare Jobs (GOC5)** panel. The following table provides more information.

*Table 9. Trigger comparison process.* The following table describes how different types of triggers are handled when the "Suppress DROP of target" field is set to No or Yes.

<b>Suppress DROP of target objects No</b>	<b>Suppress DROP of target objects Yes</b>
Source file sequence and contents are used.	Source file sequence and contents are used for all triggers in the source file.
Triggers are compared, one by one, based on the trigger name.	Triggers are compared, one by one, based on the trigger name.
If a trigger is not in the target file or if the compare finds a difference, the trigger is added or dropped and re-created. All subsequent triggers are dropped and, if applicable, re-created to maintain the correct sequence.	If a trigger is not in the target file or if the compare finds a difference, the trigger is added or dropped and re-created. All subsequent triggers are dropped and, if applicable, re-created to maintain the correct sequence.
Only triggers found in the target file are dropped.	To avoid violating the sequence of triggers in the source file, only triggers that are found in the target file are included in the first possible position. This approach maintains the original position of these triggers in the target file.

## Views

Views are represented as character strings that contain CREATE VIEW statements. To apply masks to the view definition, views are parsed and the language elements are identified. Masks are applied to the source view elements where masks are applicable, and the views are compared element by element.

The only exception to this process is that to successfully compare an unqualified name to a qualified name, the compare program attempts to determine implicit qualifiers for unqualified names. If the view changes, the change is reported and the view definition, changed or not, is stored.

When the batch compare program has processed all views, it analyzes two types of dependencies:

### View dropped

A view is dropped if one of the base tables or views that is referred to was dropped. If a view is dropped, it is re-created regardless of whether it was changed.

### View dependent on another view

The sequence in which views are created is important because a view can refer to another view. The stored view definitions are sequenced to take this into account.

This behavior means that CREATE VIEW statements are not necessarily listed in the sequence in which they were processed.

## Comparisons with LOBs

---

When comparisons involve LOBs, changes can be generated. However, some restrictions exist.

Objects with LOB columns can be unloaded, with base table data written to the SYSREC data set and the LOB column data written to the data set identified by the LOB template. This capability requires that the apply job be built as a work statement list (WSL). For details on using the utility template to unload data from LOBs and to run a work statement list, see [Unloading data from LOBs by using the utility template \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#).

### Condition codes

When you run batch compare reports with LOBs, the following condition codes are issued by the GOC2CMP program:

- 0**  
Ended normally.
- 4**  
Warning issued. Review output.
- >4**  
Error found. Review output.

### LOB restrictions

Db2 Object Comparison Tool has limited LOB column changes. The only possible conversions are:

```
CHAR, VARCHAR -> CLOB, BLOB  
GRAPHIC, VARGRAPHIC -> DBCLOB
```

The reverse sequence of a LOB column to a non-LOB column is not supported (for example CLOB to CHAR). The comparison report includes one of the following messages:

- (E ) This type change is not supported.
- (W) This type change is not supported.

LOB column length reduction is not supported and can cause a failure during the comparison. The comparison report includes one of the following messages:

- (E) LOB column length is reduced. This is not supported by Db2 Object Comparison Tool. Manual action is required if you want to reduce the length of a LOB column.
- (W) LOB column length is reduced. This is not supported by Db2 Object Comparison Tool.
- (W) LOB column length will be reduced when recovering the change. Manual action will be required to recover data for this table.

Older version files containing tables with LOBs cannot be processed. The following message is returned:

- (E) The version files are generated by a previous version of the product. The version file must be re-created because the internal representation of auxiliary tables in version files has changed.

If the base table containing the LOB column or columns is dropped and re-created, the explicit auxiliary table is also re-created according to its source definition. Changes to the auxiliary table are not reported. Updates to the auxiliary table are ignored if the base table is not re-created.

# Chapter 16. Running Compare by using a Change Management batch job

You can use the Db2 Admin Tool Change Management (CM) batch interface to run Db2 Object Comparison Tool in batch. By using this interface, you can define or propagate a change that can be managed by Db2 Admin Tool Change Management.

## Procedure

To run Compare by using CM batch:

- Create and run a CM batch job with the following parameters and specifications:

### Required parameters:

CM batch parameter	Purpose	Link to parameter information
<code>ACTION_COMPARE='Y'</code>	Indicates that you want to run Compare.	<a href="#">ACTION_COMPARE (IBM Db2 Administration Tool for z/OS 13.1.0)</a>
<code>SOURCE_TYPE</code>	Specifies the compare source. You must specify any other parameters that are required based on the type value that you specify.	<a href="#">SOURCE_TYPE (IBM Db2 Administration Tool for z/OS 13.1.0)</a>
<code>TARGET_TYPE</code>	Specifies the compare target. You must specify any other parameters that are required based on the type value you specify.	<a href="#">TARGET_TYPE (IBM Db2 Administration Tool for z/OS 13.1.0)</a>

### Optional parameters:

CM batch parameter <sup>1</sup>	Purpose	Link to parameter information
<code>ACTION_IMPORT_CHANGE='N'</code>	Prevents the generated delta change file from being imported as a new registered change and analyzed on the local system. (This behavior is the default.)	<a href="#">ACTION_IMPORT_CHANGE (IBM Db2 Administration Tool for z/OS 13.1.0)</a>
Any other relevant CM batch parameter	See the description for each parameter.	<a href="#">CM batch parameter definitions (IBM Db2 Administration Tool for z/OS 13.1.0)</a>

**Other optional specifications:**

Item to specify	Method	Link to parameter information
Compare masks	<p>Specify any compare masks by taking one of the following actions:</p> <ul style="list-style-type: none"> <li>– Use the <b>COMPARE_MASK_DSN</b> parameter to specify the name of an existing data set that contains the compare masks.</li> <li>– Use the <b>COMPARE_MASK_OWNER</b> and <b>COMPARE_MASK_NAME</b> parameters to specify an existing mask specification that is in the Change Management database.</li> <li>– Pre-allocate the compare masks file with DD name of MASKS.</li> </ul>	<p><a href="#">COMPARE_MASK_DSN (IBM Db2 Administration Tool for z/OS 13.1.0)</a>  <a href="#">COMPARE_MASK_OWNER (IBM Db2 Administration Tool for z/OS 13.1.0)</a>  <a href="#">COMPARE_MASK_NAME (IBM Db2 Administration Tool for z/OS 13.1.0)</a></p>
Ignore fields	<p>Specify any ignore fields by taking one of the following actions:</p> <ul style="list-style-type: none"> <li>– Use the <b>COMPARE_IGNORE_FIELDS_DSN</b> parameter to specify the name of an existing data set that contains the compare ignore fields.</li> <li>– Use the <b>COMPARE_IGNORE_FIELDS_OWNER</b> and <b>COMPARE_IGNORE_FIELDS_NAME</b> parameters to specify an existing ignore fields specification that is in the Change Management database.</li> <li>– Preallocate compare ignore fields file with DD name of IGNORES.</li> </ul>	<p><a href="#">COMPARE_IGNORE_FIELDS_DSN (IBM Db2 Administration Tool for z/OS 13.1.0)</a>  <a href="#">COMPARE_IGNORE_FIELDS_OWNER (IBM Db2 Administration Tool for z/OS 13.1.0)</a>  <a href="#">COMPARE_IGNORE_FIELDS_NAME (IBM Db2 Administration Tool for z/OS 13.1.0)</a></p>
Ignore changes	<p>Specify any ignore changes by using the <b>COMPARE_IGNORE_CHANGES_OWNER</b> and <b>COMPARE_IGNORE_CHANGES_NAME</b> parameters. These parameters identify an existing ignore changes specification that is stored in the Change Management database.</p>	<p><a href="#">COMPARE_IGNORE_CHANGES_OWNER (IBM Db2 Administration Tool for z/OS 13.1.0)</a>  <a href="#">COMPARE_IGNORE_CHANGES_NAME (IBM Db2 Administration Tool for z/OS 13.1.0)</a></p>

Item to specify	Method	Link to parameter information
Excludes	Specify an existing exclude specification that is stored in the Change Management database as follows: <ul style="list-style-type: none"> <li>– For the compare source, use the parameters <b>SOURCE_EXCLUDE_OWNER</b> and <b>SOURCE_EXCLUDE_NAME</b>.</li> <li>– For the compare target, use the parameters <b>TARGET_EXCLUDE_OWNER</b> and <b>TARGET_exclude_NAME</b>.</li> </ul>	<a href="#">SOURCE_EXCLUDE_OWNER (IBM Db2 Administration Tool for z/OS 13.1.0)</a> <a href="#">SOURCE_EXCLUDE_NAME (IBM Db2 Administration Tool for z/OS 13.1.0)</a> <a href="#">TARGET_EXCLUDE_OWNER (IBM Db2 Administration Tool for z/OS 13.1.0)</a> <a href="#">TARGET_exclude_NAME (IBM Db2 Administration Tool for z/OS 13.1.0)</a>

## Results

The compare report and a delta change file that describes the differences is generated.

### Examples of using SELECT statements to identify source and target objects

The following sample lines from CM batch jobs show examples of CM batch parameter specifications:

#### Specifying the SELECT statement in the SQL parameter

The following lines request that the specified SQL SELECT statement (in TGTIN DD) be used to select the target objects. Those objects are to be compared with the objects identified by the DDL in the specified source data set, SOURCE\_DATASET\_NAME (in SRCIN DD).

```
//GOCCM.PARMS DD *
CHANGE_NAME           = 'CB315061'
ACTION_ANALYZE_CHANGE = 'Y'
ACTION_RUN_CHANGE     = 'Y'
ACTION_COMPARE        = 'Y'
ACTION_IMPORT_CHANGE  = 'Y'
EXISTING_DATA_SET_ACTION = 'REPLACE'
TARGET_TYPE           = 'USER'
SOURCE_TYPE           = 'DDL'
SUPPRESS_DROP_OF_OBJECTS = 'YES'
PREFIX_FOR_DATA_SETS = '&TSOID'
REPORT_SUMMARY        = 'Y'
REPORT_ONLY_CHANGED_OBJECTS = 'N'
REPORT_OBJECT_TOTALS  = 'Y'
/*
//ADBMSG DD SYSOUT=*
//SRCIN DD DSN=SOURCE_DATASET_NAME,DISP=SHR
//TGTIN DD *
TYPE='SQL',SQL="
SELECT 'TS' AS TYPE,
DBNAME AS QUAL,
NAME FROM SYSIBM.SYSTABLESPACE
WHERE NAME LIKE 'TR31506%'";
/*
```

#### Specifying the name of that data set that contains the SELECT statement

The following lines request that both the target and source objects be selected by the given SQL SELECT statements. For the target, the SELECT statement is listed in the SQL parameter (in TGTIN DD). For the source, the SELECT statement is listed in another data set, which is identified by the DSN parameter in the SRCIN DD statement.

```

//GOCCM.PARMS DD *
ACTION_COMPARE           = 'Y'
ACTION_IMPORT_CHANGE     = 'N'
ADBTPE2_RESTART         = 'N'
ACTION_GENERATE_WSL      = 'Y'
ACTION_RUN_WSL           = 'Y'
REPORT_SUMMARY           = 'Y'
REPORT_OBJECT_TOTALS     = 'Y'
RUN_REORG_REBUILD       = 'A'
TARGET_TYPE              = 'USER'
SOURCE_TYPE              = 'USER'
SUPPRESS_DROP_OF_OBJECTS = 'YES'
EXISTING_DATA_SET_ACTION = 'REPLACE'
PREFIX_FOR_DATA_SETS     = '&TSOID'
PDS_FOR_WSL              = '&SSID.WSL'
WORKLIST_NAME            = 'CB315065';
/*
//GOCCM.REPORT DD SYSOUT=*
//GOCCM.ADBDIAG DD SYSOUT=*
//GOCCM.ADBMSGs DD SYSOUT=*
//GOCCM.SYSPRINT DD SYSOUT=*
//GOCCM.MASKS DD *
DBNAME:DB315063,DB315061
TSNAME:TR315063,TR315061
TBNAME:TB315063,TB315061
IXNAME:IX315063,IX315061
IXNAME:IX315064,IX315062
//GOCCM.IGNORES DD *
  SYSDATABASE: BPOOL
//TG TIN DD *
TYPE='SQL', SQL="SELECT 'DB' AS TYPE, '' AS QUAL,
NAME FROM SYSIBM.SYSDATABASE WHERE NAME='DB315061'";
//GOCCM.SRCIN DD DSN=SOURCE_DATASET_NAME,DISP=SHR
/*

