IBM DB2 Universal Database

Migration Guide: Migrating to DB2 Replication Version 8

REPL-MIG8-02

IBM DB2 Universal Database

Migration Guide: Migrating to DB2 Replication Version 8

REPL-MIG8-02

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 95.

This document is protected by copyright law. The information contained in this publication does not include any product warranties, and any statements provided in this manual should not be interpreted as such.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2002, 2003. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

© Copyright IBM Corp. 2002, 2003

About this document

This document describes how to migrate an existing DB2 replication environment to Version 8 replication. DB2 replication is available for various systems. It is available as an integrated part of DB2 Universal Database for workstation operating-system environments, and as separately orderable products for z/OS™ and OS/400® operating systems.

This document also describes how to migrate replication environments that currently use DB2 DataJoiner® to replicate data to or from non-DB2 relational servers.

This document does not describe how to migrate your existing VM/VSE replication environment to Version 8 replication. Version 8 function is not available on VM/VSE. To administer your VM/VSE replication objects, continue using your current environment.

This document does not describe how to migrate your existing satellite replication environment. The ASNSAT command is not supported in Version 8. Also, the ability to generalize replication subscriptions and set up a DB2 satellite replication environment is no longer available from the Satellite Administration Center. If you require data replication for a mobile work force, consider moving your satellite DB2 databases to DB2 Everyplace™, Version 8. For additional information, contact your IBM representative.

This document is available online only in PDF and HTML formats.

Who should read this document

This document is written for those DB2 database administrators and systems administrators who must plan the migration to Version 8 for DB2 replication and those who must perform the migration. It assumes that you are familiar with DB2 replication terminology, and that you have experience administering replication.

How to use this book

Read all of the following sections in this document before you use any migration tools:

- Chapter 1, "Overview of the typical migration process," on page 1
- Chapter 2, "Planning your migration," on page 7
- Chapter 3, "Checklists for migrating your servers," on page 25

© Copyright IBM Corp. 2002, 2003

Then follow the detailed instructions in the appropriate chapter that pertains to the server that you want to migrate:

- Chapter 4, “Migrating iSeries servers to Version 8,” on page 31
- Chapter 6, “Migrating Linux, UNIX, and Windows servers to Version 8,” on page 55
- Chapter 5, “Migrating z/OS servers to Version 8,” on page 41
- Chapter 7, “Migrating DataJoiner servers to Version 8,” on page 69

If you get any messages during the migration process, refer to Chapter 8, “Migration messages,” on page 83 for information.

After you migrate, refer to *DB2 Universal Database Replication Guide and Reference, SC27–1121* for information about using DB2 replication.

Terminology

The following replication terms are new or changed as of Version 8:

Apply control server

A database that contains the Apply control tables.

Before Version 8, this database was known as the *control server*.

Capture control server

(1) A database that contains the Capture control tables. (2) A system where the Capture program is running.

Before Version 8, this database was known as the *source server*.

Capture schema

The schema for the Capture control tables used by a particular instance of the Capture program. By default, the schema is ASN. Before Version 8, only one Capture schema was supported.

DataJoiner servers

A replication Capture control server, Apply control server, or target server on DataJoiner 2.1.1.

Federated servers

1. The DB2 server in a federated system. Any number of DB2 instances can be configured to function as federated servers.

2. A federated server in DB2 Version 8.1 includes the federated function that was previously available on DataJoiner.

Federated system

A special type of distributed database management system (DBMS). A federated system allows you to query and manipulate data located on other servers. The data can be in database managers such as Oracle, Sybase, Informix, and Microsoft® SQL Server, or it can be in lists or stores such as a spreadsheet, Web site, or data mart. A federated system consists of a DB2 instance that will operate as a server, a database that will serve as the federated database, one or more data sources, and clients (users and applications) who will access the database and data sources.

non-DB2 relational servers
Sources or targets that are in database management systems such as Oracle, Sybase, Informix, and Microsoft SQL Server.

How to send your comments

Your feedback helps IBM to provide quality information. Please send any comments that you have about this document or other DB2 DataPropagator documentation. You can use any of the following methods to provide comments:

- Send your comments from the Web. Visit the Web site at:

  www.ibm.com/software/data/dpropr/

  The Web site has a feedback page that you can use to enter and send comments.
- Send your comments by e-mail to comments@vnet.ibm.com. Be sure to include the name of the product, the version number of the product, and the name and part number of the book (if applicable). If you are commenting on specific text, please include the location of the text (for example, a chapter and section title, a table number, a page number, or a help topic title).

What's new?

After DB2 replication Version 8.1 became generally available, some changes were made to the migration process. The changes are introduced in the following list and are described later in this document:

The BACKUP.GRANTS table now stores pre-migration grants (z/OS): PTF UQ78718 for DataPropagator for z/OS Version 8 updates the `mig8zos.sql` script so that the script saves the grants for replication control tables and CD tables that were on your system before you migrated to replication Version 8. The script saves the information in a new table, BACKUP.GRANTS. You can use the information in the BACKUP.GRANTS table to restore grants during fallback, or to configure grants on Version 8 tables after you migrate your replication environment.

LONG VARCHAR or LONG VARCHAR FOR BIT DATA data types are no longer supported in nicknames (Federated): In DB2 Information Integrator Version 8, LONG VARCHAR and LONG VARCHAR FOR BIT DATA data types in nicknames are no longer supported. There are certain LONG data type replication restrictions when migrating from DataJoiner to DB2 Information Integrator. During migration, any nickname that is a LONG VARCHAR data type becomes a CLOB data type, and any nickname that is a LONG VARCHAR FOR BIT DATA data type becomes a BLOB data type. If your source data is shorter than 32673 bytes, you can alter the CLOB or BLOB nickname data types to VARCHAR(32672) or VARCHAR FOR BIT DATA(32762), respectively, after you migrate DB2. However, if you use VARCHAR(32672) or VARCHAR FOR BIT DATA(32762), any data exceeding 32672 bytes will be truncated and lost during replication.

Availability of data on Oracle sources is improved (Federated): Beginning with DB2 Universal Database Version 8.1.4 (FixPak 4), the Apply program no longer needs to issue lock table statements for CCD tables on Oracle sources. To take advantage of this improvement, after you migrate your replication environment to Version 8 and install FixPak 4, you must also migrate any existing registrations and subscriptions for Oracle sources.

© Copyright IBM Corp. 2002, 2003

Chapter 1. Overview of the typical migration process

DB2 replication is installed as part of DB2 on Linux, UNIX, and Windows, and as a separate DB2 DataPropagator product on z/OS and iSeries. Before you do any installs, you must apply maintenance on your existing DB2 replication and perform other pre-migration activities as described later in this document. After you complete the pre-migration activities, you can install the appropriate products that contain DB2 replication.

You can migrate your existing replication configuration (including registrations and subscriptions) using Version 8 migration tools after DB2 replication is installed. If you do *not* migrate your replication configuration, you must drop your pre-Version 8 CD tables and replication control tables, create a set of Version 8 replication control tables, create all your registrations and subscription sets in the new environment, and perform a full refresh of your target tables before you can use Version 8 replication with your existing registrations and subscriptions.

After you make the necessary preparations to the server that you are about to migrate, you can use the migration tools to first back up your replication environment and then to migrate it.

By running a migration tool to *back up* your replication environment, you create a temporary migration environment using a backup schema of your choice. The temporary migration environment contains migration control tables. It also contains a copy of your existing replication control tables, and a copy of your new Version 8 tables. If your server contained change-data (CD) tables, the temporary migration environment also contains a copy of your existing CD tables. Your existing replication environment is *not* changed during backup. The migration tools do not touch any source tables, target tables, or CCD tables. New temporary Version 8 replication control tables are created and populated. Views are created on the backup CD tables. These views will be used to create the Version 8 CD tables. (See Figure 1 on page 2.)

© Copyright IBM Corp. 2002, 2003

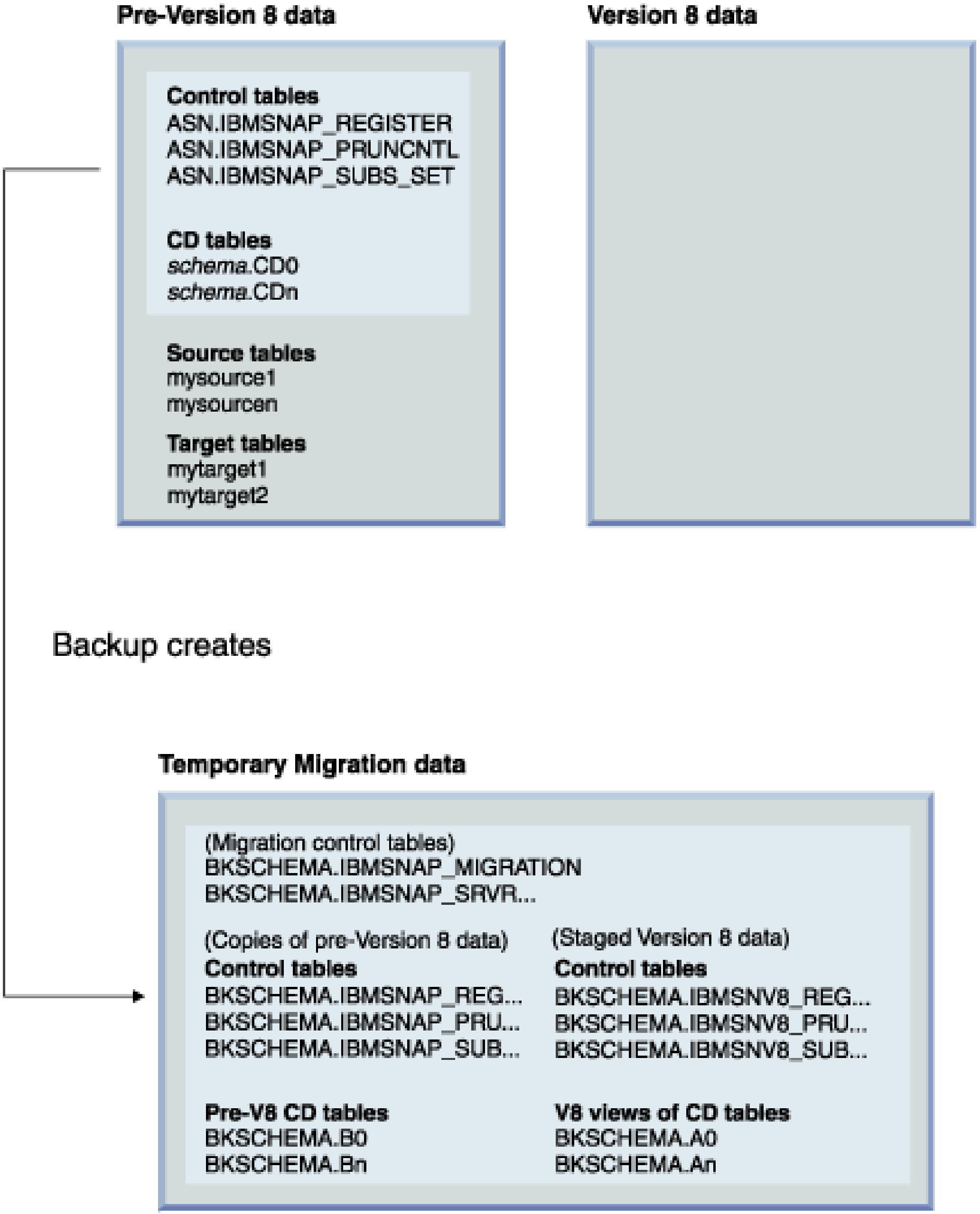

Figure 1. The backup step creates your migration environment, leaving your existing replication environment unchanged.

By running the migration tools to *migrate* your replication environment, you create the new Version 8 control tables and drop the pre-Version 8 control tables. If the server contains CD tables, migration also creates new Version 8 CD tables and drops the pre-Version 8 CD tables. (See Figure 2 on page 4.) The application data in your source and target tables is not changed.

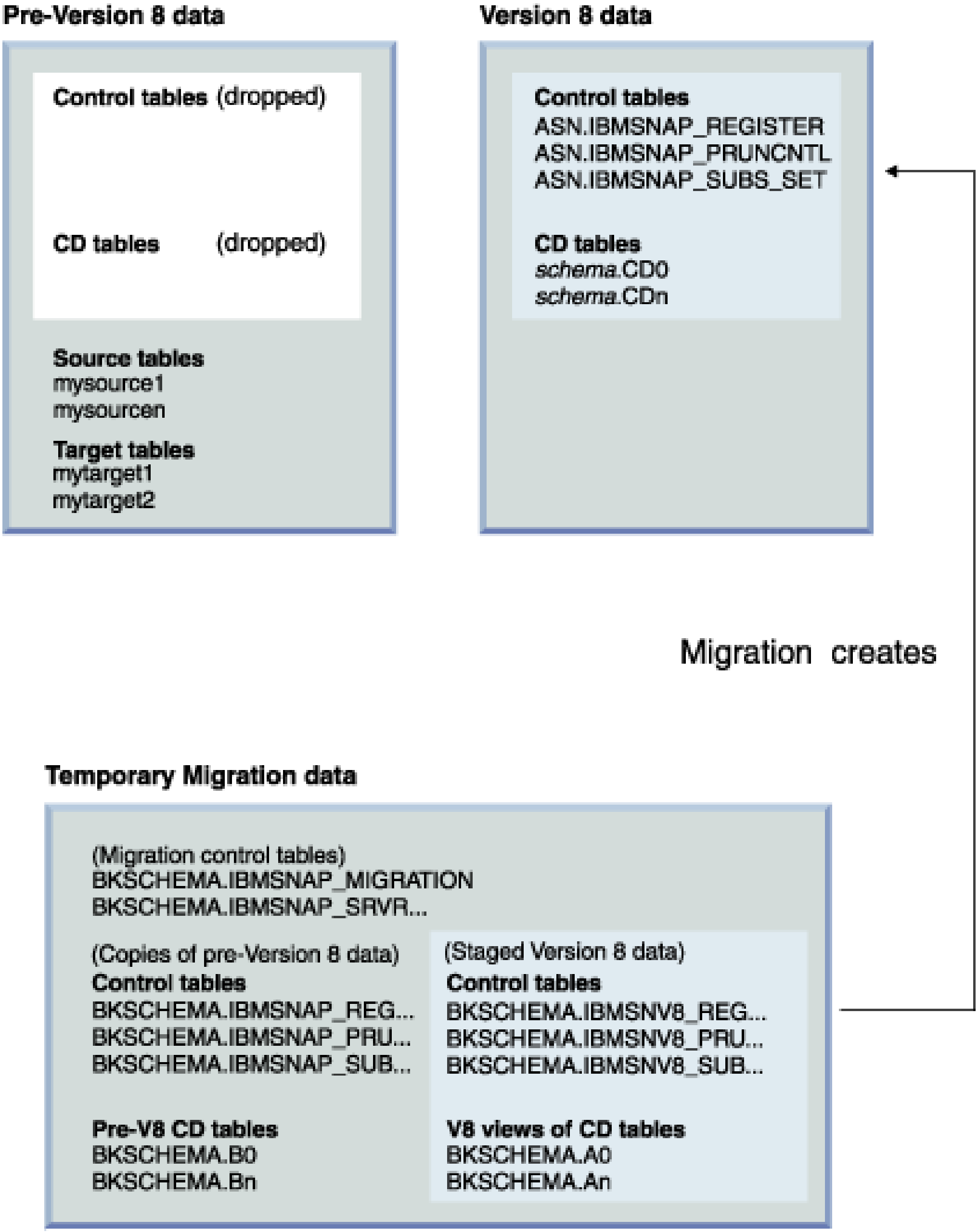

Figure 2. The migration step creates your Version 8 tables, and drops your pre-Version 8 tables.

Migration sets up the new Version 8 tables and populates them using information from your migration control environment. Registrations and subscription sets that you had in your previous environment are in your migrated Version 8 environment. Your source tables, CCD tables, and target tables are not changed during the migration.

If you are migrating iSeries servers, you also enable the Version 8 Capture and Apply programs during the migration step. After migration, the pre-Version 8 replication programs and iSeries DataPropagator administration native commands are removed from your system.

The migration process provides for some flexibility so that you can migrate Capture control servers and Apply control servers on different systems independently and still replicate data. For example, assume that you have a Version 7 Apply control server on one system and a Version 7 Capture control server on another. You can migrate the Apply control server first without having to migrate the Capture control server. Or, you can migrate the Capture control server first without having to migrate the Apply control server. More importantly, you can replicate data between the Version 8 Apply and the Version 7 Capture. In some cases you can also replicate from Version 8 Capture to Version 7 Apply, but you must apply maintenance to enable such coexistence. If you choose to migrate your environment in stages, you will have limited Version 8 function until you migrate completely to Version 8.

After migration, you can configure your new environment based on your needs. If you want to change the default values that are provided for new columns in Version 8 control tables, you can manually update the Version 8 control tables before you start replication. Defaults are assigned for the new behavior to match as closely as possible the existing settings that you have in your current environment. Depending on your pre-Version 8 environment, you might need to make some manual updates to the new Version 8 control tables for settings that are not handled by migration.

It is recommended that you practice using the migration tools in a test environment. If migration fails, in most cases you can use the migration tools to fall back to your pre-Version 8 environment. You *cannot* use fallback for databases on Linux, UNIX, and Windows that are running the Capture program because the Version 8 Capture program must use the Version 8 replication control tables. The fallback step restores the pre-Version 8 tables, and deletes the new Version 8 tables. Nothing is changed in your temporary migration environment. The backup copies of the pre-Version 8 tables and staged Version 8 tables remain after fallback. (See Figure 3 on page 6.)

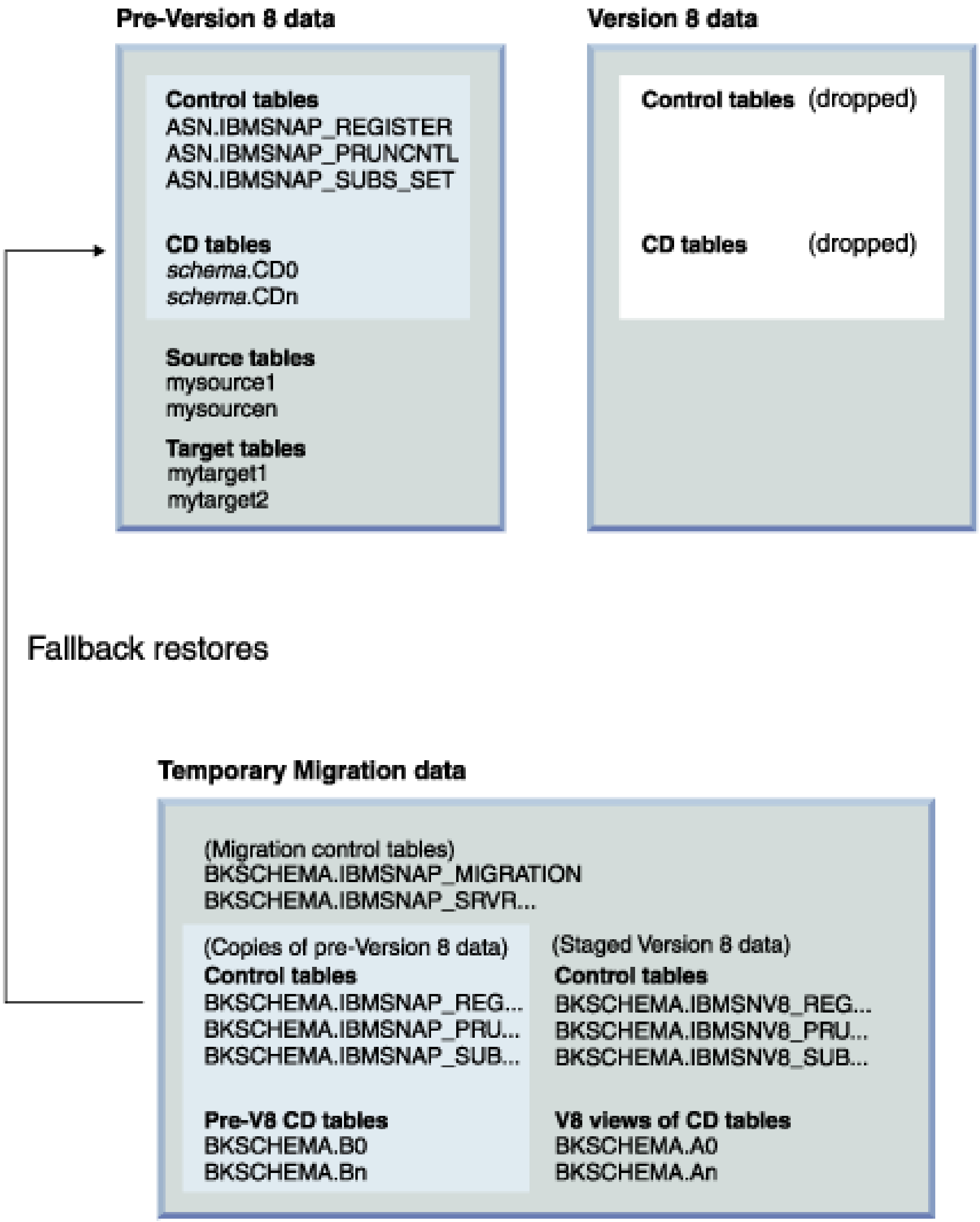

Figure 3. The fallback step re-creates your pre-Version 8 tables, and drops your Version 8 tables.

