IBM DB2 Information Integrator

Migration Guide

Version 8.2

SC18-7360-01

IBM DB2 Information Integrator

Migration Guide

Version 8.2

SC18-7360-01

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 99.

You can order IBM publications online or through your local IBM representative:

- To order publications online, go to the IBM Publications Center at www.ibm.com/shop/publications/order
- To find your local IBM representative, go to the IBM Directory of Worldwide Contacts at www.ibm.com/planetwide

Contents

About this book

This book describes how to migrate to DB2 Information Integrator, Version 8.2 from:

- DB2 DataJoiner, Version 2.1.1
- DB2 Universal Database for Linux, UNIX, and Windows, Version 7
- DB2 Information Integrator, Version 8.1

This book contains:

- Migration considerations and restrictions
- Tasks to initiate before migration
- Migration procedures
- Tasks to initiate after migration
- Migration reference information

This book is intended for system administrators, database administrators, security administrators, and system operators who need to migrate to DB2 Information Integrator, Version 8.2 from:

- IBM DB2 DataJoiner, Version 2.1.1
- IBM DB2 Universal Database for Linux, UNIX, and Windows, Version 7 federated systems, including federated systems that use DB2 Relational Connect or DB2 Life Sciences Data Connect.

You should be familiar with:

- DB2 Universal Database
- DataJoiner, if you are migrating from DataJoiner
- Standard database terminology
- Database design and database administration
- Your own applications and the data sources that you want access

Chapter 1. DB2 Information Integrator migration overview

This chapter contains the following topics:

- "Overview of migrating to DB2 Information Integrator"
- "Supported migration features and functions" on page 2
- "DataJoiner migration restrictions" on page 4
- "DB2 Version 7 federated system migration restrictions" on page 10

Overview of migrating to DB2 Information Integrator

The migration process converts existing databases and database instances to IBM® DB2® Information Integrator. You can migrate from IBM DB2 DataJoiner®, Version 2.1.1, or from IBM DB2 Universal Database™, Version 7 on UNIX®, Linux, and Windows® systems.

You can migrate to DB2 Information Integrator by using either of these methods:

- Issue migration commands, such as **db2ckmig**, **db2imigr**, and **db2 migrate database**
- Install DB2 Information Integrator and DB2 Universal Database for Linux, UNIX, and Windows, Version 8

You typically do not need to copy files or create links.

If you migrate from DB2 Universal Database Version 7, DB2 Relational Connect and DB2 Life Sciences Data Connect are included in the migration process. You do not need to explicitly migrate DB2 Relational Connect and DB2 Life Sciences Data Connect after you migrate from DB2 Universal Database Version 7 to DB2 Information Integrator.

To migrate from DataJoiner or from DB2 Universal Database you must complete several tasks:

1. Prepare for migration, which includes reading about supported migration features and functions, backing up databases, recording system and database information, and more.
2. Install required software.
3. Migrate objects such as instances, databases, explain tables, and applications.
4. **For DataJoiner for AIX® and Solaris users:** Rebind packages.
5. **For Oracle and Sybase users:** Change the wrapper for the data source. This task is optional.
6. Set up access to the data sources.

Related concepts:

- "Installation procedures for migrating to DB2 Information Integrator" on page 29
- "Supported migration features and functions" on page 2
- "DataJoiner migration restrictions" on page 4
- "DB2 Version 7 federated system migration restrictions" on page 10

Related tasks:

- “Preparing to migrate to DB2 Information Integrator” on page 23

Supported migration features and functions

If you want to access only DB2® family data sources, Informix® data sources, or OLE DB data sources, you do not need to install DB2 Information Integrator. Access to these data sources is provided when you migrate to and install DB2 Enterprise Server Edition.

DB2 Information Integrator Version 8.2 supports migration from:

- DB2 DataJoiner®, Version 2.1.1.
- DB2 Universal Database™ for Linux, UNIX®, and Windows®, Version 7 federated systems. Migration support includes federated systems that use DB2 Relational Connect or DB2 Life Sciences Data Connect.

The features and functions that are supported in DB2 DataJoiner and DB2 Universal Database for Linux, UNIX, and Windows, Version 7 federated systems are supported in DB2 Information Integrator, unless otherwise noted.

You can also use the backup and restore utilities to migrate your system. The DB2 Universal Database backup and restore utilities enable you to restore a database on a system with a different architecture and operating system. The backup and restore method supports moving from any AIX® and Solaris Operating Environment 32-bit DataJoiner Version 2.1.1 database to any AIX and Solaris Operating Environment 64-bit Version 8 database.

Supported data sources

The data sources supported by DB2 DataJoiner, DB2 UDB Version 7, and DB2 Information Integrator are listed in the following table:

Table 1. Data sources supported by DB2 DataJoiner, DB2 UDB Version 7, and DB2 Information Integrator

Data source	DB2 DataJoiner	DB2 UDB Version 7	DB2 Information Integrator Version 8
DB2 Universal Database for Linux, UNIX, and Windows	✓	✓	✓
DB2 Universal Database for z/OS™ and OS/390®	✓	✓	✓
DB2 Universal Database for iSeries™	✓	✓	✓
DB2 Server for VM and VSE	✓	✓	✓
Informix	✓	✓	✓
ODBC	✓		✓
OLE DB		✓	✓
Oracle	✓	✓	✓
Oracle RDB	✓		
Microsoft® SQL Server	✓	✓	✓
Sybase	✓	✓	✓
Sybase SQL Anywhere	✓		
Teradata	✓		✓

Table 1. Data sources supported by DB2 DataJoiner, DB2 UDB Version 7, and DB2 Information Integrator (continued)

Data source	DB2 DataJoiner	DB2 UDB Version 7	DB2 Information Integrator Version 8
BLAST		✓	✓
Documentum		✓	✓
Entrez			✓
Excel files		✓	✓
HMMER			✓
IBM® Lotus® Extended Search			✓
KEGG user defined functions (UDF)			✓
Table-structured files		✓	✓
WBI			✓
Web Services			✓
XML			✓

Migrated objects

When instances and databases are migrated to DB2 Information Integrator, some objects are updated so that they perform properly in a DB2 Information Integrator Version 8 federated system. These changes include updates to client libraries and file paths.

When instances and databases are migrated, the following objects are migrated:

- Wrapper definitions
- Server definitions
- Nicknames

  Exception: Stored procedure nicknames are not migrated because they are not supported in DB2 Information Integrator Version 8.2.
- Tables
- User mappings
- Views

  Exception: Local views that were created from a nickname using the WITH CHECK OPTION clause will be inoperative after migration. This syntax is no longer supported.
- Schema names
- Column names
- Index names
- Data type mappings

  Exception: If a data type is not supported in DB2 Information Integrator, the data type is mapped to a supported data type.
- Package names
- User-defined functions
- Function mappings

  Exception: When migrating from DB2 DataJoiner, function mappings that map a DB2 SYSIBM or SYSFUN function to a remote function are not migrated.

Note: DB2 Information Integrator relational wrappers map DB2 SYSIBM and SYSFUN functions to remote functions by using a set of default function mappings.