```

SOURCE\_DATASET\_NAME contains:

```

TYPE='SQL',
SQL="SELECT 'DB' AS TYPE, '' AS QUAL,
NAME FROM SYSIBM.SYSDATABASE
WHERE NAME='DB315063'";
;

```

### Specifying a DD name for the data set that contains the SELECT statement

The following lines request that both the target and source objects be selected by the given SQL SELECT statements. For the source, the SELECT statement is listed in the SQL parameter (in SRCIN DD). For the target, the SELECT statement is listed in another data set, and the SQL parameter (in TG TIN DD) lists the DD name. That referenced DD statement identifies the data set that contains the SQL statement.

```

//GOCCM.PARMS DD *
ACTION_COMPARE           = 'Y'
ACTION_IMPORT_CHANGE     = 'N'
ADBTPE2_RESTART         = 'N'
ACTION_GENERATE_WSL      = 'Y'
ACTION_RUN_WSL           = 'Y'
REPORT_SUMMARY           = 'Y'
REPORT_OBJECT_TOTALS     = 'Y'
RUN_REORG_REBUILD       = 'A'
TARGET_TYPE              = 'USER'
SOURCE_TYPE              = 'USER'
SUPPRESS_DROP_OF_OBJECTS = 'YES'
EXISTING_DATA_SET_ACTION = 'REPLACE'
PREFIX_FOR_DATA_SETS     = '&TSOID'
PDS_FOR_WSL              = '&SSID.WSL'
WORKLIST_NAME            = 'CB315064';
/*
//GOCCM.REPORT DD SYSOUT=*
//GOCCM.ADBDIAG DD SYSOUT=*
//GOCCM.ADBMSGs DD SYSOUT=*
//GOCCM.SYSPRINT DD SYSOUT=*
//GOCCM.MASKS DD *
DBNAME:DB315063,DB315061
TSNAME:TR315063,TR315061
TBNAME:TB315063,TB315061
IXNAME:IX315063,IX315061
IXNAME:IX315064,IX315062

```

```

//SRCIN DD *
TYPE='SQL', SQL="
SELECT 'TS' AS TYPE,
DBNAME AS QUAL,
NAME FROM SYSIBM.SYSTABLESPACE
WHERE NAME LIKE 'TR315063'";
//TGTRG DD *
TYPE='SQL', SQL='=SQLTRG';
//SQLTRG DD DSN=TARGET_DATASET_NAME,DISP=SHR
/*

```

### Related concepts

“Translation masks” on page 73

In Object Comparison Tool, you can use translation masks to account for differences in naming conventions between source and target objects when doing a comparison. You can also use masks to overwrite values for object attributes.

### Related information

[Managing Changes by using the CM batch interface \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[CM batch parameter definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

## Comparing table pairs

You can compare regular tables with an archive-enabled and archive table pair.

The following table shows the compare results for various sources and targets. B1 and B2 are regular tables that have corresponding archive and archive-enabled tables.

<i>Table 10. Archive table compare results</i>		
Source	Target	Expected result
None	Archive-enabled table (B1) and archive table (B2)	Archive-enabled table (B1) is dropped.  Archive table (B2) will be dropped as a result of dropping the archive enabled table.
Table B1	Archive-enabled table (B1) and archive table (B2)	Table B1 is compared.  ALTER TABLE B1 DISABLE ARCHIVE is generated.
Table B2	Archive-enabled table (B1) and archive table (B2)	ALTER TABLE B1 DISABLE ARCHIVE is generated.  Table B1 is dropped.  Table B2 is compared.
Table B1 Table B2	Archive-enabled table (B1) and archive table (B2)	Table B1 and B2 are compared.  ALTER TABLE B1 DISABLE ARCHIVE is generated.
Archive-enabled table (B1) and archive table (B2)	None	Table B1 and B2 are added.  ALTER TABLE B1 ENABLE ARCHIVE USE B2 is generated.

<i>Table 10. Archive table compare results (continued)</i>		
<b>Source</b>	<b>Target</b>	<b>Expected result</b>
Archive-enabled table (B1) and archive table (B2)	Table B1	Table B1 is compared. Archive table (B2) is added. ALTER TABLE B1 ENABLE ARCHIVE USE B2 is generated.
Archive-enabled table (B1) and archive table (B2)	Table B2	Table B1 is added. Table B2 is compared. ALTER TABLE B1 ENABLE ARCHIVE USE B2 is generated.
Archive-enabled table (B1) and archive table (B2)	Table B1 Table B2 (no archive relationship between these two tables)	Table B1 is compared. Table B2 is compared. ALTER TABLE B1 ENABLE ARCHIVE USE B2 is generated.

You can also compare temporal and history table pairs. The following table shows the compare results for various sources and targets. B1 and B2 are regular tables that have corresponding temporal and history tables.

<i>Table 11. Temporal-history table compare results</i>		
<b>Source</b>	<b>Target</b>	<b>Expected result</b>
None	Temporal table (B1) and history table (B2)	Temporal table (B1) is dropped. History table (B2) will be dropped by dropping the history-enabled table.
Table B1	Temporal table (B1) and history table (B2)	Table B1 is compared. ALTER TABLE TEMPORAL TABLE DROP VERSIONING is generated.
Table B2	Temporal table (B1) and history table (B2)	ALTER TABLE B1 DROP VERSIONING is generated. Table B1 is dropped. Table B2 is compared.
Temporal table and history table (no temporal-history relationship between these two tables)	Temporal table (B1) and history table (B2)	Table B1 and B2 are compared. ALTER TABLE B1 DROP VERSIONING is generated.
Temporal table (B1) and history table (B2)	None	Table B1 and B2 are added. ALTER TABLE B1 ADD VERSIONING USE B2 is generated.

Table 11. Temporal-history table compare results (continued)

Source	Target	Expected result
Temporal table (B1) and history table (B2)	Table B1	Table B1 is compared. History table (B2) is added. ALTER TABLE B1 ADD VERSIONING USE B2 is generated.
Temporal table (B1) and history table (B2)	Table B2	Table B1 is added. Table B2 is compared. ALTER TABLE B1 ADD VERSIONING USE B2 is generated.
Temporal table (B1) and history table (B2)	Table B2 Table B2 (no temporal-history relationship between these two tables)	Table B1 is compared Table B2 is compared. ALTER TABLE B1 ADD VERSIONING USE B2 is generated.



# Chapter 17. Specifying alternate input to the generate apply job program

You can specify data sets for Db2 Object Comparison Tool to use as alternate inputs to the program that generates the apply job. In addition, you can create a template that specifies the batch parameter variables you want your data set to contain.

## About this task

The following two members in the primary input data set store primary input variables: GOCSVARS and GOCSVAR2. You can instruct the generate apply job program to use the alternate data sets by adding a DD statement to the JCL. The alternate data sets can contain variables with customized values.

## Procedure

1. Create a primary input data set.
  - a) Select option 5 on the Db2 Object Comparison Tool menu to generate the data set that is referenced by the GOCSVARS DD statement.
2. Create data sets based on the primary input data set.
  - a) Enter an I in the option field on the Db2 Object Comparison Tool menu. The I option is hidden and is not listed as an option on the menu.

The List ISPF Table Extension Variables panel is displayed, as shown in the following figure:

```
DB2 Admin ----- List ISPF Table Extension Variables ----- 09:02
Enter/verify the following:
Data Set Name  ===>
Member Name    ===>
```

Figure 65. List ISPF Table Extension Variables panel (ADB2IIT)

- b) Enter the primary data set name and the member name (GOCSVARS for Db2 Object Comparison Tool) that you want to list.
  - c) Press Enter.

The list of variables and values for the specified data set and member is displayed.
  - d) Copy the content of the member to the newly created alternate data set.

**Requirement:** The alternate input data set must exist before this step.
  - e) Edit the variables listed in the newly copied alternate data set with the alternate values that you want to use as input to the generate apply job program.

**Requirement:** The alternate input data set must have a fixed record length of 80 characters with no sequence numbers. Each logical record begins in column one. Logical records continue on subsequent lines if they exceed the line length. Each logical record must end with a semicolon (;). All variables that are listed must exist in the alternate input data set.
  - f) Save the modified variable list. While in the edit session, use the REPLACE command to save your changes.
  - g) Repeat the previous steps, this time entering GOCSVAR2 for member name.
3. Add a DD statement, ALTSHV, that refers to the two input members in the alternate input data set. You must specify the member name explicitly.

Here is an example of the amended JCL:

```
//GOCSVARS DD DISP=SHR,DSN=HLQ.PRIMARY.SHV
//ALTSHV DD DISP=SHR,DSN=HLQ.ALTERNATE.ALTPDS(GOCSVARS)
// DD DISP=SHR,DSN=HLQ.ALTERNATE.ALTPDS(GOCSVAR2)
//CHANGES DD DISP=SHR,
// DSN=HLQ.THISCHG.CHG
```

4. Delete or rename members GOCSVARS and GOCSVAR2 from the primary input data set that is referenced by the GOCSVARS DD statement.

## Alternate values for the generate apply program

If you specify an alternate data set for input to the program that generates the apply job, you can specify alternate values for the input variables.

The following table provides a list of alternate shared data variable names and their meanings. Panel names that are the source of primary input values are identified in parentheses, where applicable.

**Requirement:** The variable names for the UNLOAD and LOAD utilities marked by an asterisk (\*) in the table are required and cannot be changed. These variables must display in the alternate input data set as shown in the following example:

```
USU01=;
USU02=;
USU03=;
```

Table 12. Alternate shared variable input data		
Variable	Definition	Valid Input
AAPFLIBR	Db2 Admin Tool APF authorized library.	A data set name. For example: DMTOOL.SADBLINK
ADB081CM	DB2 8 CM	Y or N. Specify Y if Db2 is at this release level or later.
ADB081NF	DB2 8 NFM	Y or N. Specify Y if Db2 is at this release level or later.
ADB091CM	DB2 9 CM	Y or N. Specify Y Db2 is at this release level or later.
ADB091NF	DB2 9 NFM	Y or N. Specify Y if Db2 is at this release level or later.
ADB101CM	Db2 10 CM	Y or N. Specify Y if Db2 is at this release level or later.
ADB101NF	Db2 10 NFM	Y or N. Specify Y if Db2 is at this release level or later.
ADB111CM	Db2 11 CM	Y or N. Specify Y if Db2 is at this release level or later.
ADB111NF	Db2 11 NFM	Y or N. Specify Y if Db2 is at this release level or later.
ADB121NF	Db2 12	Y or N. Specify Y if Db2 is at this release level or later.
ADB25TUA	Template usage (ADB25TU)	Y or N.
ADB27ACF	Percent increase for converted data sets	An integer.
ADB2CPS	Catalog copy plan suffix.	A two-character alphanumeric value.
ADB2USM1	Modify indicator (ADB utilities)	Y or N.
ADBADATA	Flag to indicate building work statement list for recovery by using the original data.	O or E. Use O to specify Original or E to specify Existing.
ADBANID	Analyzed change identifier	An Integer. Change ID from ADBC Prerequisite table.
ADBASUSB	Use trusted context in batch	YES or NO.
ADBASUSR	Use trusted context	AS USER value.
ADBBINDE	Bind error (ADBTEP2)	MAXE, SAVE, or IGNORE.
ADDBLKS	Blocksize (ADB2UPA)	An integer.
ADBELIB	Admin exec library concatenation.	A list of data set names. For example: 'DMTOOL.SGOCEXEC' 'DMTOOL.SADBEXEC'
ADBJ1	Job card line 1 (ADB2UPA).	A job card of up to 72 characters. Any valid job card syntax line.