After a successful migration, and after you are sure that you do not want to fall back to your pre-Version 8 environment, you can remove the temporary migration control environment.

Chapter 2. Planning your migration

This section describes information that you should consider before you run any migration tools. It describes the planning that you must do to ensure a successful migration.

- “Avoiding common migration problems”
- “Supported DB2 versions” on page 8
- “Unsupported replication migration configurations” on page 9
- “Planning server migration in distributed environments” on page 12
- “Planning replication administration during migration” on page 18
- “Planning post-migration work” on page 18

Avoiding common migration problems

To prevent problems from occurring during the migration to Version 8, follow these instructions:

- Read Chapter 1, “Overview of the typical migration process,” on page 1 to understand what is involved in a typical migration and how your environment is affected.
- Read this planning chapter and plan your migration *before* installing any products. Ensure that your replication environment meets all prerequisites before you proceed with the migration steps. Prerequisites include installing maintenance for some existing replication programs and triggers or stored procedures. If you do not address the prerequisites before you migrate, migration will fail; you might need to perform a full refresh and perform some manual tasks to get replication running correctly in your environment.
- Before running any of the replication migration tools, make sure to plan enough space for the temporary tables and final Version 8 replication tables. On UNIX®, Windows®, z/OS, also make sure that you correctly allocate table spaces for the migration environment. If you start running the migration tools without adequate space, you might need to perform certain steps over again. Also, make sure that you follow the pruning recommendations before using the replication migration tools for each platform. Pruning minimizes the storage needed and the time required for the migration process.
- Use Chapter 3, “Checklists for migrating your servers,” on page 25 to ensure that you cover everything during your migration. Follow the migration steps in the specified order. Do not skip a step unless it is optional or it does not apply to you.

© Copyright IBM Corp. 2002, 2003

- It is recommended that you perform the migration on a test system first.
- You can fall back to the pre-migration state of your replication environment, if necessary. You *cannot* use fallback for databases on Linux, UNIX, and Windows that are running the Capture program because the Version 8 Capture program uses only the Version 8 control tables. Once you start replicating in your Version 8 environment you should not use the **fallback** command. The fallback step involves dropping the Version 8 control tables and re-creating the pre-Version 8 control tables from the backup tables.
- Once you start running the replication migration tools for a given Capture control server or Apply control server, you cannot alter your replication configuration on that server or run the Capture and Apply programs on that server until the tools have completed successfully.
- If you must migrate your Capture and Apply control servers at different times from one another, plan to be in such a mixed environment for a short period of time. That is, do not replicate between Version 8 and pre-Version 8 control tables for long when the control tables for one server are migrated but the control tables for the other are not. If your environment contains different versions of Capture or Apply control servers, you cannot exploit the new Version 8 function and you cannot add or alter registrations or subscriptions.
- Because the CD and the control tables are dropped and re-created during the migration process, statistics that were gathered prior to migration are lost. To avoid slowing performance and exceeding resource limits, run the RUNSTATS utility on all the CD tables and the IBMSNAP_UOW table. The tables are pruned prior to migration and do not contain a representative amount of data needed for meaningful statistics. Therefore, run the RUNSTATS utility only *after* the tables contain substantial amounts of data after the migration process.
- Pre-Version 8 Capture on z/OS, Linux, UNIX, and Windows requires maintenance before you migrate to Version 8. After the maintenance is done, if you cold start the Capture program or use system commands to kill or cancel the Capture program, the log sequence number is not updated. Therefore, use the Capture stop command to stop Capture rather than using a system command to kill or cancel it. If a cold start becomes necessary, Capture must recalculate the value of the log sequence number before you continue with the migration.

Supported DB2 versions

For Linux, UNIX, and Windows operating systems, replication is part of the DB2 product. This means that migration of replication is dependent on a migration of DB2. Replication migration is supported when migrating the following DB2 databases to Version 8:

- DB2 for Linux (Intel), Version 6, Version 7

- DB2 for HP V11, Version 6, Version 7
- DB2 for Windows, Version 6, Version 7
- DB2 for AIX, Version 6, Version 7
- DB2 for Solaris, Version 6, Version 7
- DB2 for Linux for S/390 and zSeries, Version 7

Replication is part of the DB2 DataJoiner product. This means that migration of replication is dependent on migration of DB2 DataJoiner. Replication migration will be supported when the migration from DB2 DataJoiner to Version 8.1 is available for the following products:

- DataJoiner for AIX 2.1.1
- DataJoiner for NT 2.1.1
- DataJoiner for Solaris 2.1.1

Important: For migrating DataJoiner servers, see “Checklist for migrating DataJoiner servers” on page 28.

You can migrate to Version 8 replication from the following versions of DB2 DataPropagator for iSeries:

- DataPropagator for iSeries Version 7.1
- DataPropagator for iSeries Version 5.1

DB2 DataPropagator for iSeries Version 8.1 can run on OS/400 V5R2 or later.

You can migrate to Version 8 replication from the following products on OS/390:

- DB2 DataPropagator for OS/390 Version 6.1
- DB2 DataPropagator for OS/390 Version 7.1

DB2 DataPropagator for z/OS Version 8.1 can run on DB2 for z/OS Version 6 or later.

Unsupported replication migration configurations

The following replication configurations cannot be migrated to Version 8 replication:

DB2 for VM or VSE replication environment

Version 8 function is not available on the VM and VSE operating systems. To administer your VM/VSE replication objects, continue using your current environment.

DB2 Satellite Edition replication environment

The ASNSAT command is not supported in Version 8. Also, the ability to generalize replication subscriptions and set up a DB2 satellite replication environment is no longer available from the Satellite Administration Center. If you require data replication for a mobile work force, consider moving your satellite DB2 databases to DB2 Everyplace, Version 8. For additional information, contact your IBM representative.

Subscription-set types

If a subscription set was either created manually or created using the replication administration tools and subsequently manipulated manually, and it is not clear whether this subscription set is intended for read-only, peer-to-peer, or update-anywhere processing, it cannot be migrated.

During replication migration, existing subscription sets are migrated according to the values set in the WHOS_ON_FIRST column of the IBMSNAP_SUBS_SET table and a value is assigned for the new SET_TYPE column according to the following rules:

- Empty sets in either the ′F′ or ′S′ direction are dropped if no members and no statements are found.
- Sets with active ′F′ and ′S′ directions become update-anywhere subscription sets (SET_TYPE =U).
- Sets with an active ′S′ direction only, and inactive or nonexistent ′F′ directions, become read-only replicas (SET_TYPE = R). The migrated subscription set contains only the active ′S′ direction.
- Sets with an active ′F′ direction only, and inactive or nonexistent ′S′ directions, become peer-to-peer subscription sets (SET_TYPE = P). The migrated subscription set contains only the active ′F′ direction.

Any other configurations are migrated but a warning message is issued and the SET_TYPE column is left blank.

Sources and replicas residing in one database

In prior versions of replication, replica tables could exist in the same database as the source table. This is no longer true in Version 8. If you have this setup, delete these subscription sets and re-create them in Version 8 using different Capture schemas for each.

LOB columns in non-DB2 relational target servers

At this time, federated servers only support the Oracle NET8 wrapper for writing to LOB column types in the nickname. If you are replicating to non-DB2 relational targets where the source column is a LOB and the wrapper is not Oracle NET8, remove or modify the subscription to ensure that there is no LOB column in the subscription definition before you migrate. If your applications require your subscriptions to replicate to LOB columns in the nickname, continue to use replication under DataJoiner 2.1.1.

Sybase SQL Anywhere

The DB2 Information Integrator does not support Sybase SQL Anywhere as a data source. Therefore, replication to or from Sybase SQL Anywhere should not be migrated to DB2 Information Integrator.

Nicknames with LONG VARCHAR or LONG VARCHAR FOR BIT DATA data types greater than 32672

LONG VARCHAR and LONG VARCHAR FOR BIT DATA data types in nicknames are no longer supported. During DB2 migration, any nickname that is LONG VARCHAR data type becomes a CLOB data type, and any nickname that is a LONG VARCHAR FOR BIT DATA data type becomes a BLOB data type. If your source data is shorter than 32673 bytes, you can alter the CLOB or BLOB nickname data types to VARCHAR(32672) or VARCHAR FOR BIT DATA(32762), respectively, after you migrate DB2. However, if you use VARCHAR(32672) or VARCHAR FOR BIT DATA(32762), any data exceeding 32672 bytes will be truncated and lost during replication. To alter the data type of a column, use the ALTER statement. For example:

```
ALTER NICKNAME EMPLOYEE
      ALTER COLUMN INFO
       LOCAL TYPE VARCHAR(32672)
```

Pre-migration maintenance for the Capture program

Note: This step does not pertain to Capture triggers.

You must prepare your existing Capture program *before* you migrate your replication environment to Version 8. For Linux, UNIX, and Windows, the Capture preparation must be done before DB2 is migrated to Version 8. If you do not prepare the Capture program, you cannot successfully migrate to Version 8. Preparing the Capture program involves applying the maintenance for your particular platform, and running the Capture program for the specified period of time before you migrate. The maintenance that you must apply depends on the system that you are migrating and is described in the sections that help you prepare each server for migration.

Pre-migration maintenance for the DJRA tool (DataJoiner)

Before you begin the DataJoiner migration, it is recommended that you update the trigger definitions or stored procedures that are created by the DataJoiner Replication Administration (DJRA) tool. Instructions are described in *Updating triggers and stored procedures for the DataJoiner Replication Administration Tool*. The document is available on the web at http://www.ibm.com/software/data/dpropr/library.html. These updates are needed to take advantage of the new function provided in DJRA shipped with DataJoiner V2.1.1 PTF 12 (IP22531 for Windows or U483554 for AIX®). These updated triggers or stored procedures are used to replicate data from the following non-DB2 relational databases:

- Informix® Dynamic Server
- Microsoft SQL Server™
- Sybase® Adaptive Server Enterprise

If you do not apply this DJRA maintenance, after you migrate to Version 8 replication you will need to update the trigger definitions or stored procedures manually. DJRA cannot be used with Version 8 replication.

Important:

- If you do not update the definitions, *data might be lost* during the replication process from the source to a target.
- You must update the definitions *before* you start Version 8 replication or before you add any new Version 8 registrations for replication.

Planning to install the OS/400 V5R2 operating system (iSeries)

You must install OS/400 Version 5 Release 2 before you migrate your replication environment. Apply all necessary PTFs for DP2 or DP3 before upgrading the operating system to V5R2. When you install V5R2, the Version 8 replication programs (for example, Capture and Apply) will be on your system but they will not be usable until you perform the migration as described in this document. Your existing DataPropagator for iSeries licensed program still works and is *not* overwritten or removed when you install OS/400 V5R2. You can continue to use your existing replication programs until you migrate your replication environment to Version 8. For details, see "Preparing to migrate iSeries servers" on page 31.

Planning server migration in distributed environments

Almost all replication configurations include Capture and Apply programs running in different DB2 databases in a network. You can use different versions of Capture and Apply programs and Capture triggers in a network to replicate data, with some limitations. This flexibility allows you to migrate all of your replication environment or parts of it in stages, and work *temporarily* in a mixed-level environment.

If you cannot migrate your distributed replication environment to Version 8 simultaneously, you must ensure that you migrate your replication servers in the appropriate order. The order depends on the coexistence of the Apply program, the compatibility of DB2 clients and servers, and the DB2 database or instance migration. For details, see:

- "Apply program coexistence" on page 13

- “Pre-migration maintenance for the Apply program (Linux, UNIX, Windows, iSeries, DataJoiner)”
- “Supported down level servers and down level clients” on page 15
- “Database and instance migration (Linux, UNIX, Windows, DataJoiner)” on page 16
- “Guidelines for the order of server migration” on page 17

Apply program coexistence

On iSeries, Capture control servers and Apply control servers are migrated together on a given system. Although Capture control servers and Apply control servers cannot coexist at different replication levels on one system, they can coexist between systems. You can replicate in this temporary mixed-level environment.

On Linux, UNIX, Windows, and z/OS, you can choose to migrate a particular Capture control server or Apply control server before another and replicate in this temporary environment until you migrate the other server.

You can choose from two coexistence options:

- Use the Version 8 Apply program by migrating to DB2 Version 8 on UNIX or Windows or by installing DB2 DataPropagator for z/OS Version 8. The Version 8 Apply program can run with old or new control tables. You can install and run the Version 8 Apply program; you can migrate the Apply control server now or later. This is the only coexistence option for Apply servers on z/OS.
- Use the pre-Version 8 Apply program. Keep the existing version of your Apply control server and install the correct maintenance (see “Pre-migration maintenance for the Apply program (Linux, UNIX, Windows, iSeries, DataJoiner)”). The pre-Version 8 Apply program will work with remote Version 8 replication control tables and Capture programs *after* you apply the appropriate maintenance. No maintenance is available for z/OS servers.

Pre-migration maintenance for the Apply program (Linux, UNIX, Windows, iSeries, DataJoiner)

After you apply the most recent maintenance for the pre-Version 8 Apply program:

- The pre-Version 8 Apply program can work with the Version 8 Capture program or a Capture program from earlier releases.
- The pre-Version 8 Apply program can work with the Version 8 Apply control tables or Apply control tables from earlier releases.

This compatibility allows you to stage your migration so that you do not need to migrate all Capture control servers or Apply control servers at the same time.

If you want to run your existing Apply program to access Version 8 replication control tables, you must apply the maintenance for the Apply program that is available for the specified platform (see Table 1, Table 2, and Table 3 on page 15). If you do not intend to replicate between Version 8 programs and pre-Version 8 programs, you do not need to apply this maintenance.

Important: Replicating in an environment where different versions of replication control tables exist is meant to be temporary. Such coexistence provides flexibility during migration so that you don't have to migrate all servers to Version 8 at once. Do not replicate in such an environment for an extended period of time.

Table 1. iSeries maintenance for Apply coexistence

DB2 DataPropagator program	Maintenance for Apply
PID 5769-DP3 (Version 7)	SF67447
PID 5769-DP2 (Version 5)	SAVEFILE from Level 2

Table 2. Workstation maintenance for Apply coexistence

DB2 Program	Maintenance for Apply
DB2 for Linux (Intel), Version 7	FixPak 9 U486944
DB2 for HP V11(32–bit), Version 7	FixPak 9 U486942
DB2 for HP V11 (32& 64–bit), Version 7	FixPak 9 U486943
DB2 for Windows, Version 7	FixPak 9 WR21320
DB2 for AIX (4.2.1 or later), Version 7	FixPak 9 U486937
DB2 for AIX (4.3.3; 32&64 bit), Version 7	FixPak 9 U486938
DB2 for AIX (5L; 32&64 bit), Version 7	FixPak 9 U486939
DB2 for Linux S/390 and zSeries, Version 7	FixPak 9 MI00051
DB2 for Solaris (32–bit), Version 7	FixPak 9 U486940
DB2 for Solaris (32&64-bit), Version 7	FixPak 9 U486941
DB2 for Linux (Intel), Version 6	FixPak 11 IP22478
DB2 for UNIX, Version 6	FixPak 11 U482118
DB2 for Windows, Version 6	FixPak 11 WR21302
DB2 for AIX, Version 6	FixPak 11 U482115
DB2 for Solaris, Version 6	FixPak 11 U482116

Table 3. DataJoiner maintenance for Apply coexistence

DataJoiner program	Maintenance for Apply
DataJoiner for AIX 2.1.1	FixPak 13 or later
DataJoiner for NT 2.1.1	FixPak 13 or later
DataJoiner for Solaris 2.1.1	FixPak 8 or later

Supported down level servers and down level clients

If you migrate your Apply control servers before you migrate all of your Capture control servers or target servers, there are several restrictions and limitations (see the DB2 migration documentation for details about the restrictions and limitations). There are no such restrictions or limitations for DB2 Connect, nor with z/OS, OS/390, and iSeries database servers.

Table 4 on page 16 shows that all of the client and server combinations are supported except for the following key limitations or restrictions:

- DB2 for Linux, UNIX, and Windows Version 6 is not supported as a down level server. You must migrate Capture control servers on DB2 Version 6 immediately to DB2 Version 8. If your target server is on DB2 Version 6 and is remote to your Apply control server, you must migrate the target server immediately to DB2 Version 8.
- DataJoiner is not supported as a down level client or server. DataJoiner cannot communicate with Version 8 clients or servers. DB2 Version 8 contains federated function that replaces DataJoiner function. You must migrate DataJoiner Capture control servers or Apply control servers to DB2 Version 8 *before* or *at the same time as* you migrate Apply control servers to DB2 for UNIX or Windows.
- Several DB2 restrictions and limitations exist if you have a Version 7 server working with a Version 8 client (for details, refer to the DB2 migration documentation). The key ones are:
  - LOB or datalink columns cannot be replicated.
  - The Apply program LOADX parameter is not supported for replication because DB2 export, import, and load functions cannot run on Version 8 clients to Version 7 servers.
  - The Replication Center runs on a DB2 Version 8 database, typically as a DB2 client; therefore, it cannot connect to downlevel servers (for example, a DataJoiner database).

  To avoid the limitations, you must migrate Version 7 Capture control servers before or at the same time as you migrate Version 7 Apply control servers. Similarly, if your target server is on DB2 Version 6 and is remote to your Apply control server, you should migrate the target server to DB2 Version 8 before or at the same time as you migrate the Apply control server.

Table 4. Compatibility between DB2 clients and servers

DB2 Client (Apply control server)	DB2 Server (Capture control server or target server)							
	OS/390 V6	**OS/390 V7**	**OS/400 V5R1**	**OS/400 V5R2**	**UNIX Windows Linux V6**	**UNIX Windows Linux V7**	**UNIX Windows Linux V8**	**DataJoiner**
OS/390 Version 6	OK	OK	OK	OK	OK	OK	OK	OK
OS/390 Version 7	OK	OK	OK	OK	OK	OK	OK	OK
OS/400 V5R1	OK	OK	OK	OK	OK	OK	OK	OK
OS/400 V5R2	OK	OK	OK	OK	OK	OK	OK	OK
UNIX Windows Linux Version 6	OK	OK	OK	OK	OK	OK	OK	OK
UNIX Windows Linux Version 7	OK	OK	OK	OK	OK	OK	OK	OK
UNIX Windows Linux Version 8	OK	OK	OK	OK	NOT OK	OK with restrictions	OK	NOT OK
DataJoiner	OK	OK	OK	OK	OK	OK	NOT OK	OK

Database and instance migration (Linux, UNIX, Windows, DataJoiner)

Instance and database migration are not part of replication migration. You *must* migrate your instance and database *before* you migrate your replication migration.