Related concepts:

- "DataJoiner migration restrictions" on page 4
- "DB2 Version 7 federated system migration restrictions" on page 10

Related reference:

- "DataJoiner terms and their DB2 Information Integrator equivalents" on page 69

DataJoiner migration restrictions

This section contains the following migration restriction topics:

- "Supported DataJoiner versions"
- "DataJoiner Classic Connect data sources"
- "DB2 UDB run-time client support" on page 5
- "Stored procedures" on page 5
- "Two-phase commit transactions" on page 5
- "LONG data types" on page 5
- "Replication restrictions" on page 7
- "Nicknames" on page 7
- "Invalid data type mappings" on page 7
- "Invalid server options" on page 8
- "Database path location" on page 8
- "Microsoft SQL Server data sources" on page 8
- "Sybase server options" on page 9
- "Abstract data types and extended indexes" on page 9
- "Spatial data" on page 9
- "Code page support" on page 9
- "Changes in SQL DDL syntax" on page 10

Supported DataJoiner versions

DB2® Information Integrator supports migration from:

- DataJoiner® Version 2.1.1 for AIX®
- DataJoiner Version 2.1.1 for Solaris Operating Environment
- DataJoiner Version 2.1.1 for Windows® NT

If you are using DB2 DataJoiner, Version 1.2 or Version 2.1, you must upgrade to DB2 DataJoiner Version 2.1.1 before you can migrate to DB2 Information Integrator.

DataJoiner Classic Connect data sources

DB2 Information Integrator does not support access to DataJoiner Classic Connect data sources such as IMS™ and VSAM. However, you can use DB2 Information Integrator Classic Federation for z/OS™ to access these data sources. For information on using DB2 II Classic Federation for z/OS to access data sources see the *DB2 Information Integrator Classic Federation for z/OS Getting Started Guide*.

DB2 UDB run-time client support

DataJoiner does not support connections from DB2 UDB Version 8 run-time clients or servers. If you connect to DB2 UDB Version 7 run-time clients or servers from DataJoiner, you must migrate from DataJoiner to DB2 Information Integrator Version 8 before you migrate the DB2 UDB Version 7 run-time clients and servers to DB2 UDB Version 8.

Stored procedures

If you are migrating applications from DB2 DataJoiner that invoke stored procedures through nicknames, you must modify those applications. DB2 Information Integrator does not currently support the ability to invoke a stored procedure through a nickname.

Two-phase commit transactions

If you are migrating applications from DB2 DataJoiner that require two-phase commit transactions, you must modify those applications. DB2 Information Integrator currently supports only one-phase commit transactions.

LONG data types

LONG data types are not currently supported in DB2 Information Integrator. Data sources with LONG data types are mapped by default to DB2 UDB LOB data types (CLOB, DBCLOB, and BLOB).

When LONG data type information is migrated to DB2 Information Integrator, the local column data types in the federated database system catalog are changed.

Some examples of the common data type mapping changes:

- The local column data type for LONG VARCHAR columns is converted to the CLOB data type.
- The local column data type for LONG VARGRAPHIC columns is converted to the DBCLOB data type.
- The local column data type for LONG VARCHAR FOR BIT DATA columns is converted to the BLOB data type.

Applications that perform insert, update, or delete operations on LONG data types might return errors when these applications are migrated to DB2 Information Integrator. Although the local column data types are converted to LOB data types, only the Oracle Net8 wrapper currently allows write operation on these columns when the remote type is LOB and not LONG.

If you need to perform insert, update, or delete operations on these local LOB data types, you might be able to alter these LOB data types to VARCHAR data types depending on the size of the data and on the data source that you are accessing.

Accessing LONG data types in Oracle tables

In DataJoiner, you can create tables on remote data sources. When you create the remote table by using DB2 DataJoiner, a nickname for that table is also created. This capability is also available in DB2 Information Integrator and is called *transparent DDL*.

In DB2 Information Integrator, the Oracle NET8 wrapper supports read and write operations on LOB columns. However, after you migrate to DB2 Information Integrator, LONG data type columns will be read only. If you want to perform insert, update, and delete operations on LONG columns, use the Oracle utilities to alter the remote column data type to LOB then recreate the nickname.

In DataJoiner, the following data type mappings can be applied:

The following DB2 data types are mapped to Oracle LONG RAW data types:
- VARCHAR([256-4000]) FOR BIT DATA
- LONG VARCHAR FOR BIT DATA
- BLOB

The following DB2 data types are mapped to Oracle LONG data types:
- VARCHAR([2001-4000])
- CLOB
- LONG VARCHAR

When you migrate to DB2 Information Integrator, the local column data types in the catalog are changed. The local column data type for LONG VARCHAR columns is converted to the CLOB data type. The local column data type for LONG VARCHAR FOR BIT DATA columns is converted to the BLOB data type. The remote columns do not change.

Large object (LOB) data types

The following restrictions apply to applications that access LOB data types.

Read and write operations with LOBs: All relational wrappers allow read operations on LOB data type columns at the supported data sources.

Only the Oracle NET8 wrapper allows read and write operations on LOB data type columns at Oracle data sources.

With the exception of the Oracle NET8 wrapper, DB2 Information Integrator currently does not support write operations on LOB data type columns. Applications that use other wrappers and perform insert, update, or delete operations on LOB data type columns will return errors if the applications are migrated to DB2 Information Integrator.

If you need to perform insert, update, or delete operations on local LOB data type columns, you might be able to alter the data types of these columns to VARCHAR data types depending on the size of the data and on the data source that you are accessing.

Accessing LOB columns in DB2 UDB for UNIX, Windows, and z/OS Version 7 data sources: In DB2 DataJoiner, the DB2RA protocol allowed read and write operations on LOB columns in DB2 Universal Database™ for UNIX®, Windows, and z/OS, Version 7 data sources. DB2 Information Integrator does not support the DB2RA protocol. DB2 Information Integrator uses the DRDA® protocol. The DRDA protocol does not support LOB data types when accessing DB2 Universal Database for UNIX, Windows, and z/OS, Version 7 data sources.

DB2 DataJoiner applications that use the DB2RA protocol to perform operations on LOB columns in DB2 Universal Database for UNIX, Windows, and z/OS, Version 7 data sources will return errors when those applications are migrated to DB2 Information Integrator.

Replication restrictions

The following restrictions apply to replicating data with DataJoiner.

Replicating LONG and LOB data types

When LONG data type information is migrated to DB2 Information Integrator, the local column data types in the federated database system catalog are changed by default to DB2 LOB data types (CLOB, DBCLOB, and BLOB). After the column data types of the nickname are changed from LONG data types to DB2 LOB data types, DB2 replication cannot replicate the LOB data.

If you are currently replicating local LONG data types that the migration process converts to LOB data types, you can alter the local LOB data type columns to VARCHAR data type columns after you migrate to DB2 Information Integrator. Altering the column types to VARCHAR data types allows insert, update, and delete privileges on these columns by DB2 replication, and these column values can be replicated.

You can check the local column data types in the DataJoiner system catalog to determine if LONG or LOB data type columns are present in the nickname for the target database object.

Replication with the DBLIB wrapper

In DB2 Information Integrator, the DBLIB wrapper is read-only. If you use the DBLIB wrapper in DataJoiner for replication, you cannot replicate data after you migrate to DB2 Information Integrator.

After you migrate to DB2 Information Integrator, use the CTLIB wrapper instead of the DBLIB wrapper to access Sybase data sources. You must submit the correct SQL statements to configure the CTLIB wrapper. You can use the **db2look** command to help you generate the proper SQL statements. See the related links for additional information.

Nicknames

In DataJoiner Version 2.1.1, you can create nicknames on DB2 UDB aliases; however, you cannot create nicknames on DB2 UDB aliases in DB2 Information Integrator Version 8.

Triggers referencing nicknames

When migrating from DataJoiner, triggers from DataJoiner that reference nicknames are not migrated to DB2 Information Integrator because DB2 Information Integrator does not support triggers that refer to a nickname in a triggered action.

Invalid data type mappings

DataJoiner does not check to determine if a data type mapping is valid when you create it. Instead, transparent DDL statements that use invalid data type mappings and queries that use invalid data type mappings will fail. However, if you specify a server name with a type mapping, or if you specify a server type that meets some existing server names, the validity of the data type mapping is checked when you attempt to create the mapping. You know immediately that the mapping is invalid because DB2 Information Integrator does not allow you to create the mapping.