Table 12. Alternate shared variable input data (continued)		
ADBJ2	Job card line 2 (ADB2UPA)	A job card of up to 72 characters. Any valid job card syntax line pt2.
ADBJ3	Job card line 3 (ADB2UPA)	A job card of up to 72 characters. Any valid job card syntax line pt3.
ADBJ4	Job card line 4 (ADB2UPA)	A job card of up to 72 characters. Any valid job card syntax line pt4.
ADBJ5	Job card line 5 (ADB2UPA)	A job card of up to 72 characters. Any valid job card syntax line pt5.
ADBJCGN	Generate Job class (ADB2UPA)	Y. Use Y to specify ADBJCLS (or DB2AJCLS if not set) as the job class.
ADBJCLS	Job class	A-Z, 0-9.
ADBJPM1	Job parm line 1 (ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBJPM2	Job parm line 2 (ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBJPM3	Job parm line 3 (ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBJPM4	Job parm line 4 (ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBJTEP2	ADBTPE2 restart parm (ADB2UPA).	Y, N, or F. Any value other than N is interpreted as yes. (FORCE), or U (USER).
ADBLLIB	The Db2 Admin Tool steplib library concatenation.	The Db2 Admin Tool load library allocation. For example: 'DMTOOL.SADBLLIB'
ADBMXDSD	Maximum allocation to disk (DASD) (ADB2UPA)	A numeric value in kilobytes.
ADBMXPRI	Maximum primary allocation (ADB2UPA)	A numeric value up to 3145680.
ADBMXPRM	Maximum primary quantity, in kilobytes, for disk (DASD) allocation (ADB2UPA)	A numeric value up to 3145680.
ADBNL	New line character variable	A hex value of '0D15'x. Use the hex edit capability of the ISPF editor and vertically specify 0D15 as shown here:  000386 ADBNL= ; CCDD70154444444 14253ED5E0000000
ADBNLC	New line character variable	A hex value of '0D15'x. Use the hex edit capability of the ISPF editor and vertically specify 0D15 as shown here:  000386 ADBNLC= ; CCDDC701544444444 142533ED5E0000000
ADBPRIM	Primary space allocation (ADB2UPA)	A numeric value specified in &ADBSPEC units.
ADBRPM1	(ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBRPM2	(ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBRPM3	(ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBRPM4	(ADB2UPA)	Any valid /*JOBPARM card syntax. For example: SYSAFF=SY4A.
ADBSECU	Secondary space allocation (ADB2UPA)	A numeric value specified in &ADBSPEC units.
ADBSPEC	Space allocation unit (ADB2UPA)	BLK, TRK, CYL or 4096-32760.
ADBTAPU	Tape unit (ADB2UPA)	Unit to use if allocation memory exceeds ADBMXDSD value. Esoteric name, such as 'TAPE'.
ADBTPEAC	Auto check (ADBTPE2)	YES, Y, NO, or N.

Table 12. Alternate shared variable input data (continued)		
ADBTEPAI	Auto rebuild (ADBTEP2)	YES, Y, NO, or N.
ADBTEPAR	Auto reorg (ADBTEP2)	YES, Y, NO, or N.
ADBTEPCD	Check at Drop (ADBTEP2)	YES, Y, NO, or N.
ADBTEPIB	Advisory auto rebuild (ADBTEP2)	YES, Y, NO, or N.
ADBTEPIR	Advisory auto reorg (ADBTEP2)	YES, Y, NO, or N.
ADBTEPSP	SPANNED	YES or NO. Use YES to specify SPANNED YES for utility statements or NO to specify SPANNED NO.
ADBTEST	Use test plan	YES or any other value.
ADBTLTB	Template library name (ADB25TU)	The ISPF table name defined by ADBGAJOB if online processing or "Y" if batch processing.
ADBTSTPN	Test plan name	A name.
ADBUNIT	Unit (ADB2UPA)	An esoteric name, such as 'SYSALLDA'.
ADBWLDSN	Work list data set name (GOC5WL)	A data set name.
AHPULLIB	HPU load library	A data set name.
ALNALTR	DDL for the altered objects (ADB25TU3)	A template name. Associated with ALALTR keyword on ADB25TU3 panel.
ALNCMD	Db2 commands (ADB25TU3)	A template name. Associated with ALCMD keyword on ADB25TU3 panel.
ALNCNC	Load control cards for the altered objects (ADB25TU3)	A template name. Associated with ALCNC keyword on ADB25TU3 panel.
ALNCNT	Load control cards for the original objects (ADB25TU3)	A template name. Associated with ALCNT keyword on ADB25TU3 panel.
ALNCREA	DDL for the created objects (ADB25TU3)	A template name. Associated with ALCREA keyword on ADB25TU3 panel.
ALNDROP	DDL for the dropped objects (ADB25TU3)	A template name. Associated with ALDROP keyword on ADB25TU3 panel.
ALNMTC	Name of non-utility data set for multi-target change information	A template name. Associated with ALMTC keyword on ADB25TU3 panel.
ALNRBND	Db2 commands for the rebind of plans and packages (ADB25TU3)	A template name. Associated with ALRBND keyword on ADB25TU3 panel.
ALNREFR	DDL for the refresh of materialized query tables (MQT) (ADB25TU3)	A template name. Associated with ALREFR keyword on ADB25TU3 panel.
ALNULD	Unloaded data from the original objects (ADB25TU3)	A template name. Associated with ALULD keyword on ADB25TU3 panel.
ALNULDC	Converted unload data (ADB25TU3)	A template name. Associated with ALULDC keyword on ADB25TU3 panel.
ALTDSN	Alter control card data set name (ADB25TU)	A data set name.
ALUALTR	Use indicator for DDL for the altered objects (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUCMD	Use indicator for Db2 commands (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUCNC	Use indicator for load control cards for the altered objects (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUCNT	Use indicator for Load control cards for the original objects (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUCREA	Use indicator for DDL for the created objects (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUDROP	Use indicator for DDL for the dropped objects (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUMTC	Use indicator for non-utility multi-target template (ADB25TU) for multi-target change	/ or blank. Specify / to use, or blank to not use.

Table 12. Alternate shared variable input data (continued)		
ALURBND	Use indicator for Db2 commands for the rebind of plans and packages (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUREFR	Use indicator for DDL for the refresh of materialized query tables (MQT) (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUULD	Use indicator for Unloaded data from the original objects (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ALUULDC	Use indicator for converted unload data (ADB25TU3)	/ or blank. Specify / to use, or blank to not use.
ASYRECD	Use activate HPU Parallel Unload/ Load in the batch apply job (ADB2UCUS)	A template name. Associated with ALULD keyword on ADB2UCUS panel.
ASYREDCD	Use activate HPU Parallel Unload/ Load in the batch apply job (ADB2UCUS)	A template name. Associated with ALULD keyword on ADB2UCUS panel.
ASYSLIA	ISPF linklist library 2	A data set name.
ASYSLIB	ISPF linklist library 1	A data set name.
ASYMLIB	ISPF message library	A data set name.
ASYTLIB	ISPF table library	A data set name.
CLOBCOLN	LOBCOLDDN (ADB25TU4)	A template name.
CLOBCOLU	Use indicator for LOBCOLDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CREATDSN	Create control card data set name (ADB25TU)	A data set name.
CTNCOPY1	COPYDDN 1 (ADB25TU4)	A template name. Used as the first parameter to the COPYDDN keyword. For example: COPYDDN(mytemp1).
CTNCOPY2	COPYDDN 2 (ADB25TU4)	A template name. Used as the second parameter to the COPYDDN keyword. For example: COPYDDN(mytemp1,mytemp2).
CTNDISC1	DISCARDN (ADB25TU4)	A template name. Used as the parameter to the DISCARDN keyword. For example: DISCARDN(mytemp3).
CTNERR	ERRDDN (ADB25TU4)	A template name. Used as the parameter to the ERRDDN keyword. For example: ERRDDN(mytemp4).
CTNFCOPY	FCCOPYDDN (ADB25TU4)	A template name. Used as the parameter to the FCCOPYDDN keyword. For example: FCCOPYDDN(mytemp5).
CTNFILTR	FILTERDDN (ADB25TU4)	A template name. Used as the parameter to the FILTERDDN keyword. For example: FILTERDDN(mytemp6).
CTNMAPDD	MAPDDN (ADB25TU4)	A template name. Used as the parameter to the MAPDDN keyword. For example: MAPDDN(mytemp7).
CTNPUNCH	PUNCHDDN (ADB25TU4)	A template name. Used as the parameter to the PUNCHDDN keyword. For example: PUNCHDDN(mytemp8).
CTNRECV1	RECOVERYDDN 1 (ADB25TU4)	A template name. Used as the first parameter to the RECOVERYDDN keyword. For example: RECOVERYDDN(mytemp9).
CTNRECV2	RECOVERYDDN 2 (ADB25TU4)	A template name. Used as the second parameter to the RECOVERYDDN keyword. For example: RECOVERYDDN(mytemp9, mytempA).
CTNUNLDD	UNLDDN (ADB25TU4)	A template name. Used as the parameter to the UNLDDN keyword. For example: UNLDDN(mytempB).
CTNWORK1	WORKDDN 1 (ADB25TU4)	A template name. Used as the first parameter to the WORKDDN keyword. For example: WORKDDN(mytempC).
CTNWORK2	WORKDDN 2 (ADB25TU4)	A template name. Used as the second parameter to the WORKDDN keyword. For example: WORKDDN(mytempC, mytempD).

Table 12. Alternate shared variable input data (continued)		
CTUCOPY1	Use indicator for COPYDDN 1 (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUCOPY2	Use indicator for COPYDDN 2 (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUDISC1	Use indicator for DISCARDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUERR	Use indicator for ERRDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUFCOPY	Use indicator for FCCOPYDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUFILTR	Use indicator for FILTERDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUMAPDD	Use indicator for MAPDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUPUNCH	Use indicator for PUNCHDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTURECV1	Use indicator for RECOVERYDDN 1 (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTURECV2	Use indicator for RECOVERYDDN 2 (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUUNLDD	Use indicator for UNLDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUWORK1	Use indicator for WORKDDN 1 (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CTUWORK2	Use indicator for WORKDDN 2 (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
CXMLCOLN	XMLCOLDDN (ADB25TU4)	A template name.
CXMLCOLU	Use indicator for XMLCOLDDN (ADB25TU4)	/ or blank. Specify / to use, or blank to not use.
DB2AASW	Authorization switch	Y or N.
DB2AJCLS	Job class for Db2 utility jobs	Alphanumeric.
DB2ALOAD	Db2 system library concatenation	Specify a list of data sets. For example: DB2ALOAD='DB2A.SDSNEXIT' 'DB2.SDSNLOAD' ;
DB2APREL	Db2 release	Four characters, such as 0915 or 1015.
DB2ARLIB	Db2 run library	A data set name.
DB2ASERV	Db2 current server	SSID.
DB2AULIB	Db2 SDSNLOAD	A data set name.
DB2AUTH	Db2 authid	A User ID.
DB2SYS	Db2 system name	SSID.
DROPDSN	Drop control card data set name (ADB25TU)	A data set name.
GOCA1JOB	Generate one job (GOC5)	Y, N, or P (one per process).
GOCAJDSN	Data set for apply jobs (GOC5AJ)	A data set name.
GOCAJOBN	Member prefix (GOC5)	A name.
GOCAPCON	Content of apply job (GOC5)	A or D. Use A to specify All or D to specify DDL only.
GOCDELWL	Delete WSL member before writing	Y or any character. Use Y to specify Delete or anything else to specify do not delete.
GOCGACHK	Run CHECK DATA (GOC5)	Y or N.
GOCGAIMC	Run IMAGE COPY (GOC5)	R, A, B, or N. Use R to specify Reload, A to specify Alter, B to specify Both, or N to specify None.
GOCGARUN	Run RUNSTATS (GOC5)	R – Reload A – Alter B – Both M – Minimum N – None.
GOCGAWL	As work statement list (GOC5)	Y or N.