When you migrate your DB2 instance to DB2 Universal Database Version 8, you get Version 8 replication programs (Capture, Apply, Replication Alert Monitor). You must migrate your Version 6 or Version 7 Capture control server as soon as possible because the Version 8 Capture program cannot access pre-Version 8 control tables. You do *not* need to migrate the Version 6 or Version 7 Apply control server immediately because the Version 8 Apply program can access Version 6 or Version 7 control tables; however, you must create a new password file and start the Apply program with the new Version 8 syntax.

After you migrate DB2 instances or databases, you do not need to migrate the Capture triggers immediately. DataJoiner Version 2.1.1 Capture triggers work with the Version 8 Apply program. Also, the Version 8 Capture triggers will work with earlier versions of the Apply program as long as you install maintenance for Apply coexistence (see "Pre-migration maintenance for the Apply program (Linux, UNIX, Windows, iSeries, DataJoiner)" on page 13).

Guidelines for the order of server migration

Generally, use these basic guidelines:

DataJoiner

Due to DB2 down level client-server compatibility restrictions, it is recommended that you first migrate DataJoiner Capture control servers or Apply control servers if they work with remote Capture control servers or Apply control servers on DB2 for UNIX, Windows, or Linux, Version 6 or Version 7.

Linux, UNIX, Windows

- Due to DB2 down level client-server compatibility restrictions, it is recommended that you migrate the Capture control server and any target servers that are remote to the Apply control servers before migrating the Apply control servers. You might need to first install maintenance of the Apply program at down level clients.
- After you install and migrate DB2 for UNIX, Windows, Linux instance to Version 8, you must migrate the DB2 database immediately. If the DB2 database is a Capture control server, you must migrate the replication Capture control server immediately.
- To avoid restrictions on certain data types and on the use of the LOADX parameter for the Apply program, migrate the Capture control server and any target servers that are remote to the Apply control servers on DB2 for UNIX, Windows, or Linux Version 7 *at the same time* or *before* you migrate the Apply control server to DB2 Version 8.

z/OS

If you have the Apply control server on z/OS and you want that Apply program to run with any pre-Version 8 Capture, you must install Version 8 Apply on that Apply control server first, before you migrate the Capture control server.

Planning replication administration during migration

The Replication Center is a new user interface tool that you can use to set up and administer your Version 8 replication environment and to run the Capture, Apply, and Replication Alert Monitor programs.[1] DB2 DataPropagator for iSeries also provides OS/400 native commands, which you can use in replication environments that are only on OS/400.

When the control tables are at an earlier version of replication, use the DB2 Control Center or the DataJoiner Replication Administration (DJRA) tool to administer replication. You cannot use the administration tools from previous replication releases to administer Version 8 replication control tables.

If you plan to migrate your servers to Version 8 in stages, and have them work temporarily with your existing pre-Version 8 servers, you cannot expect the same function as in a pure Version 8 environment. In a mixed environment, you can only replicate data. You cannot administer replication. You can use the Version 8 Analyzer program to monitor Version 8 servers, and you can use earlier versions of the Analyzer to analyze servers at earlier levels of replication.

Planning post-migration work

After a successful migration, you might need to manually update the new control tables to complete the migration. Optionally, you might also want to modify the control tables to exploit new Version 8 function. Review the lists in this section to determine if you need to perform such updates.

Also, you might need to do some system-specific post-migration work as described later in the chapter that pertains to the system.

The list of updates that is discussed in this section is not exhaustive. If you modified your pre-migration replication environment in other ways, you need to manually reapply the changes to your Version 8 environment, as appropriate. For example, if you added aliases for any of the control tables or CD tables, you must manually add those aliases after replication migration.

Manual steps to complete migration

This section describes the information that is not migrated to Version 8 control tables on Capture control servers and on Apply control servers. You might need to manually update the new control tables to complete the migration.

[1] For more information about using the Replication Center with DB2 DataPropagator for iSeries and how you can get a copy, go to http://www.ibm.com/software/data/dpropr/iseriesv8/

Information not migrated in Capture control servers

The following information is not migrated to Version 8 Capture control servers:

Triggers on CD tables

If you added triggers to replication control tables or CD tables, they will not be migrated. For example, if you use a before insert trigger on the CD table to prevent delete operations from being replicated to the target table, you must re-create the trigger on the new Version 8 CD table. It is not created for you.

Create table index

iSeries: The indexes on all Version 8 control tables will *not* be reverse engineered from the existing tables. Indexes will be created with default values used by the Replication Center. If you modified indexes in your previous replication environment, you need to review the indexes in your Version 8 tables and re-create them manually as needed.

z/OS: The indexes on CD and control tables are reverse engineered from existing tables.

Linux, UNIX, Windows: The indexes on CD and control tables are reverse engineered from existing tables. The INCLUDE clause is the only exception; it is not reverse engineered. If you have an INCLUDE clause in your previous replication environment, you need to review the indexes in your Version 8 tables and re-create them manually as needed.

Grants, synonyms, and referential constraints on replication control tables and CD tables

If you added grants, synonyms, or referential constraints on replication control tables or CD tables, they will not be migrated. If you do not want to use default grants on the Version 8 tables, you must grant privileges to the Version 8 tables after migration. (For z/OS servers, pre-migration grant information is stored in a table, BACKUP.GRANTS. That table has one VARCHAR column called GRANT.) Also, re-create any synonyms and referential constraints after migration.

Statistics gathered prior to migration (Linux, UNIX, Windows, z/OS)

The CD and the unit of work (IBMSNAP_UOW) tables are dropped and re-created during the replication migration process. Any statistics gathered prior to migration are lost. After migration, use the RUNSTATS utility to update the DB2 catalog statistics for the CD and the unit of work (IBMSNAP_UOW) tables to improve performance. Before you use the RUNSTATS utility, if you pruned the tables before migration, make sure that Version 8 Capture program runs long enough in the migrated environment so that there is a sufficient amount of data in the tables. If the statistics are not accurate for these tables and they contain a large amount of data, the performance of the Capture and Apply programs might be slow. In some cases, SQL requests issued by the Capture or Apply programs can fail because the request exceeds resource limits. For example, if the statistics are poor, the Apply program could receive an SQLCODE -905 error message while retrieving data from DB2 for Linux, UNIX, Windows. It is recommended that you also update statistics for all other replication control tables. For more information about when to use RUNSTATS in your replication environment, refer to the *DB2 Universal Database Replication Guide and Reference, SC27–1121*.

Views on control tables and CD tables

Views are re-created on replication control tables and CD tables during migration. In a few cases, the view cannot be re-created (for example, if columns don't exist in the new tables).

Information not migrated in Apply control servers

The following items will not be migrated to Version 8 Apply control servers:

Grants, synonyms, and referential constraints on replication control tables

If you added grants, synonyms, or referential constraints on replication control tables, they will not be migrated. If you do not want to use default grants on the Version 8 tables, you must grant privileges to the Version 8 tables after migration. (For z/OS servers, pre-migration grant information is stored in a table, BACKUP.GRANTS. That table has one VARCHAR column called GRANT.) Also, re-create any synonyms and referential constraints after migration.

Create table index

iSeries: The indexes in all Version 8 control tables will *not* be reverse engineered from the existing tables. Indexes will be created with default values used by the Replication Center. If you modified indexes in your previous replication environment, you need to review the indexes in your Version 8 tables and re-create them manually as needed.

z/OS: The indexes on control tables are reverse engineered from existing tables.

Linux, UNIX, Windows: The indexes on control tables are reverse engineered from existing tables. The INCLUDE clause is the only exception; it is not reverse engineered. If you have an INCLUDE clause in your previous replication environment, you need to review the indexes in your Version 8 tables and re-create them manually as needed.

Views on control tables

Views are re-created on replication control tables during migration. In a few cases, the view cannot be re-created (for example, if columns don't exist in the new tables).

LONG VARCHAR or LONG VARCHAR FOR BIT DATA source columns replicated to non-DB2 targets

LONG VARCHAR and LONG VARCHAR FOR BIT DATA data types in nicknames are no longer supported. During DB2 migration, any nickname that is LONG VARCHAR data type becomes a CLOB data type, and any nickname that is a LONG VARCHAR FOR BIT DATA data type becomes a BLOB data type. If your source data is shorter than 32673 bytes, you can alter the CLOB or BLOB nickname data types to VARCHAR(32672) or VARCHAR FOR BIT DATA(32762), respectively, after you migrate DB2. However, if you use VARCHAR(32672) or VARCHAR FOR BIT DATA(32762), any data exceeding 32672 bytes will be truncated and lost during replication. To alter the data type of a column, use the ALTER statement. For example:

```
ALTER NICKNAME EMPLOYEE
      ALTER COLUMN INFO
       LOCAL TYPE VARCHAR(32672)
```

Changing defaults to exploit new Version 8 function

Any new function that did not exist before Version 8, or was implemented differently before Version 8, will have default values assigned in Version 8 control tables that approximate the pre-Version 8 behavior. You can change the defaults by modifying the tables after migration as appropriate for your environment (for more information, see the *DB2 Universal Database Replication Guide and Reference, SC27–1121*.)

CHGONLY registration parameter for Capture

The **chgonly** function was a global startup value in previous versions, but it is set during registration in Version 8 and stored in the register control table (IBMSNAP_REGISTER). After you migrate to Version 8, the value is set to `n` by default. You must change the value, as appropriate. Setting this value to `y` ensures that the Capture program captures only changes that occur in registered columns. If every column of a table is registered, setting **chgonly** = y can increase processing time unnecessarily. If you want to see a row in the CD table for every change, use **chgonly** = n.

Modify the **chgonly** value before you start the Version 8 Capture program for the first time. You can update the value using the Replication Center through the **Properties** action on the registered object.

You can also modify the value by updating the CHGONLY column in the Register (IBMSNAP_REGISTER) table.

For example, to have chgonly active on all tables, use the following command: `update asn.ibmsnap_register set chgonly='Y'`.

To update the registration for a single source table (SVL.DEPARTMENT), use the following command:

```
update asn.ibmsnap_register set chgonly='Y'
where source_owner='SVL' and source_table = 'DEPARTMENT'
```

COMMIT_COUNT(X) column

Prior to Version 8, you could start the Apply program with the **commit(x)** startup parameter. This parameter specified that the Apply program should use transactional processing for *all* subscription sets. In Version 8, transactional processing is specified at the subscription-set level using the COMMIT_COUNT(X) column in the subscription set (IBMSNAP_SUBS_SET) table. The column indicates the type of processing that the Apply program performs for a subscription set. After migration, if a subscription set is a read-only type, the value is set to NULL. If a subscription set involves replica processing, the value is set to zero. To change the type of processing that the Apply program performs for a subscription set, modify the value from the DB2 command line or modify the column value directly for a specific subscription set.

```
update asn.ibmsnap_subs_set set commit_count = n (where
apply_qual = aq and set_name = sn and whos_on_first = d)
```

Where:

- *n* is the number of transactions between commits.
- *d* is either the ’F’ or ’S’ direction.

Subscription member predicates

If you use PREDICATES pointing to columns in CD or IBMSNAP_UOW tables, you must manually update subscription set members to take advantage of a new feature. A message is issued by the migration program to flag the existence of such predicates.

- In earlier versions of replication, you did not have the ability to do a full-refresh if the PREDICATES included a reference to either the CD or UOW column. In Version 8, full refresh is possible by placing references to CD and UOW columns in the CD_UOW_PREDICATES column of the Version 8 IBMSNAP_SUBS_MEMBR table. Leave any references to the source table columns in the PREDICATES column.
- In Version 8, the CD and UOW tables are not joined for user copy target tables (they were always joined in prior versions). If the

PREDICATES column references a UOW column, replication will fail in Version 8. For successful replication, you must modify the Version 8 IBMSNAP_SUBS_MEMBR table:

- Place references to the UOW columns in the CD_UOW_PREDICATES column (as described earlier)
- Set JOIN_UOW_CD to yes (Y)

For example, suppose that you have the following value in the existing IBMSNAP_SUBS_MEMBR table:

```
ibmsnap_subs_membr(predicates): 'ibmsnap_uowid="USER1"'
```

You must update these column values in your Version 8 table as follows:

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR SET join_uow_cd='Y',
uow_cd_predicate='ibmsnap_uowid="USER1"', predicates=null
(WHERE SOURCE_OWNER = 'MYTABLE' AND SOURCE_TABLE = 'MYTABLE')
```

RECAPTURE column

A new column in the register (IBMSNAP_REGISTER) table for update-anywhere replication. It indicates whether changes that originate from a table or view are recaptured and forwarded to other tables or views. After migration, the value for the new RECAPTURE column is set differently for source and replica tables:

- At the master, RECAPTURE is set to Y, causing changes captured at one replica to be recaptured at the master and forwarded to all replicas.
- At the replica, RECAPTURE is set to N, preventing changes that originated at the master from being recaptured unnecessarily and forwarded to other replicas.

CHG_UPD_TO_DEL_INS column

A new column in the register (IBMSNAP_REGISTER) table. It indicates how the Capture program stores updates in the CD table. This column was called PARTITION_KEYS_CHG in previous versions. After migration, the value for this column is set to the value that is in the PARTITION_KEYS_CHG column in the register table being migrated. Do not change the value unless you want to exploit the TARGET_KEY_CHG processing by the Apply program.

TARGET_KEY_CHG column

A new column in the IBMSNAP_SUBS_MEMBR table that indicates how the Apply program handles updates when changes to target key columns are replicated. After migration, the value for the new TARGET_KEY_CHG column is set to N so that when the Apply program processes update operations, it assumes that the columns that make up the target key are never updated. Change the value to Y if you want the Apply program to use the before image value to determine what row to update in the target table. You can set TARGET_KEY_CHG = ’Y’ only if the CD table contains before-image values and CHG_UPD_TO_DEL_INS = ’N’ in the register table (IBMSNAP_REGISTER) for the columns that participate in the target key.

STOP_ON_ERROR column

A new column in the register (IBMSNAP_REGISTER) table that indicates whether the Capture program will terminate or just stop processing the registration when it encounters errors in the registration.

Linux, UNIX, Windows, z/OS: The default is set to `Y`. The Capture program terminates when an error occurs while it is trying to start, initiate, reinitiate, or insert a row into the CD table.

iSeries: The default is set to `N`. The Capture program does not terminate when an error occurs while it is trying to start, initiate, reinitiate, or insert a row into the CD table. It stops processing the registration.

Optional: Removing temporary migration tables

After a successful migration, and after you are sure that you do not want to fall back to your pre-Version 8 environment, you can remove the temporary migration information to free space on your system:

- Remove the migration control tables (*bkschema*.IBMSNAP_MIGRATION, and so on).
- Remove the staged Version 8 control tables (*bkschema*.IBMSNV8_REGISTER, and so on) and CD tables.

Chapter 3. Checklists for migrating your servers

Use the following checklists to ensure that you follow the tasks that are described in detail later in this document for each operating system. Perform each step in the exact order. If a step fails, do not go to the next step. Fix the problem and retry the step.

Checklist for migrating iSeries servers

Prepare to migrate iSeries servers:

1. [ ] Prepare the existing Capture program for migration.
2. [ ] Prepare remote existing Apply programs for temporary coexistence.
3. [ ] Install OS/400 Version 5, Release 2 and install DPP 5722–DP4.
4. [ ] Run your prepared replication environment for at least the length of the retention period.
5. [ ] Run the Analyzer.
6. [ ] Prune your pre-Version 8 control tables.
7. [ ] Stop replication.
8. [ ] Determine space requirements.
9. [ ] Back up your current environment.

Perform the migration steps on iSeries servers:

1. [ ] Customize and run the QDP4/QZSNMIG8(STEP1) SQL script to prepare the migration control tables in the backup schema.
2. [ ] Run QZSNMIG8 CONDITION to get journal and library names and non-DB2 relational server names.
3. [ ] Run QZSNMIG8 BACKUP to back up your existing CD and control tables and stage the Version 8 ones.
4. [ ] Run QZSNMIG8 MIGRATION to create your Version 8 CD and control tables and to drop the pre-Version 8 ones.

Configure your Version 8 environment:

1. [ ] Manually update control tables if needed to complete migration and to exploit new Version 8 function.
2. [ ] Create SQL packages and grant privileges to the packages.
3. [ ] Start replication.

Optional: Clean up your migration environment.

© Copyright IBM Corp. 2002, 2003

Checklist for migrating z/OS servers

Prepare to migrate z/OS servers:

1. [ ] Prepare the existing Capture program for migration.
2. [ ] Determine the order in which to migrate the replication servers. Install Apply maintenance if necessary.
3. [ ] Run the Analyzer.
4. [ ] Prune your pre-Version 8 control tables.
5. [ ] Stop replication.
6. [ ] Back up your current environment.

Perform the migration steps on z/OS for *each* replication Capture control server and Apply control server:

1. [ ] Create database, storage groups, and table spaces for migration (once per subsystem).
2. [ ] Customize the ASNMIGZD script and run the ASMMIG1D sample job to set up the migration control tables in the backup schema (once per subsystem).
3. [ ] Customize and run the ASNBNDMU sample job to bind the migration program (once per subsystem).
4. Before migrating your Apply control server, if you have any iSeries sources or non-DB2 relational sources or targets:
   - [ ] Run ASNPWD from a workstation to create the new encrypted password file for ASNMIG4C.
   - [ ] Run ASNMIG4C from a workstation.
5. Run the sample job to back up the control tables for the replication server that you are migrating:
   - [ ] To backup Apply control tables on an Apply control server, customize and run ASNMIG2C.
   - [ ] To backup Capture control tables on a Capture control server, customize and run ASNMIG2S.
6. Run the sample job to migrate the control tables for the server that you are migrating:
   - [ ] To migrate an Apply control server, customize and run ASNMIG3C. The job creates your Version 8 control tables and drops the pre-Version 8 ones.
   - [ ] To migrate a Capture control server, customize and run ASNMIG3S. The job creates your Version 8 CD and control tables and drops the pre-Version 8 ones.

Configure your Version 8 environment:

1. [ ] Manually update control tables if needed to complete migration and to exploit new Version 8 function.
2. [ ] Optionally restore grants on control tables.
3. [ ] Start replication.

Optional: Clean up your migration environment.

Checklist for migrating Linux, UNIX, and Windows servers

Important: For migrating DataJoiner servers, see "Checklist for migrating DataJoiner servers" on page 28.

Prepare to migrate Linux, UNIX, and Windows servers:

1. [ ] Prepare the existing Capture program for migration.
2. [ ] Determine the order for migrating servers (install Apply maintenance)
3. [ ] Run the Analyzer.
4. [ ] Prune your pre-Version 8 control tables.
5. [ ] Stop replication.
6. [ ] Back up your current environment.
7. [ ] Migrate DB2 instances and databases to Version 8.

Perform the replication migration steps on Linux, UNIX, and Windows for each Capture control server and Apply control server:

1. [ ] Create table spaces for migration (once per database).
2. [ ] Customize and run the `sqllib\samples\repl\mig8udb.sql` script to prepare the migration control tables in the backup schema (once per database).
3. Before migrating the Apply control server, if you have any iSeries Capture control servers or non-DB2 relational sources or targets you must run ASNMIG4C.
   - [ ] Run ASNPWD to create the new encrypted password file for ASNMIG4C.
   - [ ] Run ASNMIG4C from a workstation.
4. [ ] Bind the ASNMIG8 program (once per DB2 database).
5. [ ] Run ASNMIG8 BACKUP to back up your existing CD and control tables and stage the Version 8 ones.
6. [ ] Run ASNMIG8 MIGRATION to create your Version 8 CD and control tables and to drop the pre-Version 8 ones.

Configure your Version 8 environment:

1. [ ] Manually update control tables if needed to complete migration and to exploit new Version 8 function.
2. [ ] Run ASNPWD to create the new encrypted password file for the Version 8 Apply, Monitor, and Analyzer programs.
3. [ ] Start replication.

Optional: Clean up your migration environment.

Checklist for migrating DataJoiner servers

Prepare to migrate DataJoiner servers:

1. [ ] Ensure that the Capture control triggers or stored procedures for non-DB2 relational sources are updated with the appropriate level of maintenance.
2. [ ] Determine the order for migrating servers (install Apply maintenance).
3. [ ] Run the Analyzer.
4. [ ] Prune your pre-Version 8 control tables.
5. [ ] Stop replication.
6. [ ] Back up your current environment.
7. [ ] Migrate DB2 instances and databases to Version 8.

Perform the migration steps on each federated Capture control server and Apply control server:

1. [ ] Create table spaces for migration (once per database).
2. [ ] Customize and run the `sqllib\samples\repl\mig8fed.sql` script to prepare the migration control tables in the backup schema.
3. Before migrating the Apply control server, if you have any iSeries Capture control servers or non-DB2 relational sources or targets, you must run ASNMIG4C.
   - [ ] Run ASNPWD to create the new encrypted password file for ASNMIG4C.
   - [ ] Run ASNMIG4C from a workstation.
4. [ ] If you use the Oracle NET8 wrapper to access Oracle replication sources, save a copy of the pre-Version 8 PRUNCNTL_TRIGGER definition before migrating the Capture control server.
5. [ ] Bind the ASNMIG8 program (once per DB2 database).
6. [ ] Run ASNMIG8 BACKUP to back up your existing control tables and stage the Version 8 ones.
7. [ ] Run ASNMIG8 MIGRATION to create your control tables and to drop the pre-Version 8 ones.

Configure your Version 8 environment:

1. [ ] Manually update control tables if needed to complete migration and to exploit new Version 8 function.
2. [ ] Run ASNPWD to create the new encrypted password file for the Version 8 Apply, Monitor, and Analyzer programs.
3. [ ] Start replication.
4. [ ] Migrate Oracle sources to improve performance.

Optional: Clean up your migration environment.

Chapter 4. Migrating iSeries servers to Version 8

This section describes how to migrate iSeries servers to Version 8 replication. Before you follow the instructions in this chapter, make sure that you understand the typical migration process, that you have planned your migration, and that you reviewed the checklist for migrating iSeries servers ("Checklist for migrating iSeries servers" on page 25).

Preparing to migrate iSeries servers

This section describes the prerequisites for migrating iSeries servers.

Important: Before you proceed, ensure that you have the correct level of DB2 ("Supported DB2 versions" on page 8).

Preparing the existing Capture program for migration (iSeries)

You must prepare the pre-Version 8 Capture program for migration by performing some maintenance (see Table 5). If you are using DataPropagator Version 7, you must apply a PTF. If you are using DataPropagator Version 5, you must contact IBM Software Support and restore a SAVEFILE that your service provider gives you. Applying the PTF or restoring the SAVEFILE replaces the Capture program.

Run the prepared Capture program for the length of the retention limit, or longer. The new code from the maintenance ensures stability in your environment and, over time, eliminates new orphan rows from being created in CD tables. An orphan row is a row in the CD table that has no corresponding row in the UOW table. The retention limit pruning eliminates the old or unwanted orphan rows caused by rollbacks that occurred prior to the maintenance that you applied.

Table 5. iSeries maintenance

Program	Maintenance for Capture
PID 5769-DP3 (Version 7)	SF67250 PTF and SF66877 PTF
PID 5769-DP2 (Version 5)	SF66765 PTF and a SAVEFILE from IBM Software Support

Preparing remote Apply programs for temporary coexistence (iSeries)

On iSeries servers, all of your data is migrated at once for each system. You cannot have Version 8 components working with earlier versions on the same system. If your replication environment consists of multiple systems, however, you can migrate one system at a time. If you must continue replicating before

© Copyright IBM Corp. 2002, 2003

all remote systems are migrated, you must prepare the Apply programs that you won't be migrating so that they can work with the new Version 8 control tables that you already migrated. For details, see "Pre-migration maintenance for the Apply program (Linux, UNIX, Windows, iSeries, DataJoiner)" on page 13.

Installing OS/400 Version 5, Release 2 (iSeries)

DB2 DataPropagator for iSeries Version 8 is shipped as a licensed program product of OS/400 Version 5, Release 2 (PID 5722-DP4). If you are using an earlier version of DataPropagator for iSeries, you can install Version 8 (5722-DP4) along with your upgrade to OS/400 V5R2 without affecting your existing replication environment. Your replication environment will still be at the earlier level *until* you migrate to Version 8. The Version 8 replication components are unusable until you migrate your replication components. You can migrate to Version 8 from the versions listed in Table 6.

Table 6. Associated DataPropagator and iSeries levels

DB2 DataPropagator program	iSeries or AS/400® level
PID 5769-DP3 (Version 7)	Version 5, Release 1
	Version 4, Release 5
PID 5769-DP2 (Version 5)	Version 4, Release 4 Version 1 function shipped with V4R4 is not migrated.

Apply all necessary PTFs for DP2 or DP3 before upgrading the operating system to V5R2. To ensure successful migration to DataPropagator Version 8 (PID 5722-DP4), you must install the latest PTFs for Version 8 (PID 5722-DP4) before you start the migration process. Refer to the Web site, http://www.as400service.ibm.com, for the most current information.

Any Capture and Apply control servers on an OS/400 database will be migrated together to Version 8. You cannot choose to migrate only one control server for an OS/400 system. If you have multiple systems involved in your replication environment, you can migrate them in stages to Version 8.

Running the Analyzer (iSeries)

Run the Analyzer tool (pre-Version 8). Use the resulting report to validate the data in your control tables. Determine whether there are any problem registrations or subscription sets. If there are, remove them or fix them before you migrate. If you attempt to migrate while your replication environment is not set up correctly, the migration might fail.

Pruning your pre-Version 8 control tables (iSeries)

Prune as much data as possible from your existing CD tables and Apply trail table before you migrate to Version 8. Pruning those tables minimizes the space and time required for all of the migration steps. Use Capture to prune the CD tables. Use SQL to delete rows from the Apply trail (ASN.IBMSNAP_APPLYTRAIL) table and Capture trace (ASN.IBMSNAP_TRACE) table.

Stopping replication (iSeries)

If possible, stop updating the source tables, and then run the Apply program long enough to apply all captured changes to the targets. Before you start migrating to Version 8, you must stop all local and remote Capture and Apply programs in your existing replication environment. Do not add or remove any registrations or subscription sets until migration to Version 8 is completed.

Determining space requirements (iSeries)

Before you start migrating to Version 8, ensure that you have sufficient space for the temporary tables and the new Version 8 tables that will be created:

- Double the amount of space used by your current CD tables. This space is required to hold the backup and final version of the CD tables.
- Double the amount of space used by your current replication control tables. This space is required for the backup and final version of the control tables during migration.

Backing up your current environment (iSeries)

Back up your system data on the system that you are about to migrate. It is recommended that, at a minimum, you back up the ASN library and the libraries where your CD tables are located.

Migrating your iSeries servers

To perform the migration, you will use the QDP4/QZSNMIG8(STEP1) SQL script and the **QZSNMIG8** migration program. You run these locally on the iSeries machine. After you complete the steps in this section, you will have created the Version 8 control tables and CD tables, dropped the old control tables and CD tables, enabled the Version 8 replication programs, and dropped the old replication programs.

After each step in the migration process, view the log files that are produced to verify that the step completed successfully. Proceed to the next step *only* if the current step is successful.

Using the QDP4/QZSNMIG8(STEP1) SQL script to prepare the backup schema (iSeries)

The QDP4/QZSNMIG8(STEP1) SQL script is provided to set up your migration environment. This script lets you customize the backup schema for the migration tables and create the migration control tables.

To use the QDP4/QZSNMIG8(STEP1) SQL script:

1. Customize the backup schema in the QDP4/QZSNMIG8(STEP1) SQL script.

   The default schema is BACKUP. This is the default SQL Collection where files will be created during migration. If you want to use another schema, change all occurrences of BACKUP to the new name that you want to use. Do not change the occurrences that appear in the SQL script comments.
2. Run the QDP4/QZSNMIG8(STEP1) SQL script to create the migration control tables and a staged copy of the Version 8 control tables:

   ```
   RUNSQLSTM SRCFILE(QDP4/QZSNMIG8) SRCMBR(STEP1) COMMIT(*CHG) NAMING(*SQL)
   ```
3. View the spool file to verify that the script ran successfully. The spool file is created by the CL command RUNSQLSTM, which is used to execute the statements in the script. To view the spool file, use the following command:

   ```
   DSPSPLF FILE(STEP1) SPLNBR(*LAST)
   ```

   If you get any errors or warnings, fix the errors, drop the tables in the backup schema, and run the script again.

Important: Make sure that this step completes successfully before you continue to the next step.

Running QZSNMIG8 CONDITION to get journal and library names (iSeries)

To get the journal name and library for remote source tables and non-DB2 relational source and target server names in preparation for the backup step:

1. Use the **QZSNMIG8 CONDITION** command.

   Make sure to use the backup schema that is defined in the QDP4/QZSNMIG8(STEP1) SQL script. For command syntax and usage, see “QZSNMIG8: Migration program for iSeries” on page 37. The information gathered by this command is stored in the *backupschema*.IBMSNAP.SUBS_SET table, and the information is moved into the Version 8 IBMSNAP.SUBS_SET table during a later migration step.
2. View the end of the log file for this step to verify that QZSNMIG8 ended normally:

   ```
   EDTF '/TMP/QZSNMIG8.CONDITION.LOG'
   ```

   If the step is not successful, fix the errors and try the command again.

Important: Make sure that this step completes successfully before you continue to the next step.

Running QZSNMIG8 BACKUP to back up your existing CD and control tables and stage the Version 8 ones (iSeries)

You use the QZSNMIG8 migration program to back up your control servers. The Capture control servers and Apply control servers are backed up simultaneously. The staged Version 8 control tables are populated.

To back up your existing CD and control tables and to stage the Version 8 ones:

1. Use the **QZSNMIG8 BACKUP** command.

   Make sure to use the backup schema that is defined in the QDP4/QZSNMIG8(STEP1) SQL script. For command syntax and usage, see "QZSNMIG8: Migration program for iSeries" on page 37.
2. View the log file for this step to verify that QZSNMIG8 ended normally:

   ```
   EDTF '/TMP/QZSNMIG8.BACKUP.LOG'
   ```

   If the step is not successful, fix the errors and try the command again.

Important: Make sure that this step completes successfully before you continue to the next step.

Running QZSNMIG8 MIGRATION to create your Version 8 CD and control tables and drop the pre-Version 8 ones (iSeries)

This step migrates the Capture control servers and Apply control servers to Version 8 simultaneously and drops the old servers. The migration command on an OS/400 system converts all replication control tables and CD tables to formats used by DB2 DataPropagator for iSeries Version 8. Migration also enables the Version 8 native commands, the Capture program, and the Apply program. Migration also removes the pre-Version 8 product.

To create your Version 8 CD and control tables and to drop the pre-Version 8 ones:

1. Use the **QZSNMIG8 MIGRATION** command.

   Make sure to use the backup schema that is defined in the QDP4/QZSNMIG8(STEP1) SQL script. For command syntax and usage, see "QZSNMIG8: Migration program for iSeries" on page 37.
2. View the log file for this step to verify that QZSNMIG8 ended normally:

   ```
   EDTF '/TMP/QZSNMIG8.MIGRATION.LOG'
   ```

   If the step is not successful, fix the errors and try the command again.

Important: Make sure that this step completes successfully before you start replication.

Configuring your Version 8 environment (iSeries)

This section lists the steps that you should take after you created your Version 8 CD and control tables.

Manually updating Version 8 tables and exploiting new Version 8 function (iSeries)

Manually update anything that migration could not handle from the pre-Version 8 environment (as necessary). You might also want to modify the control tables to exploit new Version 8 function. For details, see “Planning post-migration work” on page 18.

Creating SQL packages and granting privileges to the packages (iSeries)

Create SQL packages and grant privileges in the following cases:

- When using remote journaling on your source tables.
- Before using the ADDDPRSUB or ADDDPRSUBM command to add subscription sets or subscription set members.
- When the Apply program and the Replication Analyzer are operated in a distributed replication environment.

For details, see ″Setting up the Capture and Apply programs (OS/400)″ in Chapter 2: Setting up for replication of the *DB2 Replication Guide and Reference*.

Starting replication (iSeries)

Start the Version 8 Capture and Apply programs using either the new Replication Center or native OS/400 commands. In Version 8, you can start the programs in any order. For more information about operating the Version 8 Capture and Apply programs, see the *DB2 Replication Guide and Reference*.

Using fallback to restored your pre-Version 8 iSeries environment

If the **QZSNMIG8 MIGRATION** command failed, or if you want to restore your pre-Version 8 test environment, use the **QZSNMIG8 FALLBACK** command.

On iSeries platforms, the **fallback** command restores to the previous version of DataPropagator for iSeries, 5769DP2 or 5769DP3. After you use the **fallback** command, your pre-Version 8 program product is restored, including your CD tables and replication control tables. Your Version 8 product (5722–DP4) remains on the system, but you cannot use it.

Important: If you run the Capture or Apply program after migration and then use the **fallback** command, the values in the control tables after fallback might be inconsistent with the values in your source and target tables. The **fallback** command simply restores the values that were backed up at the start of migration.

To fall back to your pre-Version 8 iSeries environment:

1. Use the **QZSNMIG8 FALLBACK** command.

   Make sure to use the backup schema that is defined in the QDP4/QZSNMIG8(STEP1) SQL script. For command syntax and usage, see "QZSNMIG8: Migration program for iSeries."

2. View the log file for this step to verify that QZSNMIG8 ended normally:

   ```
   EDTF '/TMP/QZSNMIG8.FALLBACK.LOG'
   ```

Clean up your iSeries migration environment

You might want to remove the migration backup schema tables when you are sure that you do not want to fall back to your pre-Version 8 environment. For example:

```
DLTLIB bkschema
```

where *bkschema* is the backup schema used in migration. (If message CPA7025 appears, respond with 'I'.)

Also, remove the save file QDP4/QDPRLIB. For example:

```
DLTF QDP4/QDPRLIB
```

QZSNMIG8: Migration program for iSeries

Use the **QZSNMIG8** program to run four migration commands for iSeries servers.

```
►►──call QDP4/QZSNMIG8 Parm(───command──┬────────────┬──)──────────────────────►◄
                                        └─bkschema───┘
```

Table 7. QZSNMIG8 command parameter definitions for OS/400

Parameter	Definition and prompts
command	Specifies the action being performed by the migration command: **CONDITION** Gets the OS/400 journal and library name for the remote source tables and non-DB2 relational source and target server names places them in migration control tables in the BACKUP library. **BACKUP** Backs up the existing (pre-Version 8) replication control tables on both the Capture control server and the Apply control server. It also populates the staged Version 8 tables on both servers. **MIGRATION** Creates the new Version 8 control tables for both the Capture control server and the Apply control server. It populates these new tables using data from the staged Version 8 tables. It also removes the old replication tables and product. **FALLBACK** Moves data from the backup copy of the pre-Version 8 control tables and puts it into the pre-Version 8 replication control tables. It also drops the Version 8 control tables that were created during the migration.
bkschema	Specifies the schema, which is the library that contains the replication migration control tables. **BACKUP** (default) The migration control tables reside in the BACKUP library. Delimit the schema in quotation marks to preserve case, otherwise it is folded to uppercase.

Usage notes

You must run the the QDP4/QZSNMIG8(STEP1) SQL script *before* you run the **QZSNMIG8** command. If you changed the backup schema in the script, you must specify the backup schema value when you run the **QZSNMIG8** command.

Examples for QZSNMIG8

The following examples illustrate a couple ways to use the **QZSNMIG8** command.

Example 1

To migrate the existing (pre-Version 8) replication control tables on both the Capture control server and the Apply control server, assuming that the QDP4/QZSNMIG8(STEP1) SQL script was run with the default schema: BACKUP.

```
CALL QDP4/QZSNMIG8 PARM(MIGRATION)
```

Example 2

To migrate the existing (pre-Version 8) replication control tables on both the Capture control server and the Apply control server, assuming that the QDP4/QZSNMIG8(STEP1) SQL script was run with the customized schema: BKSCHEMA.