You should drop invalid data type mappings in DataJoiner before you migrate to DB2 Information Integrator.

Invalid server options

In DataJoiner Version 2.1.1, the DATEFORMAT, TIMEFORMAT, and TIMESTAMPFORMAT server options are available for remote servers for all wrappers. However, these options are supported only for the ODBC wrapper in DB2 Information Integrator. You must drop these options before you migrate if they are specified on a remote server created for wrappers other than the ODBC wrapper. For the ODBC wrapper, you can drop these server options on the remote server after you migrate to DB2 Information Integrator.

Database path location

DataJoiner Version 2.1.1 databases use a database path convention that is not supported in DB2 Information Integrator. The DataJoiner Version 2.1.1 path is:

```
user_specified_path/instance_name/SQLNNNNN
```

SQLNNNNN is the database subdirectory.

The path convention in DB2 Information Integrator is:

```
user_specified_path/instance_name/NODEXXXX/SQLNNNNN
```

NODEXXXX is the node number, and SQLNNNNN is the database subdirectory.

All DataJoiner Version 2.1.1 databases are relocated from their original path to this new path. The database is automatically relocated during the database migration process. The database relocation consists of the following steps:

1. The database token is moved to the new database path directory.
2. The database entry for the new database path directory is cataloged in the local database directory.
3. The database entry for the original database path directory is removed from the local database directory.
4. The local database directory is deleted from the original database path directory if there are no entries in the local database directory.

Microsoft SQL Server data sources

The following information identifies the data access module naming conventions and the supported drivers for Microsoft® SQL Server data sources.

Data access modules

In DataJoiner for Windows, the default name of the data access module for Microsoft SQL Server is djxmssql. In DataJoiner for UNIX, the default name of the data access module for Microsoft SQL Server is mssqlodbc. Additionally, DataJoiner for UNIX includes a data access module for Microsoft SQL Server that uses the DataDirect SequeLink ODBC driver. The name of this data access module is djxsqlnk.

When the djxmssql, mssqlodbc, and djxsqlnk data access modules are migrated from DataJoiner to DB2 Information Integrator, the names are changed to DJXMSSQL3. Although the names are different, the wrapper links to the correct libraries and functions properly.

DB2 Information Integrator supports two default wrapper names: DJXMSSQL3 and MSSQLODBC3. The default wrapper names are no longer associated with a specific operating system. You can use either default wrapper name when you create a Microsoft SQL Server wrapper. DB2 Information Integrator determines which operating system you are using and creates the correct wrapper library files for that operating system.

If you create a wrapper for Microsoft SQL Server, use the default wrapper name MSSQLODBC3. The default wrapper name DJXMSSQL3 is provided only to support migration from DataJoiner.

DataDirect SequeLink ODBC driver

The DataDirect SequeLink ODBC driver is not supported in DB2 Information Integrator. For UNIX operating systems, you must use the DataDirect Technologies Connect ODBC Version 3.7 (or later) driver.

Sybase server options

DB2 Information Integrator does not support the Sybase server options APPLY_BUFFER_SIZE and APPLY_PACKET_SIZE.

Abstract data types and extended indexes

You cannot migrate DataJoiner databases that contain abstract data types or extended indexes, such as databases that contain DB2 Spatial Extender data. The abstract data types or extended indexes must be removed before the database can be migrated.

Spatial data

If your DataJoiner database contains spatial data, you must migrate this data separately from any nonspatial data.

Before migrating to DB2 Information Integrator, you must:

- Manually export the spatial data tables to an SDE transfer file.
- Drop the tables that contain spatial data columns.
- Run the disable_db stored procedure to remove all of the catalog tables that refer to spatial data.

After the migration of the nonspatial data is complete, you must:

- Enable the DB2 Spatial Extender database that you want to work with.
- Import the spatial data into the federated database.

The specific steps are listed in the related tasks at the end of this topic.

The commands to export and import data in previous versions of DB2 Spatial Extender have changed. You must use the current syntax to import and export data. The *DB2 Spatial Extender Version 8 User's Guide and Reference* contains the current syntax.

Code page support

The code page support for DB2 Universal Database for Linux, UNIX, and Windows might be different than the code page support for DataJoiner. The DB2 Universal Database for Linux, UNIX, and Windows code page support is listed in the topic "Supported DB2 interface languages, locales, and code pages".

Changes in SQL DDL syntax

Some of the SQL DDL statements and syntax that are supported in DataJoiner are different in DB2 Universal Database for Linux, UNIX, and Windows, Version 8. For example:

- For the CREATE VIEW statement, the WITH CHECK OPTION clause is no longer supported.
- For the CREATE TYPE MAPPING statement, a CHAR data type that does not have a specified length is equivalent to CHAR(1); where *1* is the default length of the CHAR data type.

The SQL statements that are supported in DB2 Universal Database Version 8 are documented in the *IBM DB2 Universal Database: SQL Reference*.

Related concepts:

- "About importing and exporting spatial data" in the *IBM DB2 Spatial Extender and Geodetic Extender User's Guide and Reference*

Related tasks:

- "Allowing write operations on large object data types after migrating to DB2 Information Integrator" on page 50
- "Tasks after migrating to DB2 Information Integrator" on page 45
- "Importing SDE transfer data to a new or existing table" in the *IBM DB2 Spatial Extender and Geodetic Extender User's Guide and Reference*
- "Exporting data to an SDE transfer file" in the *IBM DB2 Spatial Extender and Geodetic Extender User's Guide and Reference*

Related reference:

- "Supported DB2 interface languages" in the *Quick Beginnings for DB2 Servers*

DB2 Version 7 federated system migration restrictions

This topic contains information on certain restrictions that apply when you migrate federated systems from DB2® Universal Database Version 7 on UNIX® and Windows® operating systems to DB2 Information Integrator.

Supported DB2 Universal Database versions

DB2 Information Integrator supports federated system migration from DB2 Universal Database Version 7.1 and Version 7.2 on UNIX and Windows operating systems.

Sybase server options

DB2 Information Integrator does not support the Sybase server options APPLY_BUFFER_SIZE and APPLY_PACKET_SIZE.

LONG data types

LONG data types are not supported in DB2 Information Integrator. Data sources with LONG data types are mapped to DB2 LOB data types (CLOB, DBCLOB, and BLOB) by default.

When LONG data type information is migrated to DB2 Information Integrator, the local column data types in the federated database system catalog are changed:

- The local column data type for LONG VARCHAR columns is converted to the CLOB data type.
- The local column data type for LONG VARGRAPHIC columns is converted to the DBCLOB data type.
- The local column data type for LONG VARCHAR FOR BIT DATA columns is converted to the BLOB data type.

Applications that perform insert, update, or delete operations on LONG data types might return errors when the applications are migrated to DB2 Information Integrator. Although the local column data types are converted to LOB data types, the Oracle NET8 wrapper only allows write operations on remote Oracle LOB data type columns.

If you need to perform insert, update, or delete operations on local LOB data types, you might be able to alter these LOB data types to VARCHAR data types depending on the size of the data and on the data source that you are accessing.

Related tasks:

- "Allowing write operations on large object data types after migrating to DB2 Information Integrator" on page 50

Chapter 2. Migration considerations

Before migrating to DB2 Information Integrator, you must understand the migration restrictions and requirements.