Table 12. Alternate shared variable input data (continued)		
GOCGREB	Run REBIND (GOC5)	Y or N.
GOCGREOR	Run REORG (GOC5)	M – Mandatory A – All relevant N – None.
GOCJCL	PDS for batch jobs (GOC5)	A data set name.
GOCMIDQL	Middle level qualifier for data sets that are created	A name.
GOCONL	Generate online (GOC5)	Y or N.
GOCPRE	Prefix for data sets (GOC5)	A data set prefix
GOCUNLT	Unload method (GOC5)	U, P, or H. Use U to specify Unload, P to specify parallel unload, or H to specify HPU.
GOCUTOP	Use utility options (GOC5)	Y or N.
GOCWLN	Work list name (GOC5)	A name.
IFFDSN	Internal version file data set name (ADB25TU)	A data set name.
LOBCOLN	Name of the LOB data set	A template name. This value is used as the parameter to the LOBDDN keyword. For example: LOBDDN( <i>mytempF</i> ).
LOBCOLU	Use indicator for LOB column template	/ or blank. Specify / to use, or blank to not use &LOBCOLN.
MAPDBNAM	MAPPINGDATABASE, a utility option for REORG table space	A database name.
MAPOWNER	Mapping table owner (ADB2USOO)	An owner or schema.
MAPTBNAM	Mapping table name (ADB2USOO)	A name.
NSTUPROC	Number of steps in DSNUPROC	An integer (1 – 20).
REBDSN	Rebind control card data set name (ADB25TU)	A data set name.
RECOVER	Recover control card data set name (ADB25TU)	A data set name.
REFDSN	Refresh control card data set name (ADB25TU)	A data set name.
RUNLIB	AHPULLIB	A data set name.
XMLCOLN	Name of XML column (ADB25TU)	A template name. This value is used as the parameter to the XMLDDN keyword. For example: XMLDDN( <i>mytempE</i> ).
XMLCOLU	Use indicator for XML column template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &XMLCOLN
IMAGE COPY Utility Options		
USC01	FULL	Y or N. Use Y to specify FULL YES or N to specify FULL NO.
USC02	CHANGE LIMIT	Y or A. Use A to specify CHANGELIMIT (ANY) or Y to specify CHANGELIMIT (&USC03).
USC03	PERCENT VALUE1	0.0 to 100.0. This value is used as the first parameter to the CHANGELIMIT keyword.
USC04	PERCENT VALUE2	0.0 to 100.0. This value is used as the second parameter to the CHANGELIMIT keyword.
USC05	REPORT ONLY	Y or N. Use Y to specify REPORTONLY or N to specify no keyword.
USC06	PARALLEL	YES or an integer value between 0 and 32767.
USC07	CHECKPAGE	Y or N. Use Y to specify CHECKPAGE or N to specify no keyword.
USC08	CONCURRENT	Y or N. Use Y to specify CONCURRENT or N to specify no keyword.
USC09	SHRLEVEL	R or C. Use R to specify SHRLEVEL REFERENCE or C to specify SHRLEVEL CHANGE.
USC10	CLONE	Y or N. Use Y to specify CLONE or N to specify no keyword.

Table 12. Alternate shared variable input data (continued)		
USC11	SCOPE	A or P. Use A to specify SCOPE ALL or P to specify SCOPE PENDING.
USC113	TAPEUNITS	A numeric value. This value is used as a parameter to TAPEUNITS. For example: TAPEUNITS 3.
USC12	FLASHCOPY	Y, N, or C. Use Y to specify FLASHCOPY YES, N to specify FLASHCOPY NO, or C to specify FLASHCOPY CONSISTENT.
USC123	SYSTEMPAGES	YES or NO.
CHECK DATA Utility Options		
USK01	SCOPE	P, X, A, R, or M. Use P to specify SCOPE PENDING, X to specify SCOPE AUXONLY, A to specify SCOPE ALL, R to specify SCOPE REFONLY, or M to specify SCOPYE XMLSCHEMAONLY.
USK02	AUXERROR	R or I. Use R to specify AUXERROR REPORT or I to specify AUXERROR INVALIDATE.
USK03	EXCEPTIONS	0-32767. This number is used as a parameter to EXCEPTIONS. For example: EXCEPTIONS 257.
USK04	SORTDEVT	A device type. This value is used as a parameter to SORTDEVT. For example: SORTDEVT <i>devtype</i> .
USK05	SORTNUM	1-255. This value is used as a parameter to SORTNUM. For example: SORTNUM 93.
USK06	SHRLEVEL	R or C. Use R to specify SHRLEVEL REFERENCE or C to specify SHRLEVEL CHANGE.
USK07	CLONE	Y or N. Use Y to specify the CLONE keyword or N to specify no keyword.
USK08	LOBERROR	R or I. Use R to specify LOBERROR REPORT or I to specify LOBERROR INVALIDATE.
USK09	XMLERROR	R or I. Use R to specify XMLERROR REPORT or I to specify XMLERROR INVALIDATE.
USK10	DELETE	YES or NO. Use YES to specify DELETE YES or NO to specify no keywords.
USK11	LOG	YES or NO. Use YES to specify LOG YES or NO to specify LOG NO.
USK12	DRAIN WAITV	1-1800. This number is used as a parameter to DRAIN_WAIT. For example: DRAIN_WAIT 97.
USK13	RETRYV	0-255. This number is used as a parameter to RETRY. For example: RETRY 98.
USK14	RETRY DELAYV	1-1800. This number is used as a parameter to RETRY_DELAY. For example: RETRY_DELAY 103.
USK15	INCLUDE XML TABLESPACES	Name or ALL. This value is used as a parameter to TABLESPACES. For example: INCLUDE XML TABLESPACES mydb.myts. Substitute your database and table space for mydb.myts. For long names, also provide the table &FRTAB. Long object names are not supported.
USK16	INCLUDE XML COLUMNS	Use the following syntax: TABLE myschema.mytable XMLCOLUMN mycolumn. This value is used as a parameter to TABLESPACES. For example: INCLUDE XML TABLESPACES mydb.myts TABLE myschema.mytable XMLCOLUMN mycolumn. For long names, also provide table &TTNAME. Long object names are not supported.
USK17	INCLUDE XMLSCHEMA	YES or NO. Use YES to specify XMLSCHEMA keyword or NO to specify no keyword.
USKN1	FOR EXCEPTION IN table name	An object name. Use this value to specify FOR EXCEPTION IN name. Should also provide table &INTABL.
USKN2	USE table name	An object name. Use this value to specify USE name. Also provide the table &USTABL.
USKS1	FOR EXCEPTION IN table schema	A schema name. This value is used with &uskn1.
USKS2	USE table schema	A schema name. This value is used with &uskn2.

Table 12. Alternate shared variable input data (continued)		
MODIFY Utility Options		
USM01	AGE	0-32767. This value is used as a parameter to AGE. For example: DELETE AGE( 27).
USM02	DATE	yyyymmdd. This date value is used as a parameter to DATE. For example: DELETE DATE(20130704).
USM033	CLONE	YES or NO. Use YES to specify CLONED YES and CLONE keywords or NO to specify no keyword.
USM04	LASTV	0-32767. Use this value as a parameter to LAST. For example: RETAIN LAST(41).
USM05	LOGLIMITV	YES or NO. Use YES to specify RETAIN LOGLIMIT or NO to specify no keyword.
USM06	GDGLIMITV LASTV	0-32767. This value is used as a parameter to LAST. For example: RETAIN GDGLIMIT LAST( 12).
USM061	GDGLIMITV	YES or NO. Use YES to specify RETAIN GDGLIMIT or NO to specify no keyword.
USM07	GDGLIMITV LOGLIMITV	YES or NO. Use YES to specify RETAIN GDGLIMIT LOGLIMIT or NO to specify no keyword.
REORG Utility Options:		
USO01	REUSE	Y or N. Use Y to specify REUSE or N to specify no keyword.
USO02	LOG	Y or N. Use Y to specify LOG YES or N to specify LOG NO.
USO03	SORTDATA	Y or N. Use Y to specify SORTDATA or N to specify no keyword.
USO04	NOSYSREC	Y or N. Use Y to specify NOSYSREC or N to specify no keyword.
USO05	SORTKEYS	Y or N. Use Y to specify SORTKEYS or N to specify no keyword.
USO06	SHRLEVEL	C, R, or N. Use C to specify SHRLEVEL CHANGE, R to specify SHRLEVEL REFERENCE, or N to specify SHRLEVEL NONE.
USO07	FASTSWITCH	Y or N. Use Y to specify FASTSWITCH YES or N to specify FASTSWITCH NO.
USO08	OFFPOSLIMIT	0-65535. This value is used as a parameter to OFFPOSLIMIT. For example: OFFPOSLIMIT 1021.
USO09	INDREFLIMIT	0-65535. This value is used as a parameter to INDREFLIMIT. For example: INDREFLIMIT 201.
USO10	KEEPDICTIONARY	Y or N. Use Y to specify KEEPDICTIONARY or N to specify no keyword.
USO11	STATISTICS	Y or N. Use Y to specify STATISTICS TABLE (ALL) or N to specify no keyword.
USO12	REPORT	Y or N. Use Y to specify REPORT YES or N to specify REPORT NO.
USO13	UPDATE	A, P, S, or N. Use A to specify UPDATE ALL, P to specify UPDATE ACCESSPATH, S to specify UPDATE SPACE, or N to specify UPDATE NONE.
USO14	HISTORY	A,P,S, or N. Use A to specify HISTORY ALL, P to specify HISTORY ACCESSPATH, S to specify HISTORY SPACE, or N to specify HISTORY NONE.
USO15	FORCEROLLUP	Y or N. Use Y to specify FORCEROLLUP YES or N to specify FORCEFOLLUP NO.
USO16	PREFORMAT	Y or N. Use Y to specify PREFORMAT or N to specify no keyword.
USO17	SORTDEVT	A device type. This value is used as a parameter to SORTDEVT. For example: SORTDEVT devt.
USO18	SORTNUM	1 - 255. This number is used as a parameter to SORTNUM. For example: SORTNUM 3.

Table 12. Alternate shared variable input data (continued)		
US019	DEADLINE	N, timestamp, or labeled duration expression. Use N to specify DEADLINE NONE together with a timestamp that is used as a parameter to DEADLINE. For example: DEADLINE 13:15:01. An example of a labeled duration expression is CURRENT_DATE +3 DAYS.
US020	DRAIN_WAIT	1-1800. This value is used as a parameter to DRAIN_WAIT.
US021	RETRY	0-255. This value is used as a parameter to RETRY. For example: RETRY 8.
US022	RETRY DELAY	1-1800. This value is used as a parameter to RETRY_DELAY. For example RETRY_DELAY 17.
US024	MAXRO	D or numeric value. Use D to specify MAXRO DEFER or numeric value to specify MAXRO &uso24.
US025	DRAIN	W or A. Use W to specify DRAIN WRITERS or A to specify DRAIN ALL.
US026	LONGLOG	C, T, or D. Use C to specify LONGLOG CONTINUE, T to specify LONGLOG TERM, or D to specify LONGLOG DRAIN.
US027	DELAY	A numeric value. This value is used as a parameter to DELAY. For example DELAY 17.
US028	TIMEOUT	A or T. Use A to specify TIMEOUT ABEND or T to specify TIMEOUT TERM.
US029	CLONE	YES or NO. Use YES to specify CLONE or NO to specify no keyword.
US030	SCOPE	A or P. Use P to specify SCOPE PENDING or A to specify no keyword.
US031	REBALANCE	Y or N. Use Y to specify REBALANCE or N to specify no keyword.
US032	REPORTONLY	Y or N. Use Y to specify REPORTONLY or N to specify no keyword.
US033	UNLOAD	C, P, O, or E. Use C to specify UNLOAD CONTINUE, P to specify UNLOAD PAUSE, O to specify UNLOAD ONLY, or E to specify UNLOAD EXTERNAL.
US034	NOPAD	Y or N. Use Y to specify NOPAD or N to specify no keyword.
US035	FROM TABLE	An object name. Also provide table &FRNAME.
US036	AUX	YES or NO. Use YES to specify AUX YES or NO to specify AUX NO.
US037	A list of partitions.	Identifies the set of partitions that are to be reorganized. For example: 1, 3, 5:8
US038	FLASHCOPY	Y, C, or N. Use Y to specify FLASHCOPY YES, C to specify FLASHCOPY CONSISTENT, or N to specify FLASHCOPY NO.
US0363	This variable is not used.	
US040	LOGRANGES	<ul style="list-style-type: none"> <li>Y - Yes, REORG uses SYSLGRNX information for the LOG phase whenever possible. This option is the default behavior.</li> <li>N - NO, REORG does not use SYSLGRNX information for the LOG phase.</li> </ul>
US041	DRAIN_ALLPARTS	<ul style="list-style-type: none"> <li>Y - YES, REORG obtains the table space level drain on the entire partitioned table space first, before draining the target data partitions and the indexes.</li> <li>N - NO, REORG drains the target data partitions serially followed by the non-partitioned secondary indexes. This option is the default behavior.</li> </ul>