```
CALL QDP4/QZSNMIG8 PARM(MIGRATION BKSCHEMA)
```

Chapter 5. Migrating z/OS servers to Version 8

This section describes how to migrate z/OS servers to Version 8 replication. Before you follow the instructions in this chapter, make sure that you understand the typical migration process, that you have planned your migration, and that you reviewed the checklist for migrating z/OS servers ("Checklist for migrating z/OS servers" on page 26).

Preparing to migrate z/OS servers

This section describes the prerequisites for migrating z/OS servers.

Important: Before you proceed, ensure that you have the correct level of DB2 ("Supported DB2 versions" on page 8).

Preparing the existing Capture program for migration (z/OS)

You must prepare the pre-Version 8 Capture program for migration by performing some maintenance (see Table 8).

Table 8. Maintenance for the Capture program on z/OS

Program	Maintenance for Capture
DB2 DataPropagator for OS/390 Version 6	PTF UQ57528 or later
DB2 DataPropagator for OS/390 Version 7	PTF UQ57529 or later

Run the prepared Capture program and then stop the Capture program. Before you migrate, verify that asn.ibmsnap_register.cd_old_synchpoint<>NULL where GLOBAL_RECORD=Y.

Important: Do *not* use the Cancel command to cancel the Capture program. If you cancel the Capture program instead of stopping it gracefully, the IBMSNAP_WARM_START table will be empty and the value in the asn.ibmsnap_register.cd_old_synchpoint column is not updated. Additionally, cold starting the Capture program resets the value of cd_old_synchpoint to NULL. The Capture program must be restarted and run again to calculate a new value for this column before proceeding with migration.

You must also ensure that there are no orphan rows in your CD table. An orphan row is a row in the CD table that has no corresponding row in the IBMSNAP_UOW table and it is not from a transaction that has not been committed or rolled back. If there are orphan rows in your CD table, the

© Copyright IBM Corp. 2002, 2003

CD_OLD_SYNCHPOINT value in the global row will be the log sequence number (lsn) of the oldest orphan row in that source server. The Version 8 Capture program uses the CD_OLD_SYNCHPOINT value of the global row as a starting point for reading the DB2 log. If the CD_OLD_SYNCHPOINT belongs to an orphan row, the lsn is too far back in time to be a good starting point for Capture. Follow these steps to remove all orphan rows:

1. Compare the CD_OLD_SYNCHPOINT value with the IBMSNAP_SYNCHPOINT value in the global row of the IBMSNAP_REGISTER table.

   If the CD_OLD_SYNCHPOINT value is much lower, continue with the next step. Otherwise, go to "Determine the order in which to migrate servers and if necessary install Apply coexistence maintenance (z/OS)."
2. Review your CD and IBMSNAP_UOW tables to see if the rows are orphan rows. If the IBMSNAP_UOWID value of a row in the CD table is not in the IBMSNAP_UOWID column of the IBMSNAP_UOW table, and that row does not belong to an uncommitted transaction, remove the row from the CD table.
3. Start the pre-Version 8 Capture program.
4. Stop the Capture program to produce a new value for CD_OLD_SYNCHPOINT.

Determine the order in which to migrate servers and if necessary install Apply coexistence maintenance (z/OS)

If you cannot migrate your distributed replication environment to Version 8 simultaneously, you must ensure that you migrate your replication servers in the appropriate order. The Version 8 Apply program can work with old and new control tables at both Capture control servers and Apply control servers.

If you have remote pre-Version 8 Apply control servers on Windows, UNIX, DataJoiner, or iSeries that must coexist with Version 8 Capture control tables, ensure that the proper maintenance was applied for the Apply program on those servers. For details about the required maintenance, see "Pre-migration maintenance for the Apply program (Linux, UNIX, Windows, iSeries, DataJoiner)" on page 13.

Important: Replicating in an environment where different versions of replication control tables exist is meant to be temporary. Such coexistence provides flexibility during migration so that you don't have to migrate all servers to Version 8 at once. Do not replicate in such an environment for an extended period of time.

Running the Analyzer (z/OS)

Run the Analyzer tool (pre-Version 8) from a workstation. Use the resulting report to validate the data in your control tables. Determine whether there are any problem registrations or subscription sets. If there are, remove them or fix them before you migrate. If you attempt to migrate while your replication environment is not set up correctly, the migration might fail.

Pruning your pre-Version 8 control tables (z/OS)

Prune as much data as possible from your existing CD tables, UOW table and other tables before you migrate to Version 8.

- Issue the Capture prune command to prune the CD and IBMSNAP_UOW, tables.
- Use SQL to delete rows from the IBMSNAP_APPLYTRAIL table and IBMSNAP_TRACE table.

Pruning those tables minimizes the space and time required for all of the migration steps.

Stopping replication (z/OS)

If possible, stop updating the source tables, and then run the Capture and Apply programs long enough to apply all captured changes to the targets. Before you start migrating to Version 8, you must stop all local and remote Capture and Apply programs in your existing replication environment. Do not add or remove any registrations or subscription sets until migration to Version 8 is completed.

Backing up your current environment (z/OS)

Before you migrate your subsystem to Version 8 replication, it is recommended that you make an image copy of your Capture control servers and Apply control servers using the DB2 copy utility. If for any reason you must go back to the prior version of replication and fallback will not work, you can use the backup copies to recover replication.

Migrating z/OS servers

To perform migration, use the samples in the SASNSAMP dataset:

- ASNMIGZD sample script to create migration control tables in the backup schema.
- ASNMIG1D sample job to run the ASNMIGZD sample script.
- ASNBNDMU sample job to bind the migration program.
- ASNMIG2C sample job to backup Apply control tables on an Apply control server.
- ASNMIG2S sample job to backup Capture control tables on a Capture control server.
- ASNMIG3C sample job to migrate an Apply control server.
- ASNMIG3S sample job to migrate a Capture control server.

Typically, you customize the samples and run them locally on your DB2 subsystem. If you replicate with remote DataJoiner and iSeries servers, you will also run the **asnmig4c** program from a workstation.

After each step in the migration process, view the job output that is produced to verify that the step completed successfully. Proceed to the next step only if the current step is successful. After you complete the steps in this section, you will have created the Version 8 control tables.

Important: Before you continue, make sure that you have read and completed the steps in "Preparing to migrate z/OS servers" on page 41.

Customizing the ASNMIGZD script and running the ASNMIG1D sample job to create migration control tables (z/OS)

You must ensure that the storage group, databases, and table spaces exist *before* you start migrating your servers and ensure that the table spaces are large enough to hold all of the tables that are created during migration[2]. You must do these steps once for every subsystem.

A sample script, ASNMIGZD, is provided to help you create the migration control tables and the recommended table spaces and databases.

To create the migration control tables:

1. Customize the ASNMIGZD script.

   Update the storage group in the script. If the storage group doesn't exist, create it before you run the script.

   The default schema for the migration control tables is BACKUP. The user running migration must have SYSADM authority on the subsystem or else the views might be migrated with incorrect schemas. If you want to use another schema, change all occurrences of BACKUP in the script. The backup schema can be a string of 8 or fewer alphanumeric characters and it must not contain symbols or imbedded blanks. It is always folded to uppercase.

   Uncomment the sections in the script to create the recommended databases and table spaces. (The recommended table spaces and databases are shown in Table 9 on page 45.) You can use existing databases and table spaces instead of having the script create them. If you use existing databases and table spaces, make sure that you modify the script to reference the correct databases and table spaces. When you create the table spaces, include table space parameters such as STOGROUP, PRIQTY,

[2] For information about the tables that are created, see Chapter 1, "Overview of the typical migration process," on page 1.

SECQTY, BUFFERPOOL, CCSID, and SEGSIZE. Also, include a CREATE DATABASE with the appropriate parameters for your environment for the table spaces.

2. Customize and run the ASNMIG1D sample job.

   Modify the job card and DB2 subsystem name as appropriate, then run ASNMIG1D to execute the ASNMIGZD script.

Table 9. Recommended table spaces and databases for servers (z/OS)

Table space	Database	Description
BACKUPTS	BACKUPDB	The BACKUPTS table space must be large enough to hold the following tables: • Migration control tables (bkschema.ibmsnap_migration,...)[1] • Copies[2] of pre-Version 8 replication control tables (bkschema.ibmsnap_register, ...) • Copies[3] of the pre-Version 8 CD tables (bkschema.b0, ...). The bufferpool for the BACKUPTS table space must be large enough to accomodate the largest table space bufferpool that is associated with a CD table.
ROWTS	DPROPR	The ROWTS table space for Staged Version 8 control tables must be large enough to hold the following tables, for which row locking is recommended: • bkschema.ibmsnv8_register • bkschema.ibmsnv8_capschema • bkschema.ibmsnv8_pruncntl • bkschema.ibmsnv8_prune_set • bkschema.ibmsnv8_signal • bkschema.ibmsnv8_subs_set • bkschema.ibmsnv8_subs_event • bkschema.ibmsnv8_applytrail • bkschema.ibmsnv8_applytrace • bkschema.ibmsnv8_appparms The table space for Final Version 8 control tables must be large enough to hold the following tables, for which row locking is recommended: • bkschema.ibmsnap_register through bkschema.ibmsnap_appparms

Table 9. Recommended table spaces and databases for servers (z/OS) (continued)

Table space	Database	Description
PAGETS	DPROPR	The PAGETS table space for Staged Version 8 control tables must be large enough to hold the following tables, for which page locking is recommended: • bkschema.ibmsnv8_restart • bkschema.ibmsnv8_captrace • bkschema.ibmsnv8_capparms • bbkschema.ibmsnv8_capmon • bkschema.ibmsnv8_prune_lock • bkschema.ibmsnv8_subs_membr • bkschema.ibmsnv8_subs_cols • bkschema.ibmsnv8_subs_stmts • bkschema.ibmsnv8_compensate The table space for Final Version 8 control tables must be large enough to hold the following tables, for which page locking is recommended: • bkschema.ibmsnap_restart through bkschema.ibmsnap_compensate
UOWTS	DPROPR	This table space must be large enough to hold the following tables: • Staged UOW control table (bkschema.ibmsnv8_uow)[2] • Final UOW control table (asn.ibmsnap_uow)[2]

Notes:

[1] The size of the migration control tables depends on your replication environment. Items contributing to the size include the number of registrations, number of subscriptions, number of columns in registered tables, indexes and views defined on control tables and CD tables.

[2] Use the size of the existing control tables as a guide when estimating the space required for these tables.

[3] Use the size of the existing CD tables as a guide when estimating the space required for these tables.

The final Version 8 CD tables are *always* placed in the table space that contains the pre-Version 8 CD tables.

Using the ASNBNDMU sample job to bind the migration program (z/OS)

To bind the ASNMIG8 migration program:

1. Edit the sample job ASNBNDMU.

Modify the job card and DB2 subsystem name, as appropriate.

2. Run the sample job ASNBNDMU.
3. Repeat from Step 1 for each subsystem.

Using asnmig4c before migrating z/OS Apply control servers with iSeries sources or DataJoiner sources or targets

The **asnmig4c** program gathers information from the DataJoiner and iSeries servers that is needed for the Version 8 IBMSNAP_SUBS_SET table. The program connects to every Capture control server and target server found in the pre-Version 8 IBMSNAP_SUBS_SET table. It is recommended that you run **asnmig4c** from the workstation where the V8 Replication Center is installed to limit the number of database connections that need to be configured.

Before you run **asnmig4c**, you must catalog your z/OS location name. You must also set up an encrypted password file to enable the program to connect to all the remote Capture control servers and target servers and optionally, to the Apply control server.

To use **asnmig4c**:

1. From the Windows or UNIX system, catalog your z/OS location name. Use the catalog DCS database, catalog node, and catalog database commands as documented in the DB2 Universal Database Commands Reference.
2. Set up an encrypted password file.
   a. Ensure that you have DB2 Version 8 installed on the UNIX or Windows workstation where you want to create the password file.
   b. Use the **asnpwd** command to create the new password file.

      ```
      asnpwd init
      ```

      A file called `asnpwd.aut` is created. For **asnmig4c**:

      - You must use the default name for the password file: `asnpwd.aut`.
      - You must store the password file in the directory where **asnmig4c** will be run.

   c. Add entries to the `asnpwd.aut` file. Add one entry for every Capture control and target server. Optionally, add an entry for your Apply control server. For example, use the following command to add one entry for user ID (oneuser) with its password (mypwd). The user ID must have connect privilege to access the db2db database.

      ```
      asnpwd ADD ALIAS db2db ID oneuser PASSWORD mypwf
      ```

3. Run the **asnmig4c** command and redirect output to a file.

   The backup schema that you specify must match the schema that was used in the ASNMIGZD script that created the migration control tables. For **asnmig4c** command syntax and usage, see "asnmig4c: Conditioning program (z/OS)" on page 52.

```
asnmig4c db mydb on control server using schema backup
   for backup > asnmig4c.out
```

4. Review the command output file to ensure that the command completed successfully.

Running the sample job to back up existing control tables for replication servers (z/OS)

The **asnmig8 backup** command backs up the CD and control tables on Capture control servers and the control tables on Apply control servers. You must run this command once for each Capture control server and Apply control server in your replication environment. Two sample jobs are provided to run the command:

- ASNMIG2C runs the **asnmig8 backup** command to back up the Apply control tables on an Apply control server.
- ASNMIG2S runs the **asnmig8 backup** command to back up Capture control tables on a Capture control server.

To run **asnmig8 backup**:

1. Customize the ASNMIG2C or ASNMIG2S script, as appropriate.

   Modify the job card and DB2 subsystem name, as appropriate.

   Ensure that the backup schema in the script matches the schema that was used in the ASNMIGZD script that created the migration control tables. For the command syntax and usage, see "asnmig8: Migration program (z/OS)" on page 53.
2. Review the job output to ensure that the command completed successfully.

Important: Make sure that this command completes successfully before you continue to the next step.

Running the sample job to migrate existing control tables for replication servers (z/OS)

The **asnmig8 migration** command migrates the CD and control tables on Capture control servers and the control tables on Apply control servers. You must run this command once for each Capture control server and Apply control server in your replication environment. Two sample jobs are provided to run the command:

1. ASNMIG3C runs the **asnmig8 migration** command to migrate the Apply control tables on an Apply control server.
2. ASNMIG3S runs the **asnmig8 migration** command to migrate CD tables and Capture control tables on a Capture control server.

To run **asnmig8 migration**:

- Customize the ASNMIG3C or ASNMIG3S script, as appropriate.

  Modify the job card and DB2 subsystem name, as appropriate.

Ensure that the backup schema in the script matches the schema that was used in the ASNMIGZD script that created the migration control tables. For the command syntax and usage, see "asnmig8: Migration program (z/OS)" on page 53.

- Review the job output to ensure that the command completed successfully.

Important: Make sure that this command completes successfully before you start replication.

Configuring your Version 8 environment (z/OS)

This section lists the steps that you should take after you migrate your replication Capture control servers, Apply control servers, or both.

Manually updating Version 8 tables and exploiting new Version 8 function (z/OS)

Manually update anything that migration could not handle from the pre-Version 8 environment, as appropriate. You might also want to modify the control tables to exploit new Version 8 function. For details, see "Planning post-migration work" on page 18.

To create grants on the Version 8 tables after migration:

1. Select from the BACKUP.GRANTS table and output the results to a file.

   If you are running on z/OS, use SPUFI or DSNTEP2.

   If you are running from DB2 UDB for UNIX Version 8 and are connected to z/OS:

   ```
   db2 -x 'SELECT * FROM BACKUP.GRANTS' > grants.sql
   ```
2. Run the output file to create the grants on the applicable Version 8 tables.

   If you are running on z/OS, use SPUFI or DSNTEP2.

   If you are running from DB2 UDB for UNIX Version 8 and are connected to z/OS:

   ```
   db2 -tvf grants.sql
   ```

   The script creates grants on Version 8 CD tables and applicable version 8 replication control tables. The script does *not* create grants for tables that are new in Version 8, or renamed during migration to Version 8.
3. Create appropriate grants for the replication tables that were renamed during the migration to Version 8 replication. You can reuse the grant information in the BACKUP.GRANTS table for the equivalent pre-migration table:
   - IBMSNAP_CAPPARMS (IBMSNAP_CCPPARMS in BACKUP.GRANTS table)
   - IBMSNAP_CAPTRACE (IBMSNAP_TRACE in BACKUP.GRANTS table)

- IBMSNAP_RESTART (IBMSNAP_WARM_START in BACKUP.GRANTS table)

4. Create appropriate grants for the replication control tables that are new to Version 8:
   - IBMSNAP_APPENQ
   - IBMSNAP_APPLYTRACE
   - IBMSNAP_APPPARMS
   - IBMSNAP_CAPMON
   - IBMSNAP_CAPSCHEMAS
   - IBMSNAP_PRUNE_SET
   - IBMSNAP_SIGNAL

Starting replication (z/OS)

Start the Version 8 Capture and Apply programs using either the new Replication Center or the system commands for your operating system. In Version 8 you can start the programs in any order. By default, the Capture startup parameter is set to `warmsi`. This new startup parameter ensures that the Capture program always warm starts, except the first time the program is initialized. The Capture program will warm start after migration instead of switching to a cold start because you prepared the pre-Version 8 Capture program.

For more information about operating the Version 8 Capture and Apply programs, see the *DB2 Replication Guide and Reference*.

Using fallback to restore your pre-Version 8 z/OS environment

If the **asnmig8 migration** command failed, or if you want to restore your pre-Version 8 test environment, use the **asnmig8 fallback** command. The Version 8 control tables are dropped and the pre-Version 8 control tables are restored from the backup tables. You must run the command once for each Capture control server and Apply control server that you want to restore. On z/OS servers, the **fallback** command restores the previous DB2 DataPropagator for OS/390 product. DB2 DataPropagator for z/OS Version 8 remains on the system but you cannot use it until you migrate to Version 8.

You can run the command using the ASNMIGFB sample. You can use this sample to fallback both servers simultaneously or one at a time.

Important: If you run the Capture or Apply program after migration and then use the **fallback** command, the values in the control tables after fallback might be inconsistent with the values in your source and target tables. The **fallback** command simply restores the values that were backed up at the start of migration.

To fallback to your pre-Version 8 tables: To run **asnmig8 migration**:

1. Customize the ASNMIGFB sample, as appropriate.

   Modify the job card and DB2 subsystem name, as appropriate.

   Ensure that the backup schema in the script matches the schema that was used in the ASNMIGZD script that created the migration control tables. For the command syntax and usage, see "asnmig8: Migration program (z/OS)" on page 53.
2. Review the job output to ensure that the command completed successfully.
3. Restore the grants on the control tables.

   If you are running from DB2 UDB for UNIX Version 8 and are connected to z/OS:

   a. Select from the BACKUP.GRANTS table and output the results to a file.

      ```
      db2 -x 'SELECT * FROM BACKUP.GRANTS' > grants.sql
      ```
   b. Run the output file to restore the grants.

      ```
      db2 -tvf grants.sql
      ```

   If you are running on z/OS:

   a. Select from BACKUP.GRANTS table to an output file using SPUFI or DSNTEP2.
   b. Run the output file using SPUFI or DSNTEP2.

Important: Make sure that this command completes successfully before you start replication.

Clean up your z/OS migration environment

You might want to remove the migration control tables, temporary copies of the pre-Version 8 tables, and staged Version 8 tables when you're sure that you do not want to fall back to your pre-Version 8 environment. You can drop the tables in the backup schema individually, or you can drop the entire table space or database if there are no other tables in them.

Important: The final V8 tables are placed in the *same* table space as another set of staged V8 tables tables; therefore, you must remove the staged Version 8 tables individually from that table space.

To drop the migration control tables and the backup copies of the pre-Version 8 tables:

```
DROP DATABASE backupdb
```

Where *backupdb* is the database that was created in the ASNMIGZD control table script for copies of pre-Version 8 tables and migration control tables.

To drop selected tables individually from the PAGETS, ROWTS, and UOWTS table spaces:

```
DROP TABLE bkschema.IBMSNAP_xxxx
```

Where:

- *bkschema* - is the backup schema from the ASNMIGZD script
- IBMSNAP_*xxxx* is the name of the staged V8 control table (for example, REGISTER)

asnmig4c: Conditioning program (z/OS)

Use the **asnmig4c** command when migrating Apply control servers if you have iSeries Capture control servers or DataJoiner Capture control servers or target servers. Run this command from a UNIX or Windows system where you have DB2 Universal Database Version 8 installed. It is recommended that you run it from the system where the Version 8 Replication Center is located.

Before running this command, make sure that you set up an encrypted password file using the **asnpwd** command and catalog the z/OS subsystem from your workstation.

The syntax for the **asnmig4c** command is:

```
asnmig4c db dbname on control server using schema bkschema
 for backup [user userid using password]
```

Table 10. asnmig4c command parameter definitions

Parameter Value	Definition
dbname	Specifies the database where the Apply control server is located.
bkschema	Specifies the schema name of the migration control tables. The backup schema that you specify *must* match the schema that was used in the ASNMIGZD script that created the control tables. The schema is folded to uppercase.
userid	The user ID to connect to dbname[1].
password	The password for the user ID[1].

Notes:

[1] Optional. The user ID and password are required only when accessing remote databases. If you do not provide the user ID and password, the command will check the password file.

Examples for asnmig4c

The following example illustrates the **asnmig4c** command.

Example 1

To update the migration control tables for the mydb Apply control server on z/OS for all non-DB2 relational targets or sources, and iSeries sources, assuming that the ASNMIG1D script was run with the schema: myschema and the output is piped to an output files called `asnmig4c.out`:

```
asnmig4c db mydb on control server using schema myschema
   for backup > asnmig4c.out
```

asnmig8: Migration program (z/OS)

Use the **asnmig8** command to run migration commands on the z/OS server to be migrated. Use the sample jobs (ASNMIGxx) in SASNAMP to run the command.

The syntax for running the program:

```
RUN PROG (ASNMIG8) PLAN (ASNMIG8) +
PARMS('DATABASE dbname on servertype server -
using schema bkschema for command')
```

Table 11. asnmig4c command parameter definitions (z/OS)

Parameter Value	Definition
dbname	Specifies the location name where the Apply control server or Capture control server is located. To specify a local database, use a period (.).
servertype	Specifies the type of replication server being migrated: **source** The Capture Capture control server. **control** The Apply control server.
bkschema	Specifies the schema name of the migration control tables. The backup schema that you specify *must* match the schema that was used in the ASNMIGZD script. The schema is folded to uppercase.

Table 11. asnmig4c command parameter definitions (z/OS) (continued)

Parameter Value	Definition
command	Specifies the action being performed by the migration command: **backup** Backs up the existing (pre-Version 8) control tables for the specified server. It also populates the staged Version 8 tables for the specified server. **migration** Creates the new Version 8 control tables for the specified server. It populates the new tables using data from the staged Version 8 tables. It also removes the old replication tables. **fallback** Moves data from the backup copy of the pre-Version 8 control tables and puts it into the pre-Version 8 replication control tables. It also drops the Version 8 control tables that were created during the migration.

Examples for asnmig8

The following examples illustrate a few of the uses of the **asnmig8** command.

Examples for Capture control servers (z/OS)

The following examples are for DB2 z/OS Capture control servers:

```
RUN PROG (ASNMIG8) PLAN (ASNMIG8) +
PARMS('DATABASE . ON SOURCE SERVER USING SCHEMA BACKUP FOR BACKUP')
```

```
RUN PROG (ASNMIG8) PLAN (ASNMIG8) +
PARMS('DATABASE . ON SOURCE SERVER USING SCHEMA BACKUP FOR MIGRATION')
```

Examples for Apply control servers (z/OS)

The following examples are for z/OS Apply control servers, where the backup schema is `myschema`:

```
RUN PROG (ASNMIG8) PLAN (ASNMIG8) +
PARMS('DATABASE MYDB ON CONTROL SERVER USING SCHEMA MYSCHEMA FOR BACKUP')
```

```
RUN PROG (ASNMIG8) PLAN (ASNMIG8) +
PARMS('DATABASE MYDB ON CONTROL SERVER USING SCHEMA MYSCHEMA FOR MIGRATION')
```

Chapter 6. Migrating Linux, UNIX, and Windows servers to Version 8

This section describes how to migrate Linux, UNIX, and Windows servers to Version 8 replication. Before you follow the instructions in this chapter, make sure that you understand the typical migration process, that you have planned your migration, and that you reviewed the checklist for migrating Linux, UNIX, and Windows servers ("Checklist for migrating Linux, UNIX, and Windows servers" on page 27).

Important: For migrating DataJoiner servers, see "Checklist for migrating DataJoiner servers" on page 28.

Preparing to migrate Linux, UNIX, and Windows servers

This section describes the prerequisites for migrating Linux, UNIX, and Windows servers.

Important: Before you proceed, ensure that you have the correct level of DB2 ("Supported DB2 versions" on page 8).

Preparing the existing Capture program for migration (Linux, UNIX, Windows)

You must prepare the pre-Version 8 Capture program for migration by performing some maintenance (see Table 12 on page 56).

After installing the appropriate FixPak, run the prepared Capture program for a week, or longer. After that time, stop the Capture program and, before you migrate, verify that asn.ibmsnap_register.cd_old_synchpoint<>NULL where GLOBAL_RECORD=Y.

Important: Do *not* use the UNIX **KILL** command or the Windows Task Manager to cancel the Capture program. If you use the **KILL** command rather than stopping Capture, the IBMSNAP_WARM_START table will be empty and the value in the asn.ibmsnap_register.cd_old_synchpoint column will not be updated. Additionally, cold starting the Capture program resets the value of cd_old_synchpoint to NULL. The Capture program must be restarted and run again to calculate a new value for this column before proceeding with migration.

© Copyright IBM Corp. 2002, 2003

Table 12. Maintenance for the Capture program on Linux, UNIX, and Windows

DB2 Program	Maintenance for Capture
DB2 for Linux (Intel), Version 7	FixPak 4 U478691
DB2 for HP V11, Version 7	FixPak 4 U478689
DB2 for Windows, Version 7	FixPak 4 WR21270
DB2 for AIX, Version 7	FixPak 4 U478685
DB2 for Solaris, Version 7	FixPak 4 U478687
DB2 for Linux S/390 and zSeries, Version 7	FixPak 4 MI00035
DB2 for Linux, Version 6	FixPak 9 IP22300
DB2 for HP V11, Version 6	FixPak 9 U478302
DB2 for Windows, Version 6	FixPak 9 WR21261
DB2 for AIX, Version 6	FixPak 9 U478299
DB2 for Solaris, Version 6	FixPak 9 U478300

Determining the order in which to migrate servers and installing maintenance for Apply coexistence (Linux, UNIX, Windows)

If you cannot migrate your distributed replication environment to Version 8 simultaneously, you must ensure that you migrate your replication servers in the appropriate order. The order depends on the coexistence of apply, the compatibility of DB2 clients and servers, and the DB2 database or instance migration. If the server you are migrating must work temporarily with a remote pre-Version 8 Apply program on Linux, Windows, UNIX, or iSeries, you *must* install the maintenance for Apply coexistence on the remote server. For details, see "Planning server migration in distributed environments" on page 12.

Running the Analyzer (Linux, UNIX, Windows)

Run the Analyzer tool (pre-Version 8). Use the resulting report to validate the data in your control tables. Determine whether there are any problem registrations or subscription sets. If there are, remove them or fix them before you migrate. If you attempt to migrate while your replication environment is not set up correctly, the migration might fail.

Pruning your pre-Version 8 control tables (Linux, UNIX, Windows)

Prune as much data as possible from your existing CD tables, UOW table and other tables before you migrate to Version 8.

- Issue the Capture prune command to prune the CD and IBMSNAP_UOW tables.
- Use SQL to delete rows from the IBMSNAP_APPLYTRAIL table and IBMSNAP_TRACE table.

Pruning those tables minimizes the space and time required for all of the migration steps.

Stopping replication (Linux, UNIX, Windows)

If possible, stop updating the source tables, and then run the Capture and Apply programs long enough to apply all captured changes to the targets. Before you start migrating to Version 8, you must stop all local and remote Capture and Apply programs in your existing replication environment. Do not add or remove any registrations or subscription sets until migration to Version 8 is completed.

Backing up your current environment (Linux, UNIX, Windows)

Before you migrate your DB2 instances to Version 8, it is recommended that you back up your replication Capture control server and Apply control servers using the DB2 backup database command. If for any reason you must go back to the prior version of DB2, you can use the backup copies to restore replication.

Migrating DB2 instances and databases (Linux, UNIX, Windows)

You *must* migrate your instance and database *before* you migrate your replication environment.

To migrate DB2 instances and databases in preparation for replication migration:

1. Install DB2 Version 8 as described in the DB2 documentation.
2. Migrate instances and databases as described in the DB2 documentation.
3. For Capture control servers: Rename the Version 6 or Version 7 logs so that the new Capture program can access them. During DB2 migration, those log files were renamed to *.MIG.
   a. Look in the database cfg file for the `Path to log files`.
      ```
      db2 get db cfg for database
      ```
   b. Go to the directory specified in the cfg file.
   c. Rename the *.MIG files to *.LOG.

Migrating Linux, UNIX, and Windows servers

To perform migration, use the `sqllib\samples\repl\mig8udb.sql` script and the **asnmig4c** and **asnmig8** migration programs. Typically, you run these locally on your Linux, UNIX, or Windows machine, but you can run them remotely. After you complete the steps in this section, you will have created the Version 8 control tables.

After each step in the migration process, view the command output files that are produced to verify that the step completed successfully. Proceed to the next step *only* if the current step is successful.

Important: Before you continue, make sure that you have read and completed the steps in "Preparing to migrate Linux, UNIX, and Windows servers" on page 55.

Creating table spaces for migration (Linux, UNIX, Windows)

After you migrate to DB2 Version 8 and migrate your database, you must create one or more table spaces for replication migration. You must create the table spaces *before* you start migrating your replication servers and ensure that the table spaces are large enough to hold all of the tables that are created during migration[3]. You must create the table spaces once for each database, regardless of whether the database is a Capture control server, an Apply control server, or both.

Table 13 shows the recommended tables spaces. If you choose to, you can use your own naming convention for the table spaces, or create a different number of table spaces in your environment.

Table 13. Recommended table spaces for servers (Linux, UNIX, Windows)

Table space	Description
BACKUPTS	This table space must be large enough to hold the following tables: • Migration control tables (bkschema.ibmsnap_migration, ...)[1] • Copies[2] of pre-Version 8 replication control tables (bkschema.ibmsnap_register, ...) • Copies[3] of the pre-Version 8 CD tables (bkschema.b0, ...).
UOWTS	This table space must be large enough to hold the following tables: • Staged UOW control table (bkschema.ibmsnv8_uow)[2] • Final UOW control table (asn.ibmsnap_uow)[2]
OTHERTS	This table space must be large enough to hold the following tables: • Staged Version 8 control tables (bkschema.ibmsnv8_register,...), not including the staged UOW table[2] • Final Version 8 control tables (asn.ibmsnap_register, ...)[2]

[3] For information about the tables that are created, see Chapter 1, "Overview of the typical migration process," on page 1.

Table 13. Recommended table spaces for servers (Linux, UNIX, Windows) (continued)

Table space	Description

Notes:

[1] The size of the migration control tables is dependent on your replication environment. Items contributing to the size of migration control tables include the number of registrations, number of subscriptions, number of columns in registered tables, indexes defined on the control tables and CD tables, and views defined on the control tables and CD tables. It is recommended that the CREATE TABLESPACE command include an extentsize value that is small.

[2] Use the size of the existing control tables as a guide when estimating the space required for these tables.

[3] Use the size of the existing CD tables as a guide when estimating the space required for these tables.

The final Version 8 CD tables are *always* placed in the table space that contains the pre-Version 8 CD tables.

An example for creating the backupts table spaces:

```
db2 connect to database
db2 create tablespace backupts managed by database
    using (file 'c:\backupts.f1' tbssize extentsize 2)
```

Where:

- *backupts* is the name of the table space that you are creating
- *tbssize* is the size of the table space expressed in the number of pages

For details about the **create tablespace** command, refer to the DB2 SQL Reference.

Using the mig8udb.sql script to prepare the backup schema (Linux, UNIX, Windows)

The `mig8udb.sql` script is located in the `sqllib\samples\repl` directory. Use this script to customize the backup schema and table spaces for the migration tables and to create the migration control tables. You must run this script once for each database, regardless of whether the database is a Capture control server, an Apply control server, or both.

To use the `mig8udb.sql` script:

1. Copy the sample script and edit the copy.
2. Customize the backup schema and table space name, as necessary.

   The default schema is BACKUP. If you want to use another schema, change all occurrences of BACKUP to the new name that you want to use.

The backup schema can be a string of 30 or fewer alphanumeric characters and it must not contain symbols or imbedded blanks. It is always folded to uppercase.

Customize the CREATE TABLE SQL statements if you want to use table spaces other than those recommended in "Creating table spaces for migration (Linux, UNIX, Windows)" on page 58.

3. Run the `mig8udb.sql` script to create the migration control tables and a staged copy of the Version 8 control tables.
   a. Connect to the database by entering:
      ```
      db2 connect to database
      ```
      where database is the replication server being migrated.
   b. Run the script and generate a command output file by entering the following command:
      ```
      db2 -vtf mig8udb.sql -s > mig8udb.out
      ```
4. Review the output in `mig8udb.out` to ensure that all SQL statements completed successfully.

Using asnmig4c before migrating Linux, UNIX, or Windows Apply control servers with iSeries sources or non-DB2 relational sources or targets

Run the **asnmig4c** program when migrating a Linux, UNIX, or Windows Apply control server *only* if either of the following conditions apply:

- Any Capture control servers or target servers are DataJoiner database servers.
- Any Capture control servers are iSeries servers.

The **asnmig4c** program gathers information from the DataJoiner and iSeries servers that is needed for the Version 8 IBMSNAP_SUBS_SET table. The program runs at the Linux, UNIX, and Windows Apply control server and connects to every Capture control server and target server found in the pre-Version 8 IBMSNAP_SUBS_SET table.

Before you run **asnmig4c**, you must set up an encrypted password file to enable the program to connect to all the remote Capture control server and target servers and optionally, to the Apply control server.

To use the **asnmig4c** program:

1. Set up an encrypted password file.
   a. Use the **asnpwd** command to create the new password file.
      ```
      asnpwd init
      ```
      A file called `asnpwd.aut` is created. For **asnmig4c**:
      - You must use the default name for the password file: `asnpwd.aut`.
      - You must store the password file in the directory where **asnmig4c** will be run.

   b. Add entries to the asnpwd.aut file that **asnmig4c** will use to connect to Capture control servers and target servers. Add one entry for every Capture control server and target server. Optionally, add an entry for your Apply control server. For example, use the following command to add one entry for user ID (oneuser) with its password (mypwd). The user ID (oneuser) must have connect authority to the db2db database.

      ```
      asnpwd ADD ALIAS db2db ID oneuser PASSWORD mypwf
      ```

2. Run the **asnmig4c** command and redirect output to a file.

   The backup schema that you specify *must* match the schema that was used in the mig8udb.sql script that created the migration control tables. For **asnmig4c** command syntax and usage, see "asnmig4c: Conditioning program (Linux, UNIX, Windows)" on page 65.

   ```
   asnmig4c db mydb on control server using schema backup
      for backup > asnmig4c.out
   ```

3. Review the output in asnmig4c.out to ensure that the command completed successfully.

Binding the asnmig8 program (Linux, UNIX, Windows)

To bind the **asnmig8** program:

1. Change to the directory where the migration bind files are located:

   Windows:
   drive:\sqllib\bnd

   Where *drive* is the DB2 install directory.

   UNIX: *db2homedir*/sqllib/bnd

   Where *db2homedir* is the DB2 instance home directory.

2. For each Capture control server and control server, do the following steps:

   a. Connect to the database by entering:

      ```
      db2 connect to database
      ```

      where database is the server.

   b. Create and bind the migration program package to the database by entering the following commands:

      ```
      db2 bind asnmig8.bnd
      ```

Running asnmig8 backup to back up your existing CD and control tables and stage the Version 8 ones (Linux, UNIX, Windows)

Use the **asnmig8 backup** command to back up the CD and control tables on Apply control servers and control tables on Capture control servers. You must run this command once for each Capture control server and Apply control server in your replication environment.

To back up your existing CD and control tables and to stage the Version 8 ones:

1. Run the **asnmig8 backup** command.

   The backup schema that you specify *must* match the schema that was used in the `mig8udb.sql` script that created the migration control tables. For command syntax and usage, see "asnmig8: Migration program (Linux, UNIX, Windows)" on page 66.

   For example, to back up the Apply control server:

   ```
   asnmig8 db mydb on control server using schema backup
      for backup > asnmig8.bck
   ```
2. Review the output in `asnmig8.bck` to ensure that the SQL statements completed successfully.

Important: Make sure that this command completes successfully before you continue to the next step. If **asnmig8** stops because you did not prepare the Capture program, you must follow the instructions in "Workaround if pre-Version 8 Capture program was not prepared before migration (Linux, UNIX, Windows)" on page 63 before retrying the backup step.

Running asnmig8 migration to create your Version 8 CD and control tables and drop the pre-Version 8 ones (Linux, UNIX, Windows)

Use the **asnmig8 migration** command to migrate your servers to Version 8. You must run this command once for each Capture control server and Apply control server in your replication environment.

To create your Version 8 CD and control tables and to drop the pre-Version 8 ones:

1. Run the **asnmig8 migration** command.

   The backup schema that you specify *must* match the schema that was used in the `mig8udb.sql` script that created the migration control tables. For command syntax and usage, see "asnmig8: Migration program (Linux, UNIX, Windows)" on page 66.

   For example, to migrate the Apply control server:

   ```
   asnmig8 db mydb on control server using schema backup
      for migration > asnmig8.mig
   ```
2. Review the output in `asnmig8.mig` to ensure that the SQL statements completed successfully.

Important: Make sure that this command completes successfully before you start replication.

Configuring your Version 8 environment (Linux, UNIX, Windows)

This section lists the steps that you should take after you migrate your replication Capture control servers, Apply control servers, or both.

Manually updating Version 8 tables and exploiting new Version 8 function (Linux, UNIX, Windows)

Manually update anything that migration could not handle from the pre-Version 8 environment (as necessary). You might also want to modify the control tables to exploit new Version 8 function. For details, see "Planning post-migration work" on page 18.

Creating a password file for the Version 8 replication programs

Ensure that you have a replication password file set up that contains user ID and password combinations that can be used by the Apply program, Analyzer, and Replication Alert Monitor. You can update the `asnpwd.aut` file that you created for **asnmig4c** or you can create a new password file using the **asnpwd** command.

Starting replication (Linux, UNIX, Windows)

Start the Version 8 Capture and Apply programs using either the new Replication Center or the system commands for your operating system.

By default, the Capture startup parameter is set to `warmsi`. This new startup parameter ensures that the Capture program always warm starts, except the first time the program is initialized. After successful migration, the program will warm start instead of switching to a cold start because you migrated the control tables. (You must cold start the Version 8 Capture program if the pre-Version 8 Capture program was not caught up with the changes in the source before you started migration.)

For more information about operating the Version 8 Capture and Apply programs, see the *DB2 Replication Guide and Reference*.

Important: You can start replication after you migrate your Capture control server even if you did not migrate the Apply control server. You can run the old Apply program with Version 8 command syntax and the Version 8 encrypted password file to access both Version 8 and pre-Version 8 tables as long as you installed the maintenance for the Apply program.

Workaround if pre-Version 8 Capture program was not prepared before migration (Linux, UNIX, Windows)

If you did not prepare the Capture program before migrating your Capture control servers on Linux, UNIX, or Windows, you must manually update tables so that the **asnmig8 migration** program can run.

To manually complete the migration for Linux, UNIX, and Windows Capture control servers:

1. Determine if the pre-Version 8 Capture program has replicated all changes from the source tables.

If Capture program has replicated all changes:

a. Manually update the global row of the pre-Version 8 IBMSNAP_REGISTER table:

```
UPDATE ASN.IBMSNAP_REGISTER SET CD_OLD_SYNCHPOINT=SYNCHPOINT
WHERE GLOBAL_RECORD='Y'
```

b. Retry the migration steps, starting with "Running asnmig8 backup to back up your existing CD and control tables and stage the Version 8 ones (Linux, UNIX, Windows)" on page 61. When migration is complete, you will be able to warm start the Version 8 Capture program.

If Capture program has *not* replicated all changes:

a. Manually update the global row of the pre-Version 8 IBMSNAP_REGISTER table to any non-null value.

b. Retry the migration steps, starting with "Running asnmig8 backup to back up your existing CD and control tables and stage the Version 8 ones (Linux, UNIX, Windows)" on page 61. When migration is complete, you will have to cold start the Version 8 Capture program to ensure that data was not lost.

Using the fallback command to restore your pre-Version 8 Linux, UNIX, or Windows environment

If the **asnmig8 migration** command failed, or if you want to restore your pre-Version 8 test environment, use the **asnmig8 fallback** command. You must run the command once for each Apply control server that you want to restore. The **fallback** command is not supported for Linux, UNIX, or Windows Capture control servers. The **fallback** command drops the Version 8 control tables and re-creates the pre-Version 8 control tables from the backup tables.

Important: If you run the Capture or Apply program after migration and then use the **fallback** command, the values in the control tables after falling back might be inconsistent with the values in your source and target tables. The **fallback** command simply restores the values that were backed up at the start of migration.

To fall back to your pre-Version 8 tables:

1. Run the **asnmig8 fallback** command.

   The backup schema that you specify *must* match the schema that was used in the `mig8udb.sql` script that created the migration control tables. For command syntax and usage, see "asnmig8: Migration program (Linux, UNIX, Windows)" on page 66.
2. Review the output in the command output file to ensure that the SQL statements completed successfully.

Clean up your Linux, UNIX, or Windows migration environment

You might want to remove the migration control tables, temporary copies of the pre-Version 8 tables, and staged Version 8 tables when you're sure that you do not want to fall back to your pre-Version 8 environment. You can drop the tables in the backup schema individually, or you can drop the entire table space if there are no other tables in them.

Important: The final V8 tables are placed in the *same* table space as another set of staged V8 tables tables; therefore, you must remove the staged Version 8 tables individually from that table space.

To drop the migration control tables and the backup copies of the pre-Version 8 tables:

```
DROP TABLESPACE backupts
```

Where *backupts* is the tablespace that was created in the `mig8udb.sql` script for copies of pre-Version 8 tables and migration control tables.

To drop selected tables individually from the OTHERTS and UOWTS table spaces:

```
DROP TABLE bkschema.IBMSNAP_xxxx
```

Where:

- *bkschema* - is the backup schema from the `mig8udb.sql` script
- IBMSNAP_*xxxx* is the name of the staged V8 control table (for example, REGISTER)

asnmig4c: Conditioning program (Linux, UNIX, Windows)

Run the **asnmig4c** command from Apply control servers on Linux, UNIX, or Windows if you have iSeries sources or non-DB2 relational sources or targets.

Before running this command, make sure that you set up an encrypted password file using the **asnpwd** command.

The syntax for the **asnmig4c** command is:

```
asnmig4c db dbname on control server using schema bkschema
 for backup [user userid using password]
```

Table 14. asnmig4c command parameter definitions (Linux, UNIX, Windows)

Parameter Value	Definition
dbname	Specifies the database where the Apply control server is located.

Table 14. asnmig4c command parameter definitions (Linux, UNIX, Windows) (continued)

Parameter Value	Definition
bkschema	Specifies the schema name of the migration control tables. The backup schema that you specify *must* match the schema that was used in the `mig8udb.sql` script that created the control tables. The schema is folded to uppercase.
userid	The user ID to connect to dbname[1].
password	The password for the user ID[1].

Notes:

[1] Optional. The user ID and password are required only when accessing remote databases. If you do not provide the user ID and password, the command will check the password file.

Examples for asnmig4c

The following example illustrates the **asnmig4c** command.

Example 1

To update the migration control tables for the `mydb` Apply control server for all non-DB2 relational targets or sources, and iSeries sources, assuming that the `mig8udb.sql` script was run with the schema: `myschema` and the output is piped to an output files called `asnmig4c.out`:

```
asnmig4c db mydb on control server using schema myschema
   for backup > asnmig4c.out
```

asnmig8: Migration program (Linux, UNIX, Windows)

Use the **asnmig8** command to run migration commands for Linux, UNIX, and Windows servers.