This chapter contains the following topics:

- "Migration restrictions"
- "Migration recommendations" on page 14
- "Migration space recommendations" on page 15
- "Reverse migration restrictions" on page 17

Migration restrictions

You should be aware of the following restrictions before you migrate to DB2 UDB Version 8:

- Migration is supported only from:
  - DB2 UDB Version 6.x or Version 7.x. (All platforms supported in Version 6.x and Version 7.x; Linux must be at Version 6 FixPak 2.)
  - DB2 DataJoiner V2.1.1 32-bit (AIX, Windows NT, and Solaris Operating Environment).
- Issuing the **migrate database** command from a DB2 UDB Version 8 client to migrate a database to a DB2 Version 8 server is supported; however, issuing the migration command from an DB2 UDB Version 6 or Version 7 client to migrate a database to a DB2 UDB Version 8 server is not supported.
- When migrating from DB2 DataJoiner V2.1.1, DB2 Information Integrator is required to support non-IBM data sources.
- Migration between platforms is not supported. For example, you cannot migrate a database from a DB2 server on Windows to a DB2 server on UNIX.
- Migrating a partitioned database system that has multiple computers requires that database migration be performed after DB2 UDB Version 8 is installed on all participating computers. Any DB2 migration commands need to be run on each of the participating computers.
- Windows allows only one version of DB2 UDB to be installed on a computer. For example, if you have DB2 UDB Version 7 installed and install DB2 UDB Version 8, DB2 UDB Version 7 is removed during the installation. All instances are migrated during DB2 installation on Windows operating systems.
- User objects within your database cannot have DB2 UDB Version 8 reserved schema names as object qualifiers. These reserved schema names include: SYSCAT, SYSSTAT, and SYSFUN.
- User-defined distinct types using the names BIGINT, REAL, DATALINK, or REFERENCE must be renamed before migrating the database.
- You cannot migrate a database that is in one of the following states:
  - Backup pending
  - Roll-forward pending
  - One or more table spaces in an abnormal state
  - Transaction inconsistent

- Restoration of back-level (DB2 Version 6.x or Version 7.x) database backups is supported, but rolling forward of back-level logs is not supported.
- Database transactions executed between database backup time and the time DB2 UDB Version 8 migration is completed are not recoverable.

Related concepts:

- "DataJoiner migration restrictions" on page 4
- "DB2 Version 7 federated system migration restrictions" on page 10

Related reference:

- "Reverse migration restrictions" on page 17

Migration recommendations

Consider the following recommendations when planning your database migration:

Back up log files before migration when DB2® UDB uses replication

If you use replication for your DataJoiner® and DB2 UDB data, you must archive all of the DB2 log files before migration.

For complete information on migrating your DB2 replication environment, see the IBM® DB2 Information Integrator Migration Guide: Migrating to SQL Replication Version 8 at http://www.ibm.com/software/data/integration/db2ii/support.html.

DataJoiner instance migration

If you want to migrate an instance of DataJoiner or DB2 UDB on which you are running the Capture or Apply programs for DB2 replication, you must prepare to migrate your replication environment before you migrating the instance.

For complete information on migrating your DB2 replication environment, see the IBM DB2 Information Integrator Migration Guide: Migrating to SQL Replication Version 8 at http://www.ibm.com/software/data/integration/db2ii/support.html.

Perform hardware and operating system upgrades separately from DB2 UDB migration

Performing hardware and operating system upgrades separately from DB2 migration simplifies problem determination if you encounter migration difficulties. If you upgrade your software or hardware prior to migrating to DB2, ensure that your system is operating acceptably before attempting DB2 migration.

Dropping the detailed deadlocks event monitor

At the same time a database is created, a detailed deadlocks event monitor is also created. As with any monitor, there is some overhead associated with this event monitor. If you do not want the detailed deadlocks event monitor, then the event monitor can be dropped using the command:

```
DROP EVENT MONITOR db2detaildeadlock
```

To limit the amount of disk space that this event monitor consumes, the event monitor deactivates, and a message is written to the administration notification log, once it has reached its maximum number of output files. Removing output files that are no longer needed allows the event monitor to activate again on the next database activation.

Back-level server support

As you move your environment from DB2 Version 7 to DB2 Version 8, if you migrate your DB2 clients to Version 8 before you migrate all of your DB2 servers to Version 8, there are several restrictions and limitations. To avoid the known restrictions and limitations, migrate all of your DB2 servers to Version 8 before you migrate any of your DB2 clients to Version 8. These restrictions and limitations are not associated with DB2 Connect™; nor with zSeries®, OS/390®, or iSeries™ database servers.

Benchmark DB2 performance

Run a number of test queries before migrating DB2. Record the exact environment conditions when queries are run. Also, keep a record of the **db2expln** command output for each test query. Compare the results before and after migration. This practice can help to identify and correct any performance degradation.

Devise a plan to back out of a migration

There is no utility to reverse a migration. If you must back out of a migration, you might have to remove DB2 Version 8 code from your system, reinstall the previous version of DB2 to recreate back-level instances, and restore your database backups. If you have to back out of a migration, current database backups and detailed records of database and database configuration settings are essential.

Migrating instances with DB2 DataPropagator™ replication

Before migrating an instance of DataJoiner or DB2 UDB on which you are running the Capture or Apply programs for DB2 DataPropagator, read the migration documentation for DB2 DataPropagator Version 8. You must prepare to migrate your replication environment before you migrate the DB2 or DataJoiner instance. You must also perform specific tasks immediately after the migration of your DB2 or DataJoiner instance. Migration documentation for DB2 DataPropagator Version 8 can be found at the http://www.ibm.com/software/data/dpropr/library.html Web site.

Related concepts:

- "Benchmark testing" in the *Administration Guide: Performance*
- "Explain tools" in the *Administration Guide: Performance*

Related tasks:

- "Migrating DB2 UDB (Windows)" on page 31
- "Migrating DB2 UDB (UNIX)" on page 33

Related reference:

- "DB2 Universal Database planned incompatibilities" in the *Administration Guide: Planning*
- "Version 8 incompatibilities with previous releases" in the *Administration Guide: Planning*
- "Version 7 incompatibilities with previous releases" in the *Administration Guide: Planning*

Migration space recommendations

This topic provides information about disk space recommendations for DB2 migration.

Table spaces

Ensure that you have sufficient table space for databases you are migrating. System catalog table space is required for both old and new database catalogs during migration. The amount of space required varies, depending on the complexity of the database, as well as on the number and size of database objects.

system catalog table space (SYSCATSPACE)

Twice the space currently occupied is recommended.

temporary table space (TEMPSPACE1 is the default name)

Twice the system catalog table space is recommended.

To check the size of your table spaces you can use the following commands:

```
db2 list database directory
db2 connect to database_alias
db2 list tablespaces show detail
```

For the system catalog table space, free pages should be equal to or greater than used pages. Total pages for the temporary table space should be twice the amount of total pages for the system catalog table space. To increase the amount of space to a DMS (Database Managed Space) table space, you can add additional containers.

Log file space

The migration process makes many changes to a database in a single transaction. These changes need adequate log space to contain this transaction. If there is inadequate log space, this transaction is rolled back and migration does not complete successfully. To prevent running out of log file space, you should consider doubling the database configuration values for logfilsiz, logprimary and logsecond.

You must update these database configuration values before you migrate the instance to Version 8.2. You cannot update these database configuration parameters before any migration attempt is made on the database. However, if the database migration fails because there is insufficient log file space, you can set the logfilsiz, logprimary and logsecond database configuration parameters to higher values and then re-issue the database migration command. These settings can be reset after the migration is complete.

DataJoiner

When migrating data from DataJoiner, the log file space that you need depends on the size of the database you are migrating. The total size for all log files should be twice the size of the database backup file. Use the following formula to determine if you have sufficient log file space on your DataJoiner system:

```
4096 X logfilsiz X (logprimary + logsecond)  >
   2 X (size of the DataJoiner database backup file)
```

`4096` is the page size of the log file. `logfilsiz`, `logprimary`, and `logsecond` are database configuration parameters.

For example, if the backup file size for the DataJoiner database is 117.5 MB. The total size for all log files needs to be 2 X 117.5 = 235 MB.