Table 12. Alternate shared variable input data (continued)		
USO42	SWITCHTIME	<ul style="list-style-type: none"> <li>N - NONE, does not specify a time for the final log iteration of the LOG phase. This option is the default behavior. Specifies the time that the final log iteration of the LOG phase is to begin. This time must not have already occurred when REORG is run.</li> <li>labeled-duration-expression, SWITCHTIME labeled-duration-expression is added.</li> </ul>
USO43	NEWMAXRO	<ul style="list-style-type: none"> <li>N - NONE, specifies that when the specified SWITCHTIME is met, REORG proceeds to the last log iteration without taking log processing time into consideration. This option is the default.</li> <li>Integer, specifies the number of seconds. Valid values are 0 through 2147483647.</li> </ul>
USO44	RECLUSTER	<ul style="list-style-type: none"> <li>Y - YES</li> <li>N - NO</li> </ul>
USO45	LISTPARTS	<ul style="list-style-type: none"> <li>n - An integer representing the maximum number of data partitions to be reorganized at once. Valid values are integers 1 through 2147483647.</li> </ul>
USO47	PARALLEL	YES or an integer value between 0 and 32767.
USO50	TABLE schema	Specifies the table owner for which STATISTICS information is to be gathered.
USO51	TABLE name	Specifies the table name for which information is to be gathered. The table must belong to the specified table space. Multiple table names are not supported. To gather information for all tables in the table space, specify ALL for the table name and leaving the table owner blank.
USO52	SAMPLE	Indicates the percentage of rows to sample when collecting non-indexed column statistics. Valid values are 1 through 100. The default is 25.
USO53	COLUMN name	Specifies the columns for which column information is to be gathered. This option is valid only if a table name is specified. The utility accepts a maximum of 10 column names, but Db2 Admin Tool does not validate this number. ALL means that statistics are to be gathered for all columns in the specified table name.
USO54	COLGROUP name	Specifies that inline statistics is to collect a cardinality value on the group of named columns. Multiple column groups are not supported.
USO55	FREQVAL	<ul style="list-style-type: none"> <li>Y - YES, collect frequency statistics</li> <li>N - NO, do not collect frequency statistics</li> </ul>
USO56	COUNT	Indicates the number of frequently occurring values to be collected from the specified column group.
USO57	OCCUR	<ul style="list-style-type: none"> <li>M - MOST, collect the most frequently occurring values</li> <li>B - BOTH, collect both the most and least frequently occurring values</li> <li>L - LEAST, collect the least frequently occurring values</li> </ul>
USO58	HISTOGRAM	<ul style="list-style-type: none"> <li>Y - YES, gather histogram statistics from the specified column group</li> <li>N - NO, do not gather such statistics</li> </ul>
USO59	NUMQUANTILES for HISTOGRAM	Indicates the number of quantiles that the utility collects.
USO60	INDEX(ALL)	<ul style="list-style-type: none"> <li>Y - YES, gather information for all indexes on all tables in the table space</li> <li>N - NO, do not gather such information</li> </ul>
USO61	INDEX HISTOGRAM	<ul style="list-style-type: none"> <li>Y - YES, gather histogram statistics for all indexes on all tables in the table space</li> <li>N - NO, do not gather such statistics</li> </ul>
USO62	NUMCOLS	The number of key columns that are to be concatenated when collecting histogram statistics from the specified index.

Table 12. Alternate shared variable input data (continued)		
USO63	NUMQUANTILES for INDEX HISTOGRAM	Indicates the number of quantiles that the utility collects.
USORBALR	RBALRSN	Specifies the RBA and LRSN format in which the target object is to be left after a REORG. <ul style="list-style-type: none"> <li>N - None No conversion</li> <li>B - Basic Convert to a basic format</li> <li>E - Extended Convert to extended format</li> </ul>
RUNSTATS Utility Options		
USR03	SAMPLE	1-100. This value is used as a parameter to SAMPLE. For example SAMPLE 37.
USR06	FREQVAL COUNT	1-65535. This value is used as a parameter to FREQVAL COUNT. For example FREQVAL COUNT 49.
USR07	FREQVAL COUNT type	MOST, BEST, or LEAST. This value is used as a parameter to FREQVAL. For example: FREQVAL COUNT 50 LEAST.
USR10	PART	1-4096. This value is used as a parameter to PART. For example: PART 31.
USR11	KEYCARD	Y or N. Use Y to specify KEYCARD or N to specify no keyword.
USR12	NUMCOLS	A numeric value. This value is used as a parameter to NUMCOLS. For example: FREQVAL NUMCOLS 9 COUNT.
USR13	NUMCOLS COUNT	1 - 99999. This value is used as a parameter to COUNT. For example FREQVAL NUMCOLS 3 COUNT 7.
USR14	NUMCOLS COUNT type	MOST, LEAST, or BOTH. This value is used as a parameter to COUNT. For example: FREQVAL NUMCOLS 3 COUNT 3 BOTH.
USR15	SORTDEVT	A device type. This value is used as a parameter to SORTDEVT. For example: SORTDEVT devt.
USR16	SORTNUM	2-255. This value is used as a parameter to SORTNUM. For example SORTNUM 251.
USR17	SHRLEVEL	R or C. Use R to specify SHRLEVEL REFERENCE or C to specify SHRLEVEL CHANGE.
USR18	REPORT	Y or N. Use Y to specify REPORT YES or N to specify REPORT NO.
USR19	UPDATE	A, P, S, or N. Use A to specify UPDATE ALL, P to specify UPDATE ACCESSPATH, S to specify UPDATE, or N to specify UPDATE NONE.
USR20	HISTORY	A, P, S, or N. Use A to specify HISTORY ALL, P to specify HISTORY ACCESSPATH, S to specify HISTORY SPACE, or N to specify HISTORY NONE.
USR21	FORCEROLLUP	Y or N. Use Y to specify FORCEROLLUP YES or N to specify FORCEROLLUP NO.
USR22	NUMQUANTILES 1	1-100. This value is used as a parameter to NUMQUANTILES. For example HISTOGRAM NUMQUANTILES 8.
USR23	NUMQUANTILES 2	1-100. This value is used as a parameter to NUMQUANTILES. For example HISTOGRAM NUMCOLS 3 NUMQUANTILES 61.
USR30	PROFILE	USE or DELETE. Specify USE to specify USE PROFILE or DELETE to specify DELETE PROFILE.
USR31	FROM EXISTING INCLUDE NPI	YES or NO. Use YES to specify INCLUDE NPI or NO to specify no keyword.
USR32	TABLESAMPLE	AUTO or numeric literal between '0.01' and '100'. This value is used as a parameter to TABLESAMPLE SYSTEM. For example TABLESAMPLE SYSTEM 7.

Table 12. Alternate shared variable input data (continued)		
USR33	REPEATABLE	A numeric value. This value is used as a parameter to REPEATABLE. For example REPEATABLE 65.
USR35	SET PROFILE	SET or UPDATE. Use SET to specify SET PROFILE or UPDATE to specify UPDATE PROFILE.
USR36	FROM EXISTING STATS	YES or NO. Use YES to specify FROM EXISTING STATS or NO to specify no keyword.
USR37	HISTOGRAM NUMCOLS	A numeric value. This value is used as a parameter to HISTOGRAM NUMCOLS. For example HISTOGRAM NUMCOLS 89.
UNLOAD Utility Options		
USU01	FROMCOPY*	A data set name without quotation marks. This value is used as a parameter to FROMCOPY. For example FROMCOPY <i>my.dsn</i> .
USU02	FROMVOLUME*	CATALOG or void. This value is used as a parameter to FROMVOLUME. For example: FROMVOLUME <i>vol001</i> .
USU03	FROMCOPYDDN*	DD name. This value is used as a parameter to FROMCOPYDDN. For example: FROMCOPYDDN <i>dd001</i> .
USU04	ENCODINGScheme	E, A, or U. Use E to specify EBCDIC, A to specify ASCII, or U to specify UNICODE.
USU05	SBCS CCSID	A numeric value. This value is used as a parameter to CCSID. For example: CCSID( <i>n</i> ).
USU06	MIXED CCSID	A numeric value. This value is used as the second parameter to CCSID. For example: CCSID(1, <i>n</i> ).
USU07	DBCS CCSID	A numeric value. This value is used as the third parameter to CCSID. For example: CCSID(1, 2, <i>n</i> ).
USU08	NOSUBS	Y or N. Use Y to specify NOSUBS, or N to specify no keyword.
USU09	NOPAD	Y or N. Use Y to specify NOPAD or N to specify no keyword.
USU10	FLOAT	S or I. Use S to specify FLOAT S390 or I to specify FLOAT IEEE.
USU11	MAXERR	A numeric value. This value is used as a parameter to MAXERR. For example MAXERR 47.
USU12	SHRLEVEL	1, 2, or 3. Use 1 to specify SHRLEVEL CHANGE ISOLATION CS, 2 to specify SHRLEVEL CHANGE ISOLATION UR, or 3 to specify SHRLEVEL REFERENCE.
USU13	DELIMITED	Y or N. Use Y to specify DELIMITED or N to specify no keyword.
USU17	HEADER	O, N, or C. Use O to specify HEADER OBID, N to specify HEADER NONE, or C to specify HEADER CONST #.
USU18	CONST	A character or X'hex string'. This value is used as a parameter to CONST. For example: HEADER CONST #.
USU19	SAMPLE	A percent, where 0 < x <= 100. This value is used as a parameter to SAMPLE. For example SAMPLE 22.
USU20	LIMIT	An integer, 0 - 2147483647. This value is used as a parameter to LIMIT. For example: LIMIT 20.
USU21	SKIP LOCKED DATA	YES or NO. Use YES to specify SKIP LOCKED DATA or NO to specify no keyword.
USU22	This variable is not used.	
USU23	CLONE	YES or NO. Use YES to specify CLONE or NO to specify no keyword.
USU24	IMPLICIT TIMEZONE	+NN:NN, -NN:NN. This value is used as a parameter to IMPLICIT_TZ. For example: IMPLICIT_TZ +7.
USU25	SPANNED	YES or NO. This value is used as a parameter for SPANNED. For example: SPANNED YES.
USU27	PARALLEL	YES or an integer value between 0 and 32767.

Table 12. Alternate shared variable input data (continued)		
USURND	DECFLOAT_ROUNDMODE	ROUND_CEILING, ROUND_DOWN, ROUND_FLOOR, ROUND_HALF_DOWN, ROUND_HALF_EVEN, ROUND_HALF_UP, or ROUND_UP. This value is used as a parameter to DECFLOAT_ROUNDMODE. For example: DECFLOAT_ROUNDMODE ROUND_UP.
USUUF1	FORMAT INTERNAL	Y or N. Use Y to specify FORMAT INTERNAL or N to specify no keyword.
USULIC	LAST IC	LAST, BEFORE, or AFTER. This value is used as a parameter to FROM. Use LAST to specify FROM LAST_IC, BEFORE to specify FROM BEFORE_IC, or AFTER to specify FROM AFTER_IC.
USUICD	IC date	A date, YYYY/MM/DD. This value is used as a parameter to ICDATE. For example: FROM LAST_IC ICDATE 2013/08/04.
USUICT	IC time	Time, HH:MM:SS, used as a parameter to ICDATE. For example FROM LAST_IC ICTIME 12:04:00.
LOAD Utility Options		
UTC01	UTILITY ID	A name. This value is used for UID parameter. For example: //LOAD1 EXEC DSNUPROC,SYSTEM=DSNA,UID='PSV01'.
UTC02	DSNAME	A data set name. This value is used as the SYSREC data set name. For example: //DSNUPROC.SYSREC DD DISP=SHR,DSN= my.dsn.
UTC03	DSNAME into-table-spec	A data set name. The data set contains LOAD INTO TABLE statements.
UTC04	RESUME	YES or NO. This value is used as a parameter to RESUME. For example: RESUME YES.
UTC05	SHRLEVEL	NONE or CHANGE. This value is used as a parameter to SHRLEVEL. For example: SHRLEVEL CHANGE.
UTC06	REPLACE	YES or NO. Use YES to specify REPLACE or NO to specify no keyword.
UTC07	COPYDDN1	A name. This value is used as a parameter to COPYDDN. For example: COPYDDN( name).
UTC08	COPYDDN2	A name. This value is used as a parameter to COPYDDN. For example: COPYDDN( , name).
UTC09	RECOVERYDDN1	A name. This value is used as a parameter to RECOVERYDDN. For example: RECOVERYDDN( name).
UTC10	RECOVERYDDN2	A name. This value is used as a parameter to RECOVERYDDN. For example: RECOVERYDDN( ddn1, name).
UTC12	SAMPLE*	An integer, 1- 100. This value is used as a parameter to SAMPLE. For example: SAMPLE 48.
UTC13	INDEX ALL*	YES or NO. Use YES to specify INDEX(ALL) or NO to specify no keyword.
UTC14	REPORT*	YES or NO. Use YES to specify REPORT YES or NO to specify no keyword.
UTC15	UPDATE*	A, P, S, or N. Use A to specify UPDATE ALL, P to specify UPDATE ACCESSPATH, S to specify UPDATE SPACE, or N to specify UPDATE NONE.
UTC16	KEEPDICTIONARY	YES or NO. Use YES to specify KEEPDICTIONARY or NO to specify no keyword.
UTC17	REUSE	YES or NO. Use YES to specify REUSE or NO to specify no keyword.
UTC18	LOG	YES, NO, or NOC. Use YES to specify LOG YES, NO to specify LOG NO or NOC to specify LOG NO NOCOPYPEND.
UTC19	WORKDDN1	A name. This value is used as a parameter to WORKDDN. For example: WORKDDN( name).