```
asnmig8 db dbname on servertype server
using  schema bkschema for command
[user userid using password]
```

Table 15. asnmig8 command parameter definitions (Linux, UNIX, Windows)

Parameter Value	Definition
dbname	Specifies the database where the Capture control server or Apply control server is located.
servertype	Specifies the type of replication server being migrated: **source** The Capture control server. **control** The Apply control server.

Table 15. asnmig8 command parameter definitions (Linux, UNIX, Windows) (continued)

Parameter Value	Definition
bkschema	Specifies the schema name of the migration control tables. The backup schema that you specify *must* match the schema that was used in the `mig8udb.sql` script. The schema is folded to uppercase.
command	Specifies the action being performed by the migration command: **backup** Backs up the existing (pre-Version 8) control tables for the specified server. It also populates the staged Version 8 tables for the specified server. **migration** Creates the new Version 8 control tables for the specified server. It populates the new tables using data from the staged Version 8 tables. It also removes the old replication tables. **fallback**[2] Moves data from the backup copy of the pre-Version 8 control tables and puts it into the pre-Version 8 replication control tables. It also drops the Version 8 control tables that were created during the migration.
userid	The user ID to connect to *dbname*[1].
password	The password for the user ID[1].

Notes:

[1] Optional. The user ID and password are required only when accessing remote databases.

[2] The **fallback** command cannot be run on UNIX or Windows Capture control servers.

Examples for asnmig8

The following examples illustrate a few of the uses of the **asnmig8** command.

Examples for Capture control servers (Linux, UNIX, Windows)

The following examples are for DB2 Linux, UNIX, and Windows Capture control servers and redirect output to a file. They assume that the `mig8udb.sql` script was run with the default backup schema (`backup`) specified.

```
asnmig8 db mydb on source server using schema backup
   for backup > bksrc.out
```

```
asnmig8 db mydb on source server using schema backup
   for migration > migsrc.out
```

Examples for Apply control servers (Linux, UNIX, Windows)

The following examples are for Linux, UNIX, and Windows Apply control servers, where the backup schema is myschema and the output is directed to files:

```
asnmig8 db otherdb on control server using schema myschema
   for backup > bkctl.out
```

```
asnmig8 db otherdb on control server using schema myschema
   for migration > migctl.out
```

Chapter 7. Migrating DataJoiner servers to Version 8

This section describes how to migrate DataJoiner Capture control servers and Apply control servers to Version 8 replication. Before you follow the instructions in this chapter, make sure that you understand the typical migration process, that you have planned your migration, and that you reviewed the checklist for migrating DataJoiner servers ("Checklist for migrating DataJoiner servers" on page 28).

Preparing to migrate replication servers on DataJoiner

This section describes the prerequisites for migrating DataJoiner replication servers.

Important: Before you proceed, ensure that you have the correct level of DataJoiner ("Supported DB2 versions" on page 8).

Applying maintenance for the DJRA tool (DataJoiner)

Before you begin the DataJoiner migration, make sure that you updated the trigger definitions or stored procedures that are created by the DataJoiner Replication Administration (DJRA) tool as described in *Updating triggers and stored procedures for the DataJoiner Replication Administration Tool*. The document is available on the web at http://www.ibm.com/software/data/dpropr/library.html.

If you do not apply this DJRA maintenance, after you migrate to Version 8 replication you will need to update the trigger definitions or stored procedures manually. DJRA cannot be used with Version 8 replication.

Determine the order in which to migrate servers and if necessary install Apply coexistence maintenance (DataJoiner)

If you cannot migrate your distributed replication environment to Version 8 simultaneously, you must ensure that you migrate your replication servers in the appropriate order. The order depends on the coexistence of apply, the compatibility of DB2 clients and servers, and the DB2 database or instance migration (for details, see "Planning server migration in distributed environments" on page 12).

Running the Analyzer (DataJoiner)

Run the Analyzer tool (pre-Version 8). Use the resulting report to validate the data in your control tables. Determine whether there are any problem registrations or subscription sets. If there are, remove them or fix them before

© Copyright IBM Corp. 2002, 2003

you migrate. If you attempt to migrate while your replication environment is not set up correctly, the migration might fail.

Pruning your pre-Version 8 control tables (DataJoiner)

Use SQL to delete rows from the IBMSNAP_APPLYTRAIL table. Pruning this table minimizes the space and time required for all of the migration steps.

Stopping replication (DataJoiner)

If possible, stop updating the source tables, and then run the Apply program long enough to apply all captured changes to the targets. Before you start migrating a DataJoiner server to Version 8, you must stop all local and remote Apply programs. Do not add or remove any registrations or subscription sets until migration to Version 8 is completed.

Backing up your current environment (DataJoiner)

Before you migrate your DB2 instances to Version 8, it is recommended that you back up your replication Capture control servers and Apply control servers using the DB2 backup database command. It is recommended that you also back up the non-DB2 Capture control server because it contains replication control tables. If for any reason you must go back to the prior version of DataJoiner, you can use the backup copies to restore replication.

Migrating DB2 instances and databases (DataJoiner)

You *must* migrate your instance and database *before* you migrate your replication environment.

To migrate DB2 instances and databases in preparation for replication migration:

1. Install DB2 Version 8 as described in the DB2 documentation.
2. Migrate instances and databases as described in the DB2 documentation.

Important: If you use DataJoiner Version 2.1.1, you must migrate to DB2 Information Integrator Version 8. For current information about migrating DataJoiner 2.1.1 to DB2 Information Integrator Version 8, refer to the DB2 Information Integrator migration information and to the DB2 Information Integrator support page: http://www.ibm.com/software/data/integration/db2ii/support.html.

Migration steps for DataJoiner servers

To perform migration, use the `sqllib\samples\repl\mig8fed.sql` script and the **asnmig4c** and **asnmig8** migration programs. Typically, you run these locally on your UNIX or Windows machine, but you can run them remotely. After you complete the steps in this section, you will have created the Version 8 control tables.

After each step in the migration process, view the command output files that are produced to verify that the step completed successfully. Proceed to the next step *only* if the current step is successful.

Important: Before you continue, make sure that you have read and completed the steps in "Preparing to migrate replication servers on DataJoiner" on page 69.

Creating table spaces for migration (DataJoiner)

After you migrate to DB2 Version 8 and migrate your database, you must create one or more table spaces for replication migration. You must create table spaces *before* you start migrating your replication servers and ensure that the table spaces are large enough to hold all of the tables that are created during migration[4]. You must create the table spaces once for each database, regardless of whether the database is a Capture control server, an Apply control server, or both.

Table 16 shows the recommended tables spaces. If you choose to, you can use your own naming convention for the table spaces, or create a different number of table spaces in your environment.

Table 16. Recommended table spaces for servers (DataJoiner)

Table space	Description
BACKUPTS	This table space must be large enough to hold the following tables: • Migration control tables (bkschema.ibmsnap_migration ,...)[1] • Copies[2] of pre-Version 8 replication control tables (bkschema.ibmsnap_register, ...)
OTHERTS	This table space must be large enough to hold the following tables: • Staged Version 8 control tables (bkschema.ibmsnv8_register, ...)[2] • Final Version 8 control tables (asn.ibmsnap_capschemas)

Notes:

[1]The size of the migration control tables is dependent on your replication environment. Items contributing to the size of migration control tables include the number of registrations, number of subscriptions, number of columns in registered tables, indexes defined on the control tables, and views defined on the control tables. It is recommended that the CREATE TABLESPACE command include an extentsize value that is small.

[2]Use the size of the existing control tables as a guide when estimating the space required for these tables.

[4] For information about the tables that are created, see Chapter 1, "Overview of the typical migration process," on page 1.

The CCD tables are not changed for Version 8.

An example for creating table spaces:

```
db2 connect to database
db2 create tablespace backupts managed by database
    using (file 'c:\backupts.f1' tbssize extentsize 2)
```

Where:

- *backupts* is the name of the table space that you are creating
- *tbssize* is the size of the table space expressed in the number of pages

For details about the **create tablespace** command, refer to the DB2 SQL Reference.

Using the mig8fed.sql script to prepare the backup schema (DataJoiner)

The mig8fed.sql script is located in the sqllib\samples\repl directory. Use this script to customize the backup schema and table spaces for the migration tables and to create the migration control tables. You must run this script once for each database, regardless of whether the database is a Capture control server, an Apply control server, or both.

To use the mig8fed.sql script:

1. Copy the sample script and edit the copy.
2. Customize the backup schema and table space name, as necessary.

   The default schema is BACKUP. If you want to use another schema, change all occurrences of BACKUP to the new name that you want to use. Do *not* change the occurrences that appear in the SQL script comments. The backup schema can be a string of 30 or fewer alphanumeric characters and it must not contain symbols or imbedded blanks. It is always folded to uppercase.

   Customize the CREATE TABLE SQL statements if you want to use table spaces other than those recommended in "Creating table spaces for migration (DataJoiner)" on page 71.
3. Run the mig8fed.sql script to create the migration control tables and a staged copy of the Version 8 control tables.
   a. Connect to the database by entering:

      ```
      db2 connect to database
      ```

      where database is the server that you are migrating.
   b. Run the script and generate a command output file by entering the following command:

      ```
      db2 -vtf mig8fed.sql > mig8fed.out
      ```
4. Review the output in mig8fed.out to ensure that the SQL statements completed successfully.

Using asnmig4c before migrating DataJoiner Apply control servers with iSeries sources or non-DB2 relational sources or targets

Use the **asnmig4c** program when migrating a DataJoiner Apply control server *only* if either of the following conditions apply:

- Any Capture control servers or target servers are DataJoiner database servers.
- Any Capture control servers are iSeries servers.

The **asnmig4c** program gathers information from the DataJoiner and iSeries servers journal information that is missing in the existing IBMSNAP_SUBS_SET table but that is required for Version 8. The program runs at the Apply control server and connects to every Capture control server and target server found in the pre-Version 8 IBMSNAP_SUBS_SET table. It uses the new encrypted password file to connect to remote servers.

Before you run **asnmig4c**, you must set up an encrypted password file to enable the program to connect to all the remote Capture control server and target servers and optionally, to the Apply control server.

To use the **asnmig4c** program:

1. Set up an encrypted password file.
   a. Use the **asnpwd** command to create the new password file.
      ```
      asnpwd init
      ```

      A file called asnpwd.aut is created. For **asnmig4c**:
      - You must use the default name for the password file: asnpwd.aut.
      - You must store the password file in the directory where **asnmig4c** will be run.
   b. Add entries to the asnpwd.aut file. Add one entry for every Capture control server and target server. Optionally, add an entry for your Apply control server. For example, use the following command to add one entry for user ID (oneuser) with its password (mypwd). The user ID (oneuser) must have connect authority to the database (db2db).
      ```
      asnpwd ADD ALIAS db2db ID oneuser PASSWORD mypwf
      ```
2. Run the **asnmig4c** command and redirect output to a file.

   The backup schema that you specify *must* match the schema that was used in the mig8fed.sql script that created the migration control tables. For **asnmig4c** command syntax and usage, see “asnmig4c: Conditioning program (DataJoiner)” on page 80.
   ```
   asnmig4c db mydb on control server using schema backup
      for backup > asnmig4c.out
   ```
3. Review the output in asnmig4c.out to ensure that the SQL statements completed successfully.

Saving a copy of the pre-Version 8 PRUNCNTL_TRIGGER

If you use the Oracle NET8 wrapper to access Oracle replication sources, you must save a copy of the pre-Version 8 PRUNCNTL_TRIGGER definition that is created on Oracle before performing replication migration. This definition is needed in case you run the **fallback** command later and you need to restore the definition manually. Replication migration cannot read the trigger definition stored in Oracle using the Oracle NET8 wrapper, so you need to save the trigger definition by connecting natively to Oracle or by using the Oracle SQLNET wrapper with DB2 V8 to connect to Oracle.

After connecting to Oracle, use the following SQL statements to retrieve the trigger body definition:

```
SELECT owner, table_owner, trigger_body FROM all_triggers
  WHERE trigger_name='PRUNCNTL_TRIGGER';
```

Build the complete trigger definition by substituting the results of the select statement for *owner*, *table_owner*, and *trigger_body* to the following SQL statement:

```
CREATE TRIGGER owner."PRUNCNTL_TRIGGER"
  ON table_owner."IBMSNAP_PRUNCNTL"
  FOR UPDATE AS trigger_body;
```

Save the create trigger statement to a file. For example, `preV8pruncntl_trigger.sql`.

Binding the asnmig8 program (DataJoiner)

To bind the **asnmig8** program:

1. Change to the directory where the migration bind files are located:

   Windows:
   drive:`\sqllib\bnd`

   Where *drive* is the DB2 install directory.

   UNIX: *db2homedir*`/sqllib/bnd`

   Where *db2homedir* is the DB2 instance home directory.

2. For each Capture control server and Apply control server, do the following steps:

   a. Connect to the database by entering:

      ```
      db2 connect to database
      ```

      where database is the server that you are migrating.

   b. Create and bind the migration program package to the database by entering the following commands:

      ```
      db2 bind asnmig8.bnd
      ```

Running asnmig8 backup to back up your existing control tables and stage the Version 8 ones (DataJoiner)

Use the **asnmig8 backup** command to back up the control tables on Apply control servers and on Capture control servers. You must run this command once for each Capture control server and Apply control server in your replication environment.

To back up your existing control tables and to stage the Version 8 ones:

1. Run the **asnmig8 backup** command.

   The backup schema that you specify *must* match the schema that was used in the `mig8fed.sql` script that created the migration control tables. For command syntax and usage, see "asnmig8: Migration program (DataJoiner)" on page 81.

   For example, to back up the Apply control server:

   ```
   asnmig8 db mydb on control server using schema backup
      for backup > asnmig8.bck
   ```

2. Review the output in `asnmig8.bck` to ensure that the command completed successfully.

Important: Make sure that this command completes successfully before you continue to the next step.

Running asnmig8 migration to create your Version 8 control tables and drop the pre-Version 8 ones (DataJoiner)

Use the **asnmig8 migration** command to migrate your servers to Version 8. It also removes the pre-Version 8 programs. You must run this command once for each Capture control server and Apply control server in your replication environment. This command uses the new encrypted password file to connect to remote servers.

To create your Version 8 control tables and to drop the pre-Version 8 ones:

1. Run the **asnmig8 migration** command.

   The backup schema that you specify *must* match the schema that was used in the `mig8fed.sql` script that created the migration control tables. For command syntax and usage, see "asnmig8: Migration program (DataJoiner)" on page 81.

   For example, to migrate the Apply control server:

   ```
   asnmig8 db mydb on control server using schema backup
      for migration > asnmig8.mig
   ```

2. Review the output in `asnmig8.mig` to ensure that the SQL statements completed successfully.

Important: Make sure that this command completes successfully before you start replication.

Configuring your Version 8 environment (DataJoiner)

This section lists the steps that you should take after you migrate your replication Capture control servers, Apply control servers, or both.

Manually updating Version 8 tables and exploiting new Version 8 function (DataJoiner)

Manually update anything that migration could not handle from the pre-Version 8 environment (as necessary). You might also want to modify the control tables to exploit new Version 8 function. For details, see “Planning post-migration work” on page 18.

Creating a password file for the Version 8 replication programs (DataJoiner)

Ensure that you have a replication password file set up that contains user ID and password combinations that can be used by the Apply program, Analyzer, and Replication Alert Monitor. You can update the `asnpwd.aut` file that you created for **asnmig4c** or you can create a new password file using the **asnpwd** command.

Starting replication (DataJoiner)

Start the Version 8 Apply programs using either the new Replication Center or the system commands for your operating system. For more information about operating the Version 8 Apply program, see the *DB2 Replication Guide and Reference*.

Important: You can start replication after you migrate your replication Capture control server even if you did not migrate the Apply control server. You can run the old Apply program with Version 8 command syntax and the Version 8 encrypted password file to access both Version 8 and pre-Version 8 tables as long as you installed the maintenance for the Apply program.

Using fallback to restore your pre-Version 8 DataJoiner environment

If the **asnmig8 migration** command failed, or if you want to restore your pre-Version 8 test environment, use the **asnmig8 fallback** command. You must run the **fallback** command once for each Capture control server and Apply control server that you want to restore. Fallback involves dropping the Version 8 control tables and re-creating the pre-Version 8 control tables from the backup tables.

Important: If you replicate data after migration and then use the **fallback** command, the values in the control tables after fallback might be inconsistent with the values in your Capture control server and target tables. The **fallback** command simply restores the values that were backed up at the start of migration.

To fallback to your pre-Version 8 tables:

1. Run the **asnmig8 fallback** command.

   The backup schema that you specify *must* match the schema that was used in the `mig8fed.sql` script that created the migration control tables. For command syntax and usage, see "asnmig8: Migration program (DataJoiner)" on page 81.
2. Review the output in the command output file to ensure that the command completed successfully.
3. If you use the Oracle NET8 wrapper to access Oracle replication sources, re-create the PRUNCNTL_TRIGGER to pre-Version 8 level by using the PRUNCNTL_TRIGGER definition you that saved earlier (see "Saving a copy of the pre-Version 8 PRUNCNTL_TRIGGER" on page 74.)
4. Connect natively to Oracle and re-create the PRUNCNTL_TRIGGER.

Migrating Oracle sources for improved performance

In Version 8.1.4, the Apply program no longer needs to issue lock table statements for CCD tables on Oracle sources. To take advantage of this improvement, you must migrate any existing registrations and subscriptions for Oracle sources.

Prerequisites:

1. Migrate your DataJoiner server to Information Integrator Version 8.1 as described in Chapter 7, "Migrating DataJoiner servers to Version 8," on page 69.
2. Install Information Integrator Version 8.1 FixPak 4.

To migrate Oracle servers for improved performance:

1. Run the Analyzer tool and use the report as a guide for dropping and recreating your registrations and subscriptions.
2. Ensure that the Apply program applied all changes to the target and that no rows exist in the CCD tables. If all changes are not applied, you must perform a full refresh of the target tables after these steps.
3. Stop the Apply programs.
4. Stop all the source applications on the Oracle source server.
5. Create a new sequence generator.
   a. Create a file with the following contents:

      ```
      SET PASSTHRU  "server_name"#

      CREATE SEQUENCE "remote_authid"."SGENERATOR002"
      MINVALUE 100 INCREMENT BY 1#
      COMMIT#
      SET PASSTHRU RESET#
      ```

b. Edit the file to update the *server_name* and *remote_authid* with the exact values that are stored in the SYSIBM catalog tables.

   - *server_name* is the server name that was used when you created the server mapping to Oracle. You can find the server name by selecting the SERVERNAME from the SYSIBM.SYSSERVERS table.
   - *remote_authid* is the remote authorization ID that was used when you created the user mapping to the *server_name*. You can find the remote authorization ID by selecting the SETTING from the SYSIBM.SYSUSEROPTIONS table where OPTION = 'REMOTE_AUTHID' and SERVERNAME = '*server_name*'.

c. Run the file to create the new sequence generator, SGENERATOR002.

   ```
   db2 -td# -vf filename
   ```

6. Drop and re-create the REG_SYNCH_TRIGGER.

   a. Create a file with the following contents:

   ```
   SET PASSTHRU  "server_name"#
   DROP TRIGGER "remote_authid"."REG_SYNCH_TRIGGER"#
   CREATE TRIGGER "remote_authid"."REG_SYNCH_TRIGGER"
   AFTER UPDATE ON "remote_authid"."IBMSNAP_REG_SYNCH"
   DECLARE
   HOLD_ME RAW(10);
   BEGIN
   SELECT
      LPAD(TO_CHAR("remote_authid"."SGENERATOR001".NEXTVAL), 20, '0')
         INTO HOLD_ME FROM DUAL;
   UPDATE "remote_authid"."IBMSNAP_REGISTER"
   SET SYNCHPOINT= HOLD_ME,
   SYNCHTIME=SYSDATE;
   END;#