If the `logfilsiz` is set to 5000 and the DataJoiner database has 6 `logprimary` files and 4 `logsecond` files, then the total size for all log files is:

```
( 4096 X 5000 X ( 6 + 4) ) = 204.8 MB
```

Since the total file size for all log files is less than twice the backup file size of the DataJoiner database, migration will fail.

You can avoid this failure by increasing the `logfilsiz` log file size, or by creating additional log files before you migrate. These changes must be made before instance migration.

Related tasks:

- "Adding a container to a DMS table space" in the *Administration Guide: Implementation*
- "Migrating DB2 UDB (Windows)" on page 31
- "Migrating DB2 UDB (UNIX)" on page 33

Reverse migration restrictions

Beginning with DB2 UDB Version 8 FixPak 1, the maximum number of tablespaces supported in a database increases from 4096 to 32768.

For the most part, you should be able to move your database from DB2 UDB Version 8 to DB2 UDB Version 8 FixPak 1 without noticing a change or having to do anything special to use the new tablespace limit.

Note the following restrictions related to moving a database from DB2 UDB Version 8 FixPak 1 (or a later DB2 UDB Version 8 FixPak) back to the DB2 Version 8 level:

- If you want to move from DB2 Version 8.2 back to DB2 Version 8.1, you will need to run the **db2demigdbd** command before going back to DB2 Version 8.1. The **db2demigdbd** is a reverse database directory files tool that restores your database directory to its Version 8.1 format.
- Moving a database that contains a tablespace ID higher than 4096 to DB2 Version 8 from DB2 Version 8 FixPak 1 or later is unsupported. Attempting to do so will result in anomalous behavior and improper operation.
- Restoring a database image that contains a tablespace ID higher than 4096 on DB2 Version 8 is unsupported. Attempting to do so results in anomalous behavior and improper operation.
- When moving from DB2 UDB Version 8 FixPak 1 (or a later DB2 UDB Version 8 FixPak) back to DB2 UDB Version 8, the log-skipping functionality is disabled until such time as the DB2TSCHG.HIS file is removed.

Related concepts:

- "Migration recommendations" on page 14

Related tasks:

- "Migrating databases" on page 40
- "Migrating DB2 UDB (Windows)" on page 31
- "Migrating DB2 UDB (UNIX)" on page 33

Related reference:

- "Migration restrictions" on page 13

- “db2demigdbd - Demigrate Database Directory Files Command” in the *Command Reference*

Chapter 3. Before you migrate

Before you migrate to DB2 Information Integrator, Version 8.2, you must perform several tasks. Some tasks are required for every migration. Other tasks are specific to a data source or to an operating system.

This chapter contains the following topics:

- "Backing up databases before DB2 migration"
- "Recording your system configuration settings before DB2 migration" on page 20
- "Recording database information before migrating" on page 21
- "Recording instance information before migrating" on page 22
- "Changing the diagnostic error level before DB2 migration" on page 22
- "Preparing your DataJoiner servers for migration when using DB2 replication" on page 23
- "Preparing to migrate to DB2 Information Integrator" on page 23
- "Verifying that your databases are ready for migration" on page 25
- "Taking a DB2 UDB Version 6 or Version 7 server offline for migration" on page 26

Backing up databases before DB2 migration

Before you start the migration process, it is recommended that you perform an *offline* backup of your databases. If an error occurs during the migration process, database backups are required for recovery.

This topic does not provide the complete syntax for the backup command.

Prerequisites:

- To backup a database, you require SYSADM, SYSCTRL, or SYSMAINT authority.
- Databases must be cataloged. To view a list of all the cataloged databases in the current instance, enter the following command:

```
db2 list database directory
```

Procedure:

Back up each of your local databases by entering the backup database command:

BACKUP Command

```
►►──BACKUP──┬─DATABASE─┬──database-alias───────────────────────────►
            └─DB───────┘

►─┬─────────────────────────────────────┬───────────────────────────►◄
  └─USER──username──┬─────────────────┬─┘
                    └─USING──password─┘
```

where:

DATABASE *database-alias*
Specifies the alias of the database to back up.

USER *username*
Identifies the user name under which to back up the database.

USING *password*
Is the password used to authenticate the user name. If the password is omitted, the user is prompted to enter it.

For example, from a DB2 CLP prompt:

```
db2 backup database sample user pparker using auntmay
```

where the database alias is `sample`, the username is `pparker`, and the password is `auntmay`.

Related concepts:

- "System administration authority (SYSADM)" in the *Administration Guide: Implementation*

Related reference:

- "BACKUP DATABASE Command" in the *Command Reference*
- "Migration space recommendations" on page 15

Recording your system configuration settings before DB2 migration

It is recommended that you record database and database manager configuration settings before DB2 migration. Configuration records can be used to verify that migration was successful and can be useful in problem determination, if you encounter post-migration difficulties.

After you have migrated DB2, it is recommended that you compare the configuration records with post-migration settings to ensure that settings were migrated successfully.

Procedure:

To record system configuration settings:

1. Record your database configuration settings.

   Configuration parameters for a database should be the same on each computer in a partitioned database system. If they are not, save a copy of the database configuration settings for each partition. You can compare configuration settings before and after migration to ensure that they have migrated properly.

   You can list the database configuration settings by entering the following command:

   ```
   db2 get database configuration for database_alias
   ```

   You can direct it to a file by entering the following command:

   ```
   db2 get database configuration for database_alias > c:\db_sample_cfg.txt
   ```

   Perform this task for each database that you are migrating.
2. Record your database manager configuration settings. You can list your database manager configuration settings by entering the following command:

   ```
   db2 get database manager configuration
   ```

Cut and paste this information from the DB2 CLP window and save it as a file.

3. Record the tablespaces for each database you are migrating. You can list your tablespaces using the following command:

   ```
   db2 list tablespaces
   ```
4. Record the list of packages for each database you are migrating. You can retrieve a list of packages using the following command:

   ```
   db2 list packages
   ```
5. For migration from DB2 Version 7, you should record your DB2 profile registry variable settings. After migration, you can add or change your DB2 profile registry variable settings by referring the information you recorded before migration if required.

Related concepts:

- "Configuration parameters" in the *Administration Guide: Performance*

Related reference:

- "GET DATABASE CONFIGURATION Command" in the *Command Reference*
- "GET DATABASE MANAGER CONFIGURATION Command" in the *Command Reference*
- "LIST PACKAGES/TABLES Command" in the *Command Reference*
- "LIST TABLESPACES Command" in the *Command Reference*
- "Configuration parameters summary" in the *Administration Guide: Performance*

Recording database information before migrating

You must record the database information and the node directory information before you migrate to DB2 Information Integrator.

Follow the steps in this task to preserve the necessary database information that you will need if a problem occurs during the instance migration.

Procedure:

To record your database information:

1. Issue the **list node directory** command.

   Record the server names for the databases that you access. Also, record the node names, the host names, and the service names.
2. Issue the **list database directory** command to list the contents of the system database directory.

   Record the database alias names. These names are the aliases for the databases that you access on remote servers.
3. Issue the **list dcs directory**.

   Record the contents of the Database Connection Services directory.
4. Issue one of the following statements:
   - If you are migrating from DataJoiner, issue the following SQL statement:

     ```
     SELECT SERVER,
            SERVER_TYPE,
            NODE,
            DBNAME
       FROM SYSCAT.SERVERS;
     ```
   - If you are migrating from DB2 UDB V7 or V8, issue the following command

```
SELECT SERVERNAME,
       SERVERTYPE,
       OPTION,
       SETTING
  FROM SYSCAT.SERVEROPTIONS
  WHERE OPTION IN ('NODE','DBNAME')
    ORDER BY SERVERNAME, OPTION;
```

Record the server names, node names, and database alias names of each DB2 family server definition that you cataloged.