Table 12. Alternate shared variable input data (continued)		
UTC20	WORKDDN2	A name. This value is used as a parameter to WORKDDN. For example: WORKDDN( , name).
UTC21	SORTKEYS	An integer, 0 - 2147483647. This value is used as a parameter to SORTKEYS. For example: SORTKEYS 39.
UTC22	ENFORCE	YES or NO. Use YES to specify ENFORCE CONSTRAINTS or NO to specify ENFORCE NO.
UTC23	SORTDEVT	A device type. This value is used as a parameter to SORTDEVT. For example: SORTDEVT SYSALLDA.
UTC24	SORTNUM	1- 255. This value is used as a parameter to SORTNUM. For example: SORTNUM 12.
UTC25	SORTWK	0, 1, 2, 3, or 4. This parameter determines how many sort work DD statements are allocated. DD statements for SORTWK01, SORTWK02, SORTWK03, and SORTWK04 may be added.
UTC26	how unloaded	U or R. Use U to specify WHEN (00001:00002 = X'&XOBID') or R to specify WHEN (00004:00005 = X'&XOBID').
UTC27	DECFLOAT ROUNDING	Use &UTCRRND instead.
UTC28	IMPLICIT_TZ	+NN:NN, -NN:NN. This value is used as a parameter to IMPLICIT_TZ. For example: IMPLICIT_TZ +08.
UTC29	FLASHCOPY	Y, N, or C. Use Y to specify FLASHCOPY YES, N to specify FLASHCOPY NO, or C to specify FLASHCOPY CONSISTENT.
UTC30	PRESORTED	YES or NO. This value is used as a parameter to PRESORTED. For example: PRESORTED YES.
UTC31	PARALLEL (Db2 11 and later)	YES or an integer value between 0 and 32767.
UTC40	Table schema*	Blank. Use Blank to not specify STATISTICS TABLE(table-name) because it is not supported in the compare process.
UTC41	Table name*	Y, ALL, or blank. Use ALL to specify STATISTICS TABLE(ALL) or blank to specify no keyword.
UTC54	DISCARDS	0 - 2147483647. This value is used as a parameter to DISCARDS. For example, DISCARDS 12.
UTCRRND	DECFLOAT ROUNDING	ROUND_CEILING, ROUND_DOWN, ROUND_FLOOR, ROUND_HALF_DOWN, ROUND_HALF_EVEN, ROUND_HALF_UP, ROUND_UP. This value is used as a parameter to DECFLOAT_ROUNDMODE. For example: DECFLOAT_ROUNDMODE ROUND_UP.
UTNCOPY1	Name of data set for copy (ADB25TU)	A template name. This value is used as the first parameter to the COPYDDN keyword. For example: COPYDDN( mytemp1).
UTNCOPY2	Name of data set for copy (ADB25TU)	A template name. This value is used as the second parameter to the COPYDDN keyword. For example: COPYDDN(mytemp1, mytemp2).
UTNDISC1	Template discard data set name	A template name. This value is used as the parameter to the DISCARDN keyword. For example: DISCARDN( mytemp3).
UTNERR	Template error data set name (ADB25TU)	A template name. This value is used as the parameter to the ERRDDN keyword. For example: ERRDDN( mytemp4).
UTNFCOPY	Name of utility data set for system FCCOPY (ADB25TU)	A template name. This value is used as the parameter to the FCCOPYDDN keyword. For example: FCCOPYDDN( mytemp5).
UTNFILTR	Name of utility data set for system filter (ADB25TU)	A template name. This value is used as the parameter to the FILTERDDN keyword. For example: FILTERDDN( mytemp6).
UTNMAPDD	Name of utility data set for system map (ADB25TU)	A template name. This value is used as the parameter to the MAPDDN keyword. For example: MAPDDN( mytemp7).
UTNPUNCH	Name of utility data set for system punch (ADB25TU)	A template name. This value is used as the parameter to the PUNCHDDN keyword. For example: PUNCHDDN( mytemp8).

Table 12. Alternate shared variable input data (continued)		
UTNRECV1	Name of recovery data set (ADB25TU)	A template name. This value is used as the first parameter to the RECOVERYDDN keyword. For example: RECOVERYDDN( <i>mytemp9</i> ).
UTNRECV2	Name of recovery data set (ADB25TU)	A template name. This value is used as the second parameter to the RECOVERYDDN keyword. For example: RECOVERYDDN( <i>mytemp9</i> , <i>mytempA</i> ).
UTNUNLDD	Name of utility data set for unload (ADB25TU)	A template name. This value is used as the parameter to the UNLDDN keyword. For example: UNLDDN( <i>mytempB</i> ).
UTNWORK1	Name of utility data set for work (ADB25TU)	A template name. This value is used as the first parameter to the WORKDDN keyword. For example: WORKDDN( <i>mytempC</i> ).
UTNWORK2	Name of utility data set for work (ADB25TU)	A template name. This value is used as the second parameter to the WORKDDN keyword. For example: WORKDDN( <i>mytempC</i> , <i>mytempD</i> ).
UTUCOPY1	Use indicator for copy template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNCOPY1
UTUCOPY2	Use indicator for copy template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNCOPY2
UTUDISC1	Use indicator for template discard name	/ or blank. Specify / to use, or blank to not use &UTNDISC1
UTUERR	Use indicator for ERROR template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNERR
UTUFCOPY	Use indicator for FCCOPY template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNFCOPY
UTUFILTR	Use indicator for filter template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNFILTR
UTUMAPDD	Use indicator for Map template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNMAPDD
UTUPUNCH	Use indicator for punch template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNPUNCH
UTURECV1	Use indicator for recovery template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNRECV1
UTURECV2	Use indicator for recovery template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNRECV2
UTUUNLDD	Use indicator for unload template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNUNLDD
UTUWORK1	Use indicator for work1 template (ADB25TU)	/ . This value must be set to "/" to use &UTNWORK1
UTUWORK2	Use indicator for work2 template (ADB25TU)	/ or blank. Specify / to use, or blank to not use &UTNWORK2
REORG INDEX utility options:		
UXO01	REUSE	Y or N. Use Y to specify REUSE or N to specify no keyword.
UXO02	SHRLEVEL	R or C. Use R to specify SHRLEVEL REFERENCE or C to specify SHRLEVEL CHANGE.
UXO03	LEAFDISTLIMIT	Valid values are 0 through 2147483647.
UXO04	REPORTONLY	Y or N. Use Y to specify REPORTONLY or N to specify no keyword.
UXO05	UNLOAD	C, P, O, or E. Use C to specify UNLOAD CONTINUE, P to specify UNLOAD PAUSE, O to specify UNLOAD ONLY, or E to specify UNLOAD EXTERNAL.
UXO06	PREFORMAT	Y or N. Use Y to specify PREFORMAT or N to specify no keyword.
UXO07	DEADLINE	N, timestamp, or labeled duration expression. Use N to specify DEADLINE NONE together with a timestamp that is used as a parameter to DEADLINE. For example: DEADLINE 13:15:01. An example of a labeled duration expression is CURRENT_DATE +3 DAYS.

Table 12. Alternate shared variable input data (continued)		
UX008	DRAIN_WAIT	1-1800. This value is used as a parameter to DRAIN_WAIT.
UX009	RETRY	0-255. This value is used as a parameter to RETRY. For example: RETRY 8.
UX010	RETRY_DELAY	1-1800. This value is used as a parameter to RETRY_DELAY. For example RETRY_DELAY 17.
UX011	FASTSWITCH	Y or N. Use Y to specify FASTSWITCH YES or N to specify FASTSWITCH NO.
UX012	MAXRO	D or numeric value. Use D to specify MAXRO DEFER or numeric value to specify MAXRO &uso24.
UX013	DRAIN	W or A. Use W to specify DRAIN WRITERS or A to specify DRAIN ALL.
UX014	LONGLOG	C, T, or D. Use C to specify LONGLOG CONTINUE, T to specify LONGLOG TERM, or D to specify LONGLOG DRAIN.
UX015	DELAY	A numeric value. This value is used as a parameter to DELAY. For example DELAY 17.
UX016	TIMEOUT	A or T. Use A to specify TIMEOUT ABEND or T to specify TIMEOUT TERM.
UX017	STATISTICS	Y or N. Use Y to specify STATISTICS TABLE (ALL) or N to specify no keyword.
UX018	REPORT	Y or N. Use Y to specify REPORT YES or N to specify REPORT NO.
UX019	KEYCARD	Y or N. Use Y to specify KEYCARD or N to specify no keyword.
UX020	FREQVAL	Indicates that frequency statistics are to be gathered from the specified column group. <ul style="list-style-type: none"> <li>• Y - YES, collect frequency statistics</li> <li>• N - NO, do not collect frequency statistics</li> </ul>
UX021	NUMCOLS	A numeric value. This value is used as a parameter to NUMCOLS. For example: FREQVAL NUMCOLS 9 COUNT.
UX022	COUNT	Indicates the number of frequently occurring values to be collected from the specified column group.
UX023	UPDATE	A, P, S, or N. Use A to specify UPDATE ALL, P to specify UPDATE ACCESSPATH, S to specify UPDATE, or N to specify UPDATE NONE.
UX024	HISTORY	A, P, S, or N. Use A to specify HISTORY ALL, P to specify HISTORY ACCESSPATH, S to specify HISTORY SPACE, or N to specify HISTORY NONE.
UX025	FORCEROLLUP	Y or N. Use Y to specify FORCEROLLUP YES or N to specify FORCEROLLUP NO.
UX026	SORTDEVT	A device type. This value is used as a parameter to SORTDEVT. For example: SORTDEVT SYSALLDA.
UX027	SORTNUM	1- 255. This value is used as a parameter to SORTNUM. For example: SORTNUM 12.
UX028	CLONE	Y or N. Use Y to specify CLONE or N to specify no keyword.
UX029	FLASHCOPY	Y, C, or N. Use Y to specify FLASHCOPY YES, C to specify FLASHCOPY CONSISTENT, or N to specify FLASHCOPY NO.
UX030	HISTOGRAM	<ul style="list-style-type: none"> <li>• Y - YES, gather histogram statistics from the specified column group</li> <li>• N - NO, do not gather such statistics</li> </ul>
UX031	NUMCOLS	The number of key columns that are to be concatenated when collecting histogram statistics from the specified index.
UX032	NUMQUANTILES	Indicates the number of quantiles that the utility collects.

Table 12. Alternate shared variable input data (continued)		
UXO40	LOGRANGES	<ul style="list-style-type: none"> <li>• Y - Yes, REORG uses SYSLGRNX information for the LOG phase whenever possible. This option is the default behavior.</li> <li>• N - NO, REORG does not use SYSLGRNX information for the LOG phase.</li> </ul>
UXO42	SWITCHTIME	<ul style="list-style-type: none"> <li>• N - NONE, does not specify a time for the final log iteration of the LOG phase. This option is the default behavior. Specifies the time that the final log iteration of the LOG phase is to begin. This time must not have already occurred when REORG is run.</li> <li>• labeled-duration-expression, SWITCHTIME labeled-duration-expression is added.</li> </ul>
UXO43	NEWMAXRO	<ul style="list-style-type: none"> <li>• N - NONE, specifies that when the specified SWITCHTIME is met, REORG proceeds to the last log iteration without taking log processing time into consideration. This option is the default.</li> <li>• Integer, specifies the number of seconds. Valid values are 0 through 2147483647.</li> </ul>
UXORBALR	RBALRSN	<p>Specifies the RBA and LRSN format in which the target object is to be left after a REORG.</p> <ul style="list-style-type: none"> <li>• N - None No conversion</li> <li>• B - Basic Convert to a basic format</li> <li>• E - Extended Convert to extended format</li> </ul>

## Creating user-defined templates

You can create a data set template in Db2 Object Comparison Tool to save Db2 Admin Tool Change Management batch parameter variables. After you define a data set with Db2 TEMPLATE statements, you can reuse these template statements in apply jobs.