   COMMIT#

   SET PASSTHRU RESET#
   ```

   b. Edit the file to update the *server_name* and *remote_authid* with the exact values that are stored in the SYSIBM catalog tables.

   - *server_name* is the server name that was used when you created the server mapping to Oracle. You can find the server name by selecting the SERVERNAME from the SYSIBM.SYSSERVERS table.
   - *remote_authid* is the remote authorization ID that was used when you created the user mapping to the *server_name*. You can find the remote authorization ID by selecting the SETTING from the SYSIBM.SYSUSEROPTIONS table where OPTION = 'REMOTE_AUTHID' and SERVERNAME = '*server_name*'.

   c. Run the file to create the new REG_SYNCH_TRIGGER.

   ```
   db2 -td# -vf filename
   ```

7. Drop all subscription set members. Do not drop the target tables or else you will need to do a full refresh.

8. Drop the subscription sets.
9. Drop all registrations.
10. Create new registrations.
11. Create new subscription sets.
12. Create new subscription set members for the target tables.
13. Start the source server applications.
14. Verify that the target tables are synchronized with the source (for example, use `select count(*)` or some other mechanism).
    - If the tables are not synchronized with the source, go to Step 14 (start Apply). A full refresh will be performed when you start Apply.
    - If your tables are synchronized with the source and you want to bypass a full refresh, change the SYNCHPOINT AND SYNCHTIME values in the IBMSNAP_PRUNCNTL table and the IBMSNAP_SUBS_SET table. For example, run the following SQL statements for every subscription set that is affected:

      At the Capture control server database (source server):

      ```
      UPDATE schema.IBMSNAP_PRUNCNTL
      SET SYNCHPOINT = X'00000000000000000000',
      SYNCHTIME = CURRENT_TIMESTAMP
      WHERE SET_NAME = 'SET001' AND APPLY_QUAL ='AQ001';
      ```

      At the Apply control server database:

      ```
       UPDATE ASN.IBMSNAP_SUBS_SET
      SET LASTRUN = CURRENT_TIMESTAMP,
      LASTSUCCESS = CURRENT_TIMESTAMP,
       SYNCHTIME = CURRENT_TIMESTAMP,
      SYNCHPOINT = NULL
      WHERE SET_NAME = 'SET001' AND APPLY_QUAL ='AQ001';
      ```
15. Start the Apply programs.

Clean up your DataJoiner migration environment

You might want to remove the migration control tables, temporary copies of the pre-Version 8 tables, and staged Version 8 tables when you're sure that you do not want to fall back to your pre-Version 8 environment. You can drop the tables in the backup schema individually, or you can drop the entire table space if there are no other tables in them.

Important: The IBMSNAP_CAPSCHEMAS control table is placed in the *same* table space as another set of staged V8 tables; therefore, you must remove the staged Version 8 tables individually from that table space. The other replication control tables are nicknames in the database.

To drop the migration control tables and the backup copies of the pre-Version 8 tables:

```
DROP TABLESPACE backupts
```

Where *backupts* is the tablespace that was created in the `mig8fed.sql` script for copies of pre-Version 8 tables and migration control tables.

To drop selected tables individually from OTHERTS table space:

```
DROP TABLE bkschema.IBMSNAP_xxxx
```

Where:

- *bkschema* is the backup schema from the `mig8fed.sql` script
- IBMSNAP_*xxxx* is the name of the staged V8 control table (for example, REGISTER)

asnmig4c: Conditioning program (DataJoiner)

Run the **asnmig4c** command from DataJoiner Apply control servers if you have iSeries sources or non-DB2 relational sources or targets.

Before running this command, make sure that you set up an encrypted password file using the **asnpwd** command.

The syntax for the **asnmig4c** command is:

```
asnmig4c db dbname on control server using schema bkschema
 for backup [user userid using password]
```

Table 17. asnmig4c command parameter definitions (DataJoiner)

Parameter Value	Definition
dbname	Specifies the database where the Apply control server is located.
bkschema	Specifies the schema name of the migration control tables. The backup schema that you specify *must* match the schema that was used in the `mig8fed.sql` script that created the control tables. The schema is folded to uppercase.
userid	The user ID to connect to *dbname*[1].
password	The password for the user ID[1].

Notes:

[1] Optional. The user ID and password are required only when accessing remote databases. If you do not provide the user ID and password, the command will check the password file.

Examples for asnmig4c

The following example illustrates the **asnmig4c** command.

Example 1

To update the migration control tables for the mydb Apply control server for all non-DB2 relational targets or sources, and iSeries sources, assuming that the mig8fed.sql script was run with the schema: myschema and the output is piped to an output files called asnmig4c.out:

```
asnmig4c db mydb on control server using schema myschema
   for backup > asnmig4c.out
```

asnmig8: Migration program (DataJoiner)

Use the **asnmig8** command to run migration commands for DataJoiner servers.

```
asnmig8 db dbname on servertype server
using  schema bkschema for command
[user userid using password]
```

Table 18. asnmig8 command parameter definitions (DataJoiner)

Parameter Value	Definition
dbname	Specifies the database where the Capture control server or Apply control server is located.
servertype	Specifies the type of replication server being migrated: **source** The Capture control server. **control** The Apply control server.
bkschema	Specifies the schema name of the migration control tables. The backup schema that you specify *must* match the schema that was used in the mig8fed.sql script. The schema is folded to uppercase.
command	Specifies the action being performed by the migration command: **backup** Backs up the existing (pre-Version 8) control tables for the specified server. It also populates the staged Version 8 tables for the specified server. **migration** Creates the new Version 8 control tables for the specified server. It populates the new tables using data from the staged Version 8 tables. It also removes the old replication tables. **fallback** Moves data from the backup copy of the pre-Version 8 control tables and puts it into the pre-Version 8 replication control tables. It also drops the Version 8 control tables that were created during the migration.
userid	The user ID to connect to *dbname*[1].

Table 18. asnmig8 command parameter definitions (DataJoiner) (continued)

Parameter Value	Definition
password	The password for the user ID[1].

Notes:

[1] Optional. The user ID and password are required only when accessing remote databases.

Examples for asnmig8

The following examples illustrate a few of the uses of the **asnmig8** command.

Examples for Capture control servers (DataJoiner)

The following examples are for DataJoiner Capture control servers and redirect output to a file:

```
asnmig8 db mydb on source server using schema backup for backup > bksrc.out
```

```
asnmig8 db mydb on source server using schema backup for migration > migsrc.out
```

Examples for Apply control servers (DataJoiner)

The following examples are for Linux, UNIX, and Windows Apply control servers, where the backup schema is myschema and the output is directed to files:

```
asnmig8 db otherdb on control server using schema myschema
   for backup > bkctl.out
```

```
asnmig8 db otherdb on control server using schema myschema
   for migration > migctl.out
```

Chapter 8. Migration messages

This section lists messages that are related to the migration process only. For other replication messages, see the *DB2 Replication Guide and Reference*.

ASN5000E ASNMIG8 for *action* ended abnormally at *time_stamp*.

Explanation: For reasons stated in one or more preceding messages, the migration action (backup, migration, or fallback) failed.

User Response: Review the error messages that preceded this one, and respond accordingly.

ASN5001I A successful *action* is complete at *timestamp*.

Explanation: This migration action (backup, migration, or fallback) completed successfully.

User Response: No action is required.

ASN5002E SQL error at line *source_line* in function *function_name*, SQLCODE *sqlcode*, SQLSTATE *sqlstate*.

Explanation: The migration utility detected an SQL error, and presents the function, line, SQLCODE, and SQLSTATE for diagnosis. Any related error message from DB2 follows immediately.

User Response: Consult a DB2 message reference for an explanation of the SQLCODE and SQLSTATE. If the origin of the problem remains unclear, report the full message text to IBM Software Support.

ASN5003E Migration does not recognize the DB2 server.

Explanation: The user has connected to an unsupported server. Migration will end abnormally (abend).

User Response: Migration runs only on DB2 for Linux, UNIX, and Windows, on DB2 for z/OS, or on DB2 for iSeries.

ASN5004E The prerequisite level of Capture has not run.

Explanation: Migration can run only after you prepare the existing Capture program.

User Response: Install and run the correct Capture maintenance:

- "Preparing the existing Capture program for migration (iSeries)" on page 31
- "Preparing the existing Capture program for migration (z/OS)" on page 41
- "Preparing the existing Capture program for migration (Linux, UNIX, Windows)" on page 55

For Linux, UNIX, and Windows, manual steps are described in "Workaround if pre-Version 8 Capture program was not prepared before migration (Linux, UNIX, Windows)" on page 63.

ASN5005E The table serializing migration is missing. Possible causes are incomplete Migration script or an incorrect backup schema parameter.

Explanation: Migration requires temporary tables in its backup schema. One table serializes migration execution, and that table is missing.

User Response: Verify that you used the correct backup schema name when you ran the command. The schema must be the same as the one used in the migration control table script.

For the server to be migrated, you must run the migration script to create migration tables before running the migration programs.

© Copyright IBM Corp. 2002, 2003

ASN5006E The migration status table has *r* rows, not 2.

Explanation: The script that creates migration control tables must run once and only once, with exit upon first error. Otherwise, the tables in the backup schema might be empty or otherwise corrupted.

User Response: Run the script that creates migration control tables if you did not run it already.

If you ran the script before you got the error, drop any migration control tables that were created and rerun the script to create the migration control tables. Ensure that you use the correct backup schema.

ASN5007E The server or its version *version-release* is not supported for migration.

Explanation: The server DB2 is running a version of DB2 not supported for migration. The connected server was not a valid DB2 version that is supported by replication migration.

User Response: See "Supported DB2 versions" on page 8 for a list of DB2 hosts and versions supported for Version 8 replication migration.

ASN5008E The global row is missing

Explanation: For replication Capture control servers other than Federated, the IBMSNAP_REGISTER table is lacking its global row.

User Response: Run the Capture program to insert the global row.

ASN5009E The column definitions of *table_owner*.*table_name* and *backuptable_owner*.*backuptable_name* differ.

Explanation: A table and its backup are out of synch with respect to their column definitions. Table manipulations were probably made that corrupted the asn or backup schema.

User Response: If table manipulations were made, attempt to restart the migration process from the last backup step, and manipulate the tables using extreme caution.

ASN5010E In subscription set appy_qual *apply_qual* set_name *set_name*, the Replica target *table_owner*.*table_name* resides in the same database as its source table.

Explanation: Migration detected an update-anywhere configuration that is not supported in Version 8 without changes. A master source table and an associated replica table cannot coexist in the same DB2 database, subsystem, or data sharing group. To support this configuration, the master source table must be registered in a separate capture schema from the associated replica table.

User Response: Remove the replica subscription sets and replica registrations that are causing the migration failure. Then start migration again. When migration is complete, create a new capture schema and redefine the update-anywhere scenario.

ASN5011I Migration of database *database* on *server_type* server using schema *backup_schema* for *action* is starting at *timestamp*.

Explanation: This message echoes your migration invocation parameters.

- *server_type* is a Capture control server or Apply control server.
- *action* is one of backup, migration, or fallback.

User Response: No action required. The migration utility is working.

ASN5012E After control table script, only *server_type*-server backup is valid.

Explanation: After running the script, you must backup a Capture control server or an Apply control server.

- *server_type* is a Capture control server or Apply control server.

User Response: Run **asnmig8** to backup an Apply control server or Capture control server.

ASN5013E After backup, only a *server_type*-server backup (repeated) and migration are valid.

Explanation: After backing up an Apply control server or Capture control server, you must either back-up again or migrate.

- *server_type* is a Capture control server or Apply control server.

User Response: Run **asnmig8** to backup or migrate an Apply control server or Capture control server.

ASN5014E After migration, the migration is complete, and only a *server_type*-server fallback is valid if desired.

Explanation: After running **asnmig8** to migrate an Apply control server or Capture control server, you can only fall back to the backed-up state.

- *server_type* is a Capture control server or Apply control server.

User Response: Run **asnmig8** for fallback of an Apply control server or Capture control server. See the migration documentation for DB2 Replication for details. Normally, you do not run **asnmig8** for fallback.

ASN5015I asnmig8 usage: asnmig8 database *dbname* on *server_type* server using schema *schema* for *action* [user *user* [using *password*]].

Explanation: This message presents **asnmig8** invocation syntax when you prompt with no arguments, -h, or invalid arguments.

- *dbname* is a database name.
- *server_type* is a Capture control server or Apply control server.
- *schema* is the backup schema.
- *action* is one of backup, migration, or fallback.

User Response: Invoke **asnmig8** per the syntax indicated.

ASN5016E *server_type* server is already at Version 8.

Explanation: Asnmig8 detected that the server database already contains replication Version 8 tables for the specified server.

- *server_type* is a Capture control server or Apply control server.

User Response: Verify that the database is already migrated.

ASN5017E Start and stop Capture.

Explanation: Migration detected an empty warm start table which indicates an invalid value in the CD_OLD_SYNCHPOINT column of the global row of the IBMSNAP.REGISTER table. This could have happened by canceling Capture instead of using the Capture stop command **asncmd**.

User Response: Start pre-Version 8 Capture and stop it using the **asncmd** in z/OS. Pre-Version 8 Capture is unavailable on UNIX or windows. Contact IBM Support to proceed with migration.

ASN5018W SQL warning at line *line* in function *function*, SQLCODE *sqlcode*, SQLSTATE *sqlstate*.

Explanation: The migration utility detected an SQL warning and presents the function, line, SQLCODE, and SQLSTATE for information. Any related warning message from DB2 follows immediately.

User Response: Consult a DB2 message reference for an explanation.

ASN5019I The remote database *database* is running *platform*, *version*.

Explanation: The migration utility queried system catalogs for the remote server name associated with ASN.IBMSNAP_REGISTER for remote server information.

User Response: Verify the information presented.

ASN5021W A CONNECT to the server *server* **failed. If the server is a Federated or iSeries server, re-run asnmig4c.**

Explanation: The connection to the named server failed.

User Response: If the named server is a federated server, re-run **asnmig4c** when the server becomes available.

If the named server is an iSeries server, rerun QZSNMIG(CONDITION) when the server becomes available. Refer to the Migration Guide for details.

ASN5022I asnmig4c usage: asnmig4c database *dbname* **on control server using schema** *schema* **for backup [user** *user* **[using** *password* **]]**

Explanation: This message displays the **asnmig4c** utility invocation syntax if you enter the command with no arguments, with the -h parameter, or with incorrect arguments.

User Response: Invoke **asnmig4c** with the appropriate parameters. Refer to the Migration Guide for details.

ASN5023W The number of user copy predicates referring to table IBMSNAP_UOW is *number***.**

Explanation: Predicates are referencing columns in the UOW table.

User Response: Update the IBMSNAP_SUBS_MEMBR table to take advantage of the new predicate handling function. See the section on subscription member predicates in "Changing defaults to exploit new Version 8 function" on page 21.

ASN5024W The subscription set with apply_qualifier = *apply_qualifier* **and set_name =** *set_name* **is ambiguously defined with regard to its type of set.**

Explanation: Migration code was unable to recognize the type of subscription set. The subscription set was either manually created or created using the replication administration tools and subsequently manipulated manually. It is not clear whether this subscription set is intended for read-only, update-anywhere, or peer-to-peer processing.

User Response: For an overview of set types, see the Subscription set types section in "Unsupported replication migration configurations" on page 9. Contact your IBM Software Support.

ASN5025E To migrate an iSeries database, QZSNMIG8 must run locally.

Explanation: To migrate an iSeries server, you must run QZSNMIG8 locally, not as a remote client, not even as a remote iSeries client.

User Response: Login to the iSeries server that you want to migrate, and run QZSNMIG8 locally.

ASN5027W *Table* **resides in implicit table space.**

Explanation: Migration found a UOW or CD table residing in an implicit table space.

User Response: Move the table to an explicitly named database.

ASN5028E Remote server *server* **contains replication source triggers that must be updated before you run replication after migration.**

Explanation: The required maintenance for the replication triggers and stored procedures created by DJRA was not installed successfully.

User Response: To update the triggers or stored procedures, drop and recreate registrations that are defined for Informix, Sybase, or Microsoft SQL Server source tables before you run Version 8 replication.

ASN5029E Federated replication of source LOB column to target LOB column *column name* **is not supported.**

Explanation: The DB2 Information Integrator cannot write to federated LOB data types.

User Response: Remove the subscription member before you run Version 8 replication.

ASN5030E Federated replication of source column to target column *column name* **must be modified before replication will run successfully.**

Explanation: The DB2 Information Integrator migration transforms DataJoiner nickname columns with a data type of LONG VARCHAR to a CLOB. This transformation causes an error in the replication Apply program when it tries to replicate from a source LONG VARCHAR data type to a migrated target nickname data type of CLOB. Therefore, no data will be replicated for this member.

User Response: You cannot change the nickname data type back to a LONG VARCHAR. For some federated sources, you can change the nickname to a VARCHAR(*x*). See the DB2 Information Integrator migration information for details..

ASN5031W Oracle NET8-wrapper users must manually save pruncntl_trigger before running asnmig8 migration.

Explanation: The Oracle NET8 wrapper cannot read the Oracle long data type which **asnmig8** uses to save the PRUNCNTL_TRIGGER information.

User Response: Save a copy of PRUNCNTL_TRIGGER in Oracle before you continue migrating to Version 8. Refer to “Saving a copy of the pre-Version 8 PRUNCNTL_TRIGGER” on page 74 for instructions on saving a copy of PRUNCNTL_TRIGGER.

ASN5032W Oracle NET8-wrapper users must manually restore pruncntl_trigger after running asnmig8 fallback.

Explanation: **Asnmig8** was unable to restore the prior version of the PRUNCNTL_TRIGGER definition.

User Response: Restore the copy of PRUNCNTL_TRIGGER that you saved before migration. Refer to “Saving a copy of the pre-Version 8 PRUNCNTL_TRIGGER” on page 74 for instructions on saving a copy of PRUNCNTL_TRIGGER.

Index

© Copyright IBM Corp. 2002, 2003

Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

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

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country/region or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country/region where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make

© Copyright IBM Corp. 2002, 2003

improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product, and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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 that has been exchanged, should contact:

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems, and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements, or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious, and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain 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.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (*your company name*) (*year*). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _*enter the year or years*_. All rights reserved.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both, and have been used in at least one of the documents in the DB2 UDB documentation library.

ACF/VTAM
AISPO
AIX
AIXwindows
AnyNet
APPN
AS/400
BookManager
C Set++
C/370
CICS
Database 2
DataHub
DataJoiner
DataPropagator
DataRefresher
DB2
DB2 Connect
DB2 Extenders
DB2 OLAP Server
DB2 Universal Database
Distributed Relational Database Architecture
DRDA
eServer
Extended Services
FFST
First Failure Support Technology
IBM
IMS
IMS/ESA
iSeries
LAN Distance
MVS
MVS/ESA
MVS/XA
Net.Data
NetView
OS/390
OS/400
PowerPC
pSeries
QBIC
QMF
RACF
RISC System/6000
RS/6000
S/370
SP
SQL/400
SQL/DS
System/370
System/390
SystemView
Tivoli
VisualAge
VM/ESA
VSE/ESA
VTAM
WebExplorer
WebSphere
WIN-OS/2
z/OS
zSeries

The following terms are trademarks or registered trademarks of other companies and have been used in at least one of the documents in the DB2 UDB documentation library:

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

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

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

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

Printed in USA

REPL-MIG8-02

Spine information:

IBM DB2 Universal Database Migration Guide: DB2 Replication Version 8