Related tasks:

- "Recording instance information before migrating" on page 22

Related reference:

- "LIST DATABASE DIRECTORY Command" in the *Command Reference*
- "LIST NODE DIRECTORY Command" in the *Command Reference*
- "LIST DCS DIRECTORY Command" in the *Command Reference*

Recording instance information before migrating

You must record the instance information before you migrate to DB2 Information Integrator.

Follow the steps in this task to preserve the necessary instance information that you will need if a problem occurs during the instance migration.

On UNIX operating systems, you need root authorization to perform this task.

Procedure:

To record your instance information, issue the **db2ilist** command at the operating system prompt. This command lists all the instances that are available on your system.

Related tasks:

- "Recording database information before migrating" on page 21

Related reference:

- "db2ilist - List Instances Command" in the *Command Reference*

Changing the diagnostic error level before DB2 migration

For the duration of migration activities, change the diagnostic error level to 4. The diagnostic error level 4 records all errors, warnings, and informational messages. This information can be used for problem determination should you encounter migration errors. The diagpath configuration parameter specifies the directory that contains the error file, event log file (Windows only), alert log file, and any dump files that may be generated based on the value of the diaglevel parameter.

Procedure:

To change the diagnostic error level before DB2 migration:

- Set the diagnostic error level in the database manager configuration file using the following command:

```
db2 update dbm configuration using diaglevel 4
```

- Type the **db2stop** command.
- Type the **db2start** command.

The diagpath parameter can be set in the database manager configuration file using the following command:

```
db2 update dbm configuration using diagpath directory
```

where *directory* represents the location that you have chosen to store your log files.

Related concepts:

- "Migration recommendations" on page 14

Related reference:

- "UPDATE DATABASE CONFIGURATION Command" in the *Command Reference*
- "Migration restrictions" on page 13

Preparing your DataJoiner servers for migration when using DB2 replication

If you are using DB2 replication, you need to prepare your DataJoiner servers for migration.

| For information about migrating your DB2 replication environment, see the *IBM DB2 Information Integrator Migration Guide: Migrating to SQL Replication Version 8* on theDB2 Information Integrator support Web page.

Related tasks:

- "Migrating databases" on page 40
- "Maintaining an SQL replication environment" in the *IBM DB2 Information Integrator SQL Replication Guide and Reference*

Related reference:

- "asnanalyze: Operating the Analyzer" in the *IBM DB2 Information Integrator SQL Replication Guide and Reference*

Preparing to migrate to DB2 Information Integrator

Before you migrate to DB2 Information Integrator, you must complete the following tasks.

- "Copying the configuration settings"
- "Removing or modifying write operations on LONG and LOB data types from applications" on page 24
- "Dropping local views created using the WITH CHECK OPTION clause" on page 25
- "Verifying your DataJoiner PTF level" on page 25

Copying the configuration settings

Make a copy of the environment variable configuration settings that are currently on your system. The procedures for making a copy of this information are different for DataJoiner and DB2 Universal Database Version 7 federated systems.

Copying the DataJoiner environment variables

You must make a copy of the environment variable settings for the data sources that you access. You must make a copy of this information so that you can restore the configuration settings after you migrate to DB2 Information Integrator.

Procedure:

From the operating system command line, issue the following command:

```
set > djenv.log
```

This command copies the environment variable settings to the djenv.log file. Store this file in a directory that you can retrieve the file from after you migrate the instances and databases.

This command is valid on Windows and UNIX operating systems.

Copying the DB2 UDB Version 7 db2dj.ini file

The db2dj.ini file contains the settings for the environment variables for the data sources that you access. You must make a copy of this information so that you can restore the configuration settings after you migrate to DB2 Information Integrator.

The db2dj.ini file is located in the sqllib/cfg directory. This file contains configuration information about your data sources.

Copy the db2dj.ini file into a directory that you can retrieve the file from after you migrate the instances and databases.

Recommendation: Although the db2dj.ini file is not required if you migrate using the instance migration steps, you can use the information in this file to verify that the migration of your system configuration settings was performed correctly.

Copying the Sybase interfaces file

If you plan to access a Sybase data source after you migrate to DB2 Information Integrator, copy the Sybase interfaces file. The interfaces file is typically located in the $HOME/sqllib directory. You will need to restore the copy of this file after you migrate.

Removing or modifying write operations on LONG and LOB data types from applications

LONG data types are currently not supported in DB2 Information Integrator. Data sources with LONG data types are mapped to DB2 LOB data types (CLOB, DBCLOB, and BLOB) by default.

Applications that perform insert, update, or delete operations on LONG data types might return errors when the applications are migrated to DB2 Information Integrator. Although the local column data types are converted to LOB data types, currently only the Oracle® NET8 wrapper allows write operations on LOB columns. Attempts to perform write operations with other wrappers will return errors.

To avoid these errors, remove write operations on LOB columns from applications that use a wrapper other than the Oracle NET8 wrapper.

Dropping local views created using the WITH CHECK OPTION clause

Local views that were created from a nickname by using the WITH CHECK OPTION clause will be inoperative after migration. This syntax is no longer supported. For example, a view that is created in DataJoiner with the following syntax will be inoperative after you migrate:

```
CREATE VIEW my_view AS SELECT * FROM my_nickname
    WHERE SCORE > 90 WITH CHECK OPTION
```

Before you migrate, drop local views that were created from a nickname by using the WITH CHECK OPTION clause. Create the views again without the WITH CHECK OPTION clause.

Verifying your DataJoiner PTF level

If you are migrating from DataJoiner, check the PTF level that is installed on your DataJoiner database.

Recommendation: Install PTF 12 or later on your DataJoiner database before you migrate your DataJoiner database to DB2 Information Integrator.

Migrating DataJoiner on AIX Version 4.1

DB2 Information Integrator does not support AIX Version 4.1. If your DataJoiner system runs on AIX Version 4.1, you must complete the following steps before you migrate DataJoiner to DB2 Information Integrator:

1. Upgrade AIX Version 4.1 to AIX Version 4.3.3 with maintenance level 9 or later.
2. Install DB2 Information Integrator Version 8.2 on the upgraded AIX server. See the *DB2 Information Integrator Installation Guide*.

Related concepts:

- "DataJoiner migration restrictions" on page 4
- "DB2 Version 7 federated system migration restrictions" on page 10

Verifying that your databases are ready for migration

This task describes how to use the **db2ckmig** command to verify that your databases are ready for migration.

The **db2ckmig** command to verify that databases owned by the current instance are ready to be migrated. The **db2ckmig** command verifies that:

- A database is not in an inconsistent state
- A database is not in a backup-pending state
- A database is not in rollforward-pending state
- Tablespaces are in a normal state

Procedure:

1. Log on to the instance that you are migrating from.
   - If you are migrating from a previous version of DB2 for Linux, UNIX, and Windows, log on as the DB2 instance owner.
   - If you are migrating from DataJoiner, log on as the DataJoiner instance owner.

2. Issue the **db2stop** command to stop the instance.
3. On Windows, insert the DB2 Version 8 product CD in the CD-ROM.
4. From a DB2 command line prompt, change to the appropriate directory.
   - On AIX, the path for this command is /usr/opt/db2_08_01/bin/db2ckmig.
   - On Linux, HP-UX, and the Solaris Operating Environment, the path for this command is /opt/IBM/db2/V8.1/bin/db2ckmig.
   - On Windows, change to the \db2\Windows\utilities directory on the CD.
5. Issue the **db2ckmig** *db_alias* **/l directory** command to verify that the databases that are owned by the current instance are ready to be migrated and to generate a log file.