### About this task

Without a reusable template, the settings of each new apply job that you run overwrite the settings of your previous apply job. To create a reusable template, you must save the Change Management batch variables in USRTEMPL DD. USRTEMPL and ADB25TU templates can be used at the same time. USRTEMPL templates take precedence over 25TU templates.

### Procedure

1. Create a data set and name it ADBPRE.USRTEMPL.
  - a) Define the logical record length of 80.
  - b) Enter the prefix value ADBPRE in panel GOC5, ADBPALT, or ADB2C11A, depending on the method you are using to run the apply job.
2. Add the Change Management batch parameters and variables for the templates that you want to use. In the USRTEMPL data set, you must set the parameters before adding the templates. You can store the following Change Management batch parameters in the USRTEMPL data set:
  - **UTIL\_TEMPLATE\_DISCARD\_NAME**
  - **UTIL\_TEMPLATE\_DISCARD\_USE**
  - **UTIL\_CLONE\_TEMPLATE\_DISCARD\_NAME**
  - **UTIL\_CLONE\_TEMPLATE\_DISCARD\_USE**
  - **UTIL\_TEMPLATE\_DISCARD\_NC\_NAME**

- UTIL\_TEMPLATE\_DISCARDDDNC\_USE
- UTIL\_CLONE\_TEMPLATE\_DISCARDDDNC\_NAME
- UTIL\_CLONE\_TEMPLATE\_DISCARDDDNC\_USE
- UTIL\_TEMPLATE\_UNLOAD\_PUNCHDDN\_NAME
- UTIL\_TEMPLATE\_UNLOAD\_PUNCHDDN\_USE
- UTIL\_CLONE\_TEMPLATE\_UNLOAD\_PUNCHDDN\_NAME
- UTIL\_CLONE\_TEMPLATE\_UNLOAD\_PUNCHDDN\_USE
- UTIL\_TEMPLATE\_UNLOAD\_PUNCHDDNC\_NAME
- UTIL\_TEMPLATE\_UNLOAD\_PUNCHDDNC\_USE
- UTIL\_CLONE\_TEMPLATE\_UNLOAD\_PUNCHDDNC\_NAME
- UTIL\_CLONE\_TEMPLATE\_UNLOAD\_PUNCHDDNC\_USE
- UTIL\_TEMPLATE\_UNLOAD\_UNLDDN\_NAME
- UTIL\_TEMPLATE\_UNLOAD\_UNLDDN\_USE
- UTIL\_TEMPLATE\_UNLOAD\_UNLDDNC\_NAME
- UTIL\_TEMPLATE\_UNLOAD\_UNLDDNC\_USE
- UTIL\_CLONE\_TEMPLATE\_UNLOAD\_UNLDDN\_NAME
- UTIL\_CLONE\_TEMPLATE\_UNLOAD\_UNLDDNC\_NAME
- UTIL\_TEMPLATE\_COPYDDN1\_NAME
- UTIL\_CLONE\_TEMPLATE\_COPYDDN1\_NAME
- UTIL\_TEMPLATE\_COPYDDN2\_NAME
- UTIL\_CLONE\_TEMPLATE\_COPYDDN2\_NAME
- UTIL\_TEMPLATE\_ERRDDN\_NAME
- UTIL\_CLONE\_TEMPLATE\_ERRDDN\_NAME
- UTIL\_TEMPLATE\_FCCOPYDDN\_NAME
- UTIL\_CLONE\_TEMPLATE\_FCCOPYDDN\_NAME
- UTIL\_TEMPLATE\_LOBCOL\_NAME
- UTIL\_CLONE\_TEMPLATE\_LOBCOL\_NAME
- UTIL\_TEMPLATE\_MAPDDN\_NAME
- UTIL\_CLONE\_TEMPLATE\_MAPDDN\_NAME
- UTIL\_TEMPLATE\_PUNCHDDN\_NAME
- UTIL\_CLONE\_TEMPLATE\_PUNCHDDN\_NAME
- UTIL\_TEMPLATE\_RECOVERYDDN1\_NAME
- UTIL\_CLONE\_TEMPLATE\_RECOVERYDDN1\_NAME
- UTIL\_TEMPLATE\_RECOVERYDDN2\_NAME
- UTIL\_CLONE\_TEMPLATE\_RECOVERYDDN2\_NAME
- UTIL\_TEMPLATE\_UNLDDN\_NAME
- UTIL\_CLONE\_TEMPLATE\_UNLDDN\_NAME
- UTIL\_TEMPLATE\_WORKDDN1\_NAME
- UTIL\_CLONE\_TEMPLATE\_WORKDDN1\_NAME
- UTIL\_TEMPLATE\_WORKDDN2\_NAME
- UTIL\_CLONE\_TEMPLATE\_WORKDDN2\_NAME
- UTIL\_CLONE\_TEMPLATE\_WORKDDN2\_NAME
- UTIL\_TEMPLATE\_XMLCOL\_NAME

- UTIL\_CLONE\_TEMPLATE\_XMLCOL\_NAME
- UTIL\_TEMPLATE\_COPYDDN1\_USE
- UTIL\_CLONE\_TEMPLATE\_COPYDDN1\_USE
- UTIL\_TEMPLATE\_COPYDDN2\_USE
- UTIL\_CLONE\_TEMPLATE\_COPYDDN2\_USE
- UTIL\_TEMPLATE\_ERRDDN\_USE
- UTIL\_CLONE\_TEMPLATE\_ERRDDN\_USE
- UTIL\_TEMPLATE\_FCCOPYDDN\_USE
- UTIL\_CLONE\_TEMPLATE\_FCCOPYDDN\_USE
- UTIL\_TEMPLATE\_LOBCOL\_USE
- UTIL\_CLONE\_TEMPLATE\_LOBCOL\_USE
- UTIL\_TEMPLATE\_MAPDDN\_USE
- UTIL\_CLONE\_TEMPLATE\_MAPDDN\_USE
- UTIL\_TEMPLATE\_PUNCHDDN\_USE
- UTIL\_CLONE\_TEMPLATE\_PUNCHDDN\_USE
- UTIL\_TEMPLATE\_RECOVERYDDN1\_USE
- UTIL\_CLONE\_TEMPLATE\_RECOVERYDDN1\_USE
- UTIL\_TEMPLATE\_RECOVERYDDN2\_USE
- UTIL\_CLONE\_TEMPLATE\_RECOVERYDDN2\_USE
- UTIL\_TEMPLATE\_UNLDDN\_USE
- UTIL\_CLONE\_TEMPLATE\_UNLDDN\_USE
- UTIL\_TEMPLATE\_WORKKDDN1\_USE
- UTIL\_CLONE\_TEMPLATE\_WORKKDDN1\_USE
- UTIL\_TEMPLATE\_WORKKDDN2\_USE
- UTIL\_CLONE\_TEMPLATE\_WORKKDDN2\_USE
- UTIL\_TEMPLATE\_XMLCOL\_USE
- UTIL\_CLONE\_TEMPLATE\_XMLCOL\_USE

3. Add the template by typing one of the following formats in the data set:

- Db2 template format

```
TEMPLATE template_name
DSN dsn_definition
template_details
```

- XML template format

```
<TEMPLATE>
<NAME>
template_name
</NAME>
<DSN>
dsn_definition
</DSN>
<OTHER>
template_details
</OTHER>
</TEMPLATE>
```

## What to do next

Now you can use USRTEMPL data sets to run different apply jobs without losing the template settings for each job due to overwrites. Reusable templates are useful if you are frequently running more than one apply job.

**Related information**

[Using DB2 templates: Change Management \(CM\) batch interface \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

[CM batch parameter definitions \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)



# Chapter 18. Recommendations when comparing a large number of objects

Enterprise Resource Planning (ERP) systems typically have a large number of objects.

When you use Db2 Object Comparison Tool to compare a large number of objects, consider the following recommendations:

- If online compare fails with an ONCODE=451 (out of storage), try running the comparison as a batch job. Using the same version files, a batch object compare job running under an initiator usually runs to completion. The online compare failure is due to TSO and ISPF control blocks, tasks and code that are not present in a batch address space, and limits on the region size for the TSO address space in which online compare is running.
- Specify a large region size on the job card to ensure that the batch job can get sufficient virtual storage. If possible, specify OM.
- Ensure that your batch jobs can get sufficient CPU time. When you compare a large number of objects, you might, depending on your installation settings and processor speed, need to add a TIME=*n* option on your job card. The recommended initial value for *n* is 300 (CPU minutes).
- Ensure that the data sets for the version file output are large enough to contain the data for the objects. If the data sets are not large enough, Step 1 or Step 2 of the compare batch job can terminate with a x37 abend. To prevent this storage problem, modify the JCL before submitting the job to use PACE=(CYL,(10,100)) for the following data sets:
  - CAT (in two places)
  - SRCSIN
  - SRCSOUT
  - TGTSOUT
- To avoid data set extension failures caused by referback, allocate the data set in advance.

The following JCL shows how the JCL should look before and after you modify it:

## Before

```
//CAT      DD DSN= ....
//          DISP=(NEW,CATLG,DELETE),
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//          SPACE=(CYL,(10,10),RLSE),
//          UNIT=SYSDA
(in two places)
...
//SRCSIN   DD DSN=&SRCSIN,DISP=(,DELETE),
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//          SPACE=(CYL,(10,20),RLSE),
//          UNIT=SYSALLDA
//SRCSOUT  DD DSN=&SRCSOUT,DISP=(,DELETE),
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//          SPACE=(CYL,(10,20),RLSE),
//          UNIT=SYSALLDA
//TGTSIN   DD DISP=SHR,
//          DSN= ....
//TGTSOUT  DD DSN=&TGTSOUT,DISP=(,DELETE),
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//          SPACE=(CYL,(10,20),RLSE),
//          UNIT=SYSALLDA
```

## After

```
//CAT      DD DSN= ....
//          DISP=(NEW,CATLG,DELETE),
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//          SPACE=(CYL,(10,100),RLSE),
//          UNIT=SYSDA
(in two places)
```

```

...
//SRCSIN DD DSN=&SRCSIN,DISP=(,DELETE),
//        DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//        SPACE=(CYL,(10,100),RLSE),
//        UNIT=SYSALLDA
//SRCSOUT DD DSN=&SRCSOUT,DISP=(,DELETE),
//        DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//        SPACE=(CYL,(10,100),RLSE),
//        UNIT=SYSALLDA
//TGTSIN DD DISP=SHR,
//        DSN= ...
//TGTSOUT DD DSN=&TGTSOUT,DISP=(,DELETE),
//        DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
//        SPACE=(CYL,(10,100),RLSE),
//        UNIT=SYSALLDA

```

---

## Chapter 19. Troubleshooting and messages

When customizing or using Db2 Object Comparison Tool, you will likely receive messages that begin with the prefix ADB or CCQ.

Messages with a prefix of ADB are from IBM Db2 Administration Tool for z/OS. For information about those messages, see [Db2 Admin Tool messages \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

Messages with a prefix of CCQ are from Tools Customizer (TCz). For information about these messages, see [Tools Customizer messages \(IBM Tools Customizer for z/OS 1.1\)](#). For additional troubleshooting help with TCz, see [Tools Customizer troubleshooting \(IBM Tools Customizer for z/OS 1.1\)](#).

Db2 Object Comparison Tool does not have its own message prefix.

### Related information

[Gathering diagnostic information \(IBM Db2 Administration Tool for z/OS 13.1.0\)](#)

---

## Db2 Object Comparison Tool condition codes

Object Comparison Tool programs return condition codes to indicate whether the program completed successfully. If you receive a condition code greater than zero, review the messages carefully.

### ADB2GEN condition codes

The ADB2GEN program is used to create a version file. The following condition codes are issued by ADB2GEN:

**0**

The program ran successfully.

**4**

One of the following conditions occurred:

- A parameter error occurred. The parameter was ignored or the default was used. No generate requests were issued.
- A requested object was not found. A warning is issued.