   If the **db2ckmig** command does not find errors, the following message appears:

   ```
   db2ckmig was successful.  Database(s) can be migrated.
   ```
6. If the **db2ckmig** command finds errors, correct the problems before you migrate.
7. Ensure that the migration.log file, found in the instance owner's home directory, contains the following text: `Version of DB2CKMIG being run: VERSION 8`.

Related concepts:

- "Installation procedures for migrating to DB2 Information Integrator" on page 29

Related tasks:

- "Migrating DB2 UDB (UNIX)" on page 33

Related reference:

- "db2ckmig - Database Pre-migration Tool Command" in the *Command Reference*

Taking a DB2 UDB Version 6 or Version 7 server offline for migration

This task describes how to take your DB2 UDB Version 6 or Version 7 server offline for DB2 migration. Before you can continue with the migration process, you must stop the DB2 license service, stop all command line processor sessions, disconnect applications and users, and stop the database manager.

Prerequisites:

- Your system must meet the installation requirements for DB2 Version 8 before starting the migration process.
- You must have SYSADM authority.

Procedure:

To take your server offline:

1. Stop the DB2 license service by entering the **db2licd -end** command.
2. On Windows 2000, the properties of a service can be set so that it restarts if the service fails. If the *restart on failure* option is set for any DB2 services, it must be disabled before proceeding.
3. Stop all command line processor sessions by entering the **db2 terminate** command in each session that was running the command line processor.
4. Disconnect all applications and users. To get a list of all database connections for the current instance, enter the **db2 list applications** command. If all applications are disconnected, this command returns the following message:

```
SQL1611W No data was returned by the Database System Monitor.
SQLSTATE=00000
```

You can disconnect applications and users by issuing the **db2 force applications** command.

5. When all applications and users are disconnected, stop each database manager instance by entering the **db2stop** command.

Related reference:

- "db2stop - Stop DB2 Command" in the *Command Reference*
- "FORCE APPLICATION Command" in the *Command Reference*
- "LIST APPLICATIONS Command" in the *Command Reference*

Chapter 4. Migrating to DB2 Information Integrator

The procedures that you use to migrate to DB2 Information Integrator vary depending on the operating system that you use. This chapter contains information about the sequence of steps that you must follow when you install DB2 Information Integrator and the steps for migrating your instances and databases.

Installation procedures for migrating to DB2 Information Integrator

Before you migrate to DB2® Information Integrator, you must install the software in the correct sequence. The sequence that you use depends on the operating system that you use and the version of the product that you are migrating from.

Installation sequence for migrating instances and databases on UNIX operating systems

You must complete the premigration tasks before you install the software, with the exception of the task to verify your databases.

DataJoiner 2.1.1

To migrate from DataJoiner® 2.1.1, follow the instructions in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows* to install the most recent versions of DB2 Universal Database™ Enterprise Server Edition and DB2 Information Integrator. Use the steps for setting up the server for the data sources that you want to access.

Migrating from DataJoiner to DB2 Information Integrator on the same computer:

- To migrate instances, issue the **db2imigr** command.
- To migrate databases, issue the **db2 migrate database** command.

Migrating from DataJoiner on one computer to DB2 Information Integrator on a different computer:

- To migrate databases, issue the **db2 restore db <database_name>** command.

DB2 UDB for Windows and UNIX Version 7 (or earlier), DB2 Relational Connect Version 7, or DB2 Life Sciences Data Connect

To migrate from DB2 UDB for Windows® and UNIX® Version 7 (or earlier), DB2 Relational Connect Version 7, or DB2 Life Sciences Data Connect, follow the instructions in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows* to install the most recent versions of DB2 Universal Database Enterprise Server Edition and DB2 Information Integrator. Use the steps for setting up the server for the data sources that you want to access.

Migrating from DB2 UDB to DB2 Information Integrator on the same computer:
To migrate instances, issue the **db2imigr** command. To migrate databases, issue the **db2 migrate database** command.

Migrating from DB2 UDB on one computer to DB2 Information Integrator on a different computer:
To migrate databases, issue the **db2 restore db <database_name>** command.

DB2 Information Integrator for UNIX, Version 8

To migrate from Information Integrator for UNIX, Version 8: :

1. If DB2 Information Integrator Version 8.1 is installed, install Fix Pack 7 to bring DB2 Information Integrator to version 8.2.
2. If you want to install wrappers that are new for DB2 Information Integrator Version 8.2, follow the instructions in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows* to install DB2 Information Integrator. Use the steps for setting up the server for the data sources that you want to access.

Installation sequence for migrating instances and databases on Windows operating systems

Before you install DB2 Universal Database Enterprise Server Edition and DB2 Information Integrator, complete all the premigration tasks, including the steps in the topic ″Verifying that your databases are ready for migration.″

DataJoiner 2.1.1

To migrate from DataJoiner 2.1.1:

1. Follow the instructions in *IBM DB2 Universal Database Quick Beginnings for DB2 Servers* to install the most recent version of DB2 Universal Database Enterprise Server Edition.
2. Follow the instructions in the *IBM DB2 Information Integrator Installation Guide* to install DB2 Information Integrator. Use the steps for setting up the server for the data sources that you want to access.

   Migrating from DataJoiner to DB2 Information Integrator on the same computer:

   To migrate databases, issue the **db2 migrate database** command.

   Migrating from DataJoiner on one computer to DB2 Information Integrator on a different computer:

   To migrate databases, issue the **db2 restore db <database_name>** command.

   On Windows systems, when you install DB2 Information Integrator Version 8.2, the instances are migrated automatically. You do not need to run the db2imigr command.

DB2 UDB for Windows and UNIX, Version 7 (or earlier), DB2 Relational Connect, or DB2 Life Sciences Data Connect

To migrate from DB2 UDB for Windows and UNIX, Version 7 (or earlier), DB2 Relational Connect, or DB2 Life Sciences Data Connect:

1. Follow the instructions in *IBM DB2 Universal Database Quick Beginnings for DB2 Servers* to install the most recent version of DB2 Universal Database Enterprise Server Edition.
2. Follow the instructions in the *IBM DB2 Information Integrator Installation Guide* to install DB2 Information Integrator. Use the steps for setting up the server for the data sources that you want to access.

Migrating from DB2 UDB to DB2 Information Integrator on the same computer:

To migrate databases, issue the **db2 migrate database** command.

Migrating from DB2 UDB on one computer to DB2 Information Integrator on a different computer:

To migrate databases, issue the **db2 restore db <database_name>** command.

DB2 Information Integrator Version 8

To migrate from DB2 Information Integrator Version 8:

1. If DB2 Information Integrator version 8.1 is installed, install Fix Pack 7 to bring DB2 Information Integrator to version 8.2.
2. If you want to install wrappers that are new for DB2 Information Integrator Version 8.2, follow the instructions in the *IBM DB2 Information Integrator Installation Guide* to install DB2 Information Integrator. Use the steps for setting up the server for the specific data sources that you want to access.

Related concepts:

- "DB2 UDB Enterprise Server Edition" in the *Quick Beginnings for DB2 Servers*
- "DB2 Information Integrator installation process - overview" in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows*

Related tasks:

- "Verifying that your databases are ready for migration" on page 25
- "Migrating databases" on page 40
- "Migrating instances (UNIX)" on page 34
- "Installing DB2 Information Integrator (Windows)" in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows*

Related reference:

- "DB2 Universal Database Version 8.2 is installed" in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows*
- "Clean installation" in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows*
- "An unsupported edition or version of DB2 Universal Database is installed" in the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows*

Migrating Windows servers

This section contains information about migrating DB2 Information Integrator on Windows servers.

For information about migrating databases, see "Migrating databases" on page 40 and "Migrating databases to a new system" on page 42.