**8**

One of the following conditions occurred:

- No parameters were found. Processing ended.
- The Db2 version is not supported. Other errors might be issued.

**12**

One of the following conditions occurred:

- The Db2 version is not supported. Processing ended.
- The remote location is not defined or is not a Db2 for z/OS system. This error is an internal error or limitation.
- Other severe errors were detected.

**16**

A severe error occurred.

### GOC2CMP condition codes

The following condition codes are issued by the GOC2CMP program:

**0**

The program ran successfully.

4

The index was not dropped, which can lead to loss of referential integrity. See the listed error message.

6

SQL PL functions have been bypassed, because the BYPASSSQLPL parameter was specified. Examine the generated APPLY job or work statement list to verify that the content is complete.

8

Problems with referential constraints were detected. Manual action is required. See the listed error message.

16

A severe error occurred. See the listed error message.

### GOC2DTC condition codes

The GOC2DTC program can issue the following condition code:

12

A string (enclosed in quotation marks) is too long.

## Troubleshooting: The Compare report shows changes to bind options for trigger packages

If the report from Object Comparison Tool includes unexpected changes to bind options for trigger packages, you might need to rebind some packages. This situation can occur when you migrate to Db2 11 or a later version.

**Symptom:** The compare report includes unexpected changes to bind options for trigger packages, as shown in the following example:

```
Compare Trigger source <table_schema>.<table_name> and target <table_schema>.<table_name>
Source type : <trigger_type> Target type : <trigger_type>
(A)Field SYSTEM_TIME SENSITIVE changed from YES to NO
(A)Field BUSINESS_TIME SENSITIVE changed from YES to NO
(A)Field ARCHIVE SENSITIVE changed from YES to NO
```

**Explanation:** When a trigger is created, the following fields in the SYSPACKAGE table have a default value of YES:

- SYSTIMESENSITIVE
- BUSTIMESENSITIVE
- ARCHIVESENSITIVE

These values are stored in packages at the time they are bound. Check these field attributes in your trigger packages. If you set them to NO on your old system and then these trigger packages are created on a new system, the default values of these fields on the new system are YES.

**Solution:** Rebind the packages.

To rebind the packages:

1. On the **DB2 Administration Menu (ADB2)** panel, specify option I, and press Enter.
2. On the **DB2I PRIMARY OPTION MENU** panel, specify option 5, and press Enter.
3. On the **BIND/REBIND/FREE** panel, specify option 6, and press Enter.
4. Change the **PLAN MANAGEMENT** field to OFF.
5. Change the **SYSTEM\_TIME SENSITIVE**, **BUSINESS\_TIME SENSITIVE**, and **ARCHIVE SENSITIVE** field to the values that you want.

### Related information

[BIND and REBIND options for packages, plans, and services \(Db2 13 for z/OS\)](#)

## Performance considerations for Db2 Object Comparison Tool

The performance of Db2 Object Comparison Tool can be impacted by several factors.

For optimal performance, avoid the following situations if possible:

- Large lists of translation masks
- Comparisons of many objects (especially many views)
- A large number of changes

If you have many objects to compare, consider running multiple operations instead of doing all the comparisons in a single operation.

## Recommendations when comparing a large number of objects

Enterprise Resource Planning (ERP) systems typically have a large number of objects.

When you use Db2 Object Comparison Tool to compare a large number of objects, consider the following recommendations:

- If online compare fails with an ONCODE=451 (out of storage), try running the comparison as a batch job. Using the same version files, a batch object compare job running under an initiator usually runs to completion. The online compare failure is due to TSO and ISPF control blocks, tasks and code that are not present in a batch address space, and limits on the region size for the TSO address space in which online compare is running.
- Specify a large region size on the job card to ensure that the batch job can get sufficient virtual storage. If possible, specify 0M.
- Ensure that your batch jobs can get sufficient CPU time. When you compare a large number of objects, you might, depending on your installation settings and processor speed, need to add a TIME=*n* option on your job card. The recommended initial value for *n* is 300 (CPU minutes).
- Ensure that the data sets for the version file output are large enough to contain the data for the objects. If the data sets are not large enough, Step 1 or Step 2 of the compare batch job can terminate with a x37 abend. To prevent this storage problem, modify the JCL before submitting the job to use PACE=(CYL,(10,100) for the following data sets:
  - CAT (in two places)
  - SRCSIN
  - SRCSOUT
  - TGTSOUT
- To avoid data set extension failures caused by referback, allocate the data set in advance.

The following JCL shows how the JCL should look before and after you modify it:

### Before

```
//CAT DD DSN= . . . .
// DISP=(NEW,CATLG,DELETE),
// DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
// SPACE=(CYL,(10,10),RLSE),
// UNIT=SYSDA
(in two places)
. . .
//SRCSIN DD DSN=&SRCSIN,DISP=(,DELETE),
// DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
// SPACE=(CYL,(10,20),RLSE),
// UNIT=SYSALLDA
//SRCSOUT DD DSN=&SRCSOUT,DISP=(,DELETE),
// DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
// SPACE=(CYL,(10,20),RLSE),
// UNIT=SYSALLDA
//TGTSIN DD DISP=SHR,
// DSN= . . . .
//TGTSOUT DD DSN=&TGTSOUT,DISP=(,DELETE),
// DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),
```

```
//      SPACE=(CYL,(10,20),RLSE),  
//      UNIT=SYSALLDA
```

## After

```
//CAT      DD DSN= ....  
//          DISP=(NEW,CATLG,DELETE),  
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),  
//          SPACE=(CYL,(10,100),RLSE),  
//          UNIT=SYSDA  
(in two places)  
....  
//SRCSIN   DD DSN=&SRCSIN,DISP=(,DELETE),  
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),  
//          SPACE=(CYL,(10,100),RLSE),  
//          UNIT=SYSALLDA  
//SRCSOUT  DD DSN=&SRCSOUT,DISP=(,DELETE),  
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),  
//          SPACE=(CYL,(10,100),RLSE),  
//          UNIT=SYSALLDA  
//TGTSIN   DD DISP=SHR,  
//          DSN= ....  
//TGTSOUT  DD DSN==&TGTSOUT,DISP=(,DELETE),  
//          DCB=(LRECL=16384,RECFM=VB,BLKSIZE=27998),  
//          SPACE=(CYL,(10,100),RLSE),  
//          UNIT=SYSALLDA
```

## Notices

---

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785 US

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785 US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

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

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as shown below:

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. (enter the year or years).

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

## Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)<sup>®</sup> are trademarks or registered marks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at: <http://www.ibm.com/legal/copytrade.shtml>.

Linux<sup>®</sup> is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

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

Java<sup>™</sup> and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

## Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions:

**Applicability:** These terms and conditions are in addition to any terms of use for the IBM website.

**Personal use:** You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

**Commercial use:** You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

**Rights:** Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

## **Privacy policy considerations**

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offerings use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Statement at <http://www.ibm.com/privacy>.



# Index

## A

- accessibility
  - overview [21](#)
- ADB2GEN
  - condition codes [219](#)
- ADB2GEN program [175](#)
- ADB2UCUS skeleton
  - configuration
    - explained [35](#)
    - restrictions [35](#)
    - SET statements [35](#)
  - configuration variables [35](#)
- ADBL CLIST [34, 38](#)
- alter identity column attributes [180](#)
- alternate form of syntax [18](#)
- alternate input data set
  - creating [195](#)
  - listing current values [195](#)
  - requirements [195](#)
- alternate input to the generate apply job [195](#)
- alternate statements [18](#)
- archive table
  - comparing [191](#)
- archive-enabled table
  - comparing [191](#)

## B

- batch Db2 catalog extraction programs
  - sample output [175](#)
- batch DDL file extraction programs
  - sample output [174](#)
- buffer pool analysis
  - overview [15](#)

## C

- CLIST library
  - VB version [30](#)
- Compare report
  - troubleshooting [220](#)
- comparing functions [180](#)
- comparing partitioned tables [180](#)
- comparing renamed objects [180](#)
- comparing table columns [180](#)
- comparing triggers [180](#)
- comparing version files [179](#)
- comparing views [180](#)
- components [17](#)
- condition codes [219](#)
- configuration
  - ADB2UCUS skeleton [35](#)
  - variables [35](#)
- constraint names [180](#)
- cookie policy [223, 225](#)
- CREATE TRIGGER [180](#)

- CREATE VIEW [180](#)
- creating an alternate input data set [195](#)
- creating version reports
  - from the Db2 catalog [175](#)
- customization
  - initial [25](#)
  - list of customization job members [32](#)
  - overview [23](#)
- Customization
  - checklist [23](#)
  - data set names [25](#)
  - software requirements [24](#)
- customization jobs
  - sort sequence [32](#)
  - submitting [32](#)

## D

- data sets
  - alternate values [195](#)
  - listing current values [195](#)
  - naming conventions, using ADB2UCUS [35](#)
- DB2 objects, large number of [217, 221](#)
- DDL
  - DROP statements [180](#)
- deleting jobs [28](#)
- dependencies
  - view dependent [180](#)
  - view dropped [180](#)
- documentation
  - accessing [20](#)
  - sending feedback [20](#)
- drop skip conversion [180](#)
- DROP statements in the source DDL [180](#)

## E

- ERP (Enterprise Resource Planning) [217, 221](#)
- EXEC library
  - VB version [30](#)

## F

- Finish Product Customization panel [32](#)
- first-time customization [25](#)
- functions
  - overview [15](#)
- functions, comparing [180](#)

## G

- general description [41](#)
- generate apply job, alternate input [195](#)
- GOC1 [56](#)
- GOC2CMP
  - condition codes [219](#)

GOCFB2VB job  
editing [30](#)

## I

installation  
    customizing Db2 Object Comparison Tool [25](#)  
ISPF interface  
    general description [17](#)

## J

JCL  
    customizing [34](#)

## L

legal notices  
    cookie policy [223](#), [225](#)  
    notices [223](#)  
    programming interface information [223](#)  
    trademarks [223](#), [224](#)  
libraries  
    allocating [34](#)  
listing current data set values [195](#)  
LOBs  
    changing tables [186](#)  
    restriction [175](#)  
    unloading tables [186](#)

## M

multiple-LPAR environments  
    customization [23](#), [25](#), [29](#), [32](#)

## N

native SQL procedures  
    comparing [180](#)  
    masking [180](#)  
notices [223](#)

## O

Object authorizations [180](#)  
objects  
    source definitions [56](#)  
online schema evolution  
    limitations  
        alter identity column attributes [180](#)  
        partition rotation [180](#)  
        partitioning key [180](#)  
overview [1](#)

## P

panels  
    Finish Product Customization [32](#)  
    Specify Compare Source (GOC1) [56](#)  
partition rotation [180](#)  
partitioned tables, comparing [180](#)  
partitioning key [180](#)

product parameters  
    changing [28](#)  
programming interface information [223](#)  
programs  
    ADB2GEN [175](#)  
    reverse engineering [175](#)

## R

recustomizing  
    Object Comparison Tool [28](#)  
renamed object [180](#)  
reports  
    batch Db2 catalog extraction programs [175](#)  
    creating versions from Db2 catalog [175](#)  
restoring objects [19](#)  
reverse engineering programs [175](#)  
roadmaps  
    customizing for the first time [25](#)  
    recustomizing Object Comparison Tool [28](#)

## S

scenarios [41](#)  
screen readers and magnifiers [21](#)  
skip conversion [180](#)  
software requirements [24](#)  
source [18](#)  
source objects [56](#)  
Specify Compare Source (GOC1) panel [56](#)  
SQL statements  
    CREATE TRIGGER [180](#)  
    CREATE VIEW [180](#)  
suppress DROP of objects [18](#)  
suppressing DROP, sequence [180](#)

## T

table columns, comparing [180](#)  
tables  
    comparing partitioned [180](#)  
target [18](#)  
terminology  
    source [18](#)  
    suppress DROP of objects [18](#)  
    target [18](#)  
    version file [18](#)  
Tools Customizer  
    list of customization job members [32](#)  
trademarks [223](#), [224](#)  
trigger creation, sequence [180](#)  
triggers, comparing [180](#)  
troubleshooting [219](#)

## U

user-defined template [212](#)  
using alternate forms of syntax [18](#)

## V

version file [18](#)  
version files

version files (*continued*)  
  advantages [19](#)  
  creating [19, 20](#)  
  explained [19](#)  
version files, comparing [179](#)  
views, comparing [180](#)







Product Number: 5655-CH1

SC28-2787-00