Migrating DB2 UDB (Windows)

This topic lists the steps for migrating to DB2 UDB Version 8 on Windows.

Migration is required if you have instances and databases from DB2 Version 6 or Version 7 that you want to use with DB2 Version 8. No database migration is required if the database has been migrated to a DB2 Version 8 FixPak level. On Windows, all existing instances are automatically migrated during the DB2 Version 8 installation.

Prerequisites:

Before migrating:

- Review the migration recommendations, restrictions, and space recommendations.
- Record configuration settings before DB2 migration.
- Change the diagnostic error level.
- Verify that databases are ready for DB2 migration.
- Back up your databases.
- If you are using replication, you must archive all of the DB2 log files.
- You must have SYSADM authority.
- Take the DB2 server offline for DB2 migration.

Procedure:

To migrate DB2 UDB:

1. Install your DB2 server. On Windows, all existing instances are migrated during the DB2 Version 8 installation.
2. Migrate databases.
3. Optional: Migrate DB2 Explain tables.

Related concepts:

- "System administration authority (SYSADM)" in the *Administration Guide: Implementation*
- "Migration recommendations" on page 14

Related tasks:

- "Recording your system configuration settings before DB2 migration" on page 20
- "Changing the diagnostic error level before DB2 migration" on page 22
- "Verifying that your databases are ready for migration" on page 25
- "Backing up databases before DB2 migration" on page 19
- "Taking a DB2 UDB Version 6 or Version 7 server offline for migration" on page 26
- "Migrating databases" on page 40
- "Migrating explain tables" on page 39

Related reference:

- "ARCHIVE LOG Command" in the *Command Reference*
- "Migration space recommendations" on page 15
- "Migration restrictions" on page 13

Migrating UNIX servers

The following topics are included in this section on migrating UNIX servers:

- "Migrating DB2 UDB (UNIX)" on page 33
- "Migrating instances (UNIX)" on page 34

When migrating UNIX servers, you have the option of migrating from 32-bit servers to 64-bit servers. The steps to migrate to 64-bit enabled servers are discussed in separate topics:

- "Migrating to DB2 Version 8 64-bit (UNIX)" on page 36

- “Migrating to DB2 UDB Version 8 on a UNIX 64-bit system from DataJoiner 2.1.1 on a UNIX 32-bit system” on page 37

For information about migrating databases, see “Migrating databases” on page 40 and “Migrating databases to a new system” on page 42.

Migrating DB2 UDB (UNIX)

This topic lists the steps for migrating to DB2 Version 8 on UNIX.

Migration is required if you have instances and databases from DB2 Version 6 or Version 7 that you want to use with DB2 Version 8.

No database migration is required if the database has been migrated to a DB2 Version 8 FixPak level.

Prerequisites:

- Review migration recommendations, restrictions, and space recommendations.
- Record configuration settings before DB2 migration.
- Change the diagnostic error level.
- Take the DB2 server offline for DB2 migration.
- Back up your databases.
- If you will be using replication, you must archive all of the DB2 log files.
- You require SYSADM authority.

Procedure:

To migrate DB2 UDB:

1. Install your DB2 server.
2. Migrate instances.
3. *Optional:* If you have created a DB2 tools catalog and want to use your existing pre-Version 8 scripts and schedules (for the Control Center), you must migrate the DB2 Administration Server.
4. Migrate databases.

Related concepts:

- “System administration authority (SYSADM)” in the *Administration Guide: Implementation*
- “Migration recommendations” on page 14

Related tasks:

- “Recording your system configuration settings before DB2 migration” on page 20
- “Changing the diagnostic error level before DB2 migration” on page 22
- “Taking a DB2 UDB Version 6 or Version 7 server offline for migration” on page 26
- “Backing up databases before DB2 migration” on page 19
- “Migrating instances (UNIX)” on page 34
- “Migrating the DB2 Administration Server (DAS)” on page 35
- “Migrating databases” on page 40

Related reference:

- "ARCHIVE LOG Command" in the *Command Reference*
- "Migration space recommendations" on page 15
- "Migration restrictions" on page 13

Migrating instances (UNIX)

You can migrate existing DB2 Version 6 or DB2 Version 7 instances using the **db2imigr** command. Migrating instances is done after installing DB2 Version 8.

The **db2imigr** command does the following:

- Checks cataloged databases owned by the instance to make sure they are ready for migration.
- Runs the **db2icrt** command to create the DB2 Version 8 instance.
- Updates system and local database directories to a Version 8 format.
- Merges the DB2 Version 6 and DB2 Version 7 database manager configurations with DB2 Version 8 database manager configuration.

Prerequisites:

You must be logged in as a user with root authority.

Before running the **db2imigr** command, it is recommended:

- You should have 20 MB of /tmp free space. The instance migration trace file is written to /tmp.
- That you run the **db2ckmig** command manually or each database in the instance and resolve any problems prior to running the **db2imigr** command for the instance. The **db2imigr** command will not migrate as long as the **db2ckmig** command finds problems.

Restrictions:

Migration is supported only from:

- DB2 Version 6.x or Version 7.x. (All platforms supported in Version 6.x and Version 7.x; Linux must be at Version 6 FixPak 2.)
- DB2 DataJoiner V2.1.1 (AIX, Windows NT, and Solaris Operating Environment).

Procedure:

To migrate an instance:

1. Migrate instances using the **db2imigr** command:

   ```
   DB2DIR/instance/db2imigr [-u fencedID] InstName
   ```

   where

   DB2DIR

   is /usr/opt/db2_08_01 on AIX and /opt/IBM/db2/V8.1 on all other UNIX operating systems.

   -u fencedID

   Is the user under which the fenced user-defined functions (UDFs) and stored procedures will execute. This parameter is required only when migrating from a client instance to a server.

InstName
Is the login name of the instance owner.

If you have migrated from a single-partitioned database environment version of DB2 to a partitioned database environment version of the DB2 Enterprise Server Edition, you must update your instances to a partitioned format using the **db2iupdt** command.

Related tasks:

- "Verifying that your databases are ready for migration" on page 25

Related reference:

- "db2ckmig - Database Pre-migration Tool Command" in the *Command Reference*
- "db2imigr - Migrate Instance Command" in the *Command Reference*
- "db2icrt - Create Instance Command" in the *Command Reference*
- "db2iupdt - Update Instances Command" in the *Command Reference*

Migrating the DB2 Administration Server (DAS)

If you have created a DB2 tools catalog on your DB2 Version 8 system and want to use your existing pre-Version 8 scripts and schedules (for the Control Center) that were created in the pre-Version 8 DB2 Administration Server (DAS), you must migrate the DAS to Version 8.

On Windows, this migration is done automatically if you created the DB2 tools catalog during the installation of Version 8. If you created the DB2 tools catalog after installation, this migration must be done manually.

On UNIX, this migration must be done manually after the DB2 tools catalog has been created, either during the installation or at a later time.

Prerequisites:

You must have:

- An existing DB2 tools catalog.
- DASADM authority on Windows systems to migrate the pre-Version 8 information into the DB2 tools catalog.
- root authority on UNIX systems to migrate the pre-Version 8 information into the DB2 tools catalog.

Procedure:

To migrate a pre-Version 8 DAS to the DB2 tools catalog, enter the command:

```
dasmigr previous_das_name new_das_name
```

where *previous_das_name* represents the name of the pre-Version 8 DAS instance, and *new_das_name* represents the name of the new DB2 Version 8 DAS.

Related tasks:

- "Migrating DB2 UDB (Windows)" on page 31
- "Migrating DB2 Personal Edition (Windows)" in the *Quick Beginnings for DB2 Personal Edition*

- “Migrating DB2 Personal Edition (Linux)” in the *Quick Beginnings for DB2 Personal Edition*

Related reference:

- “dasmigr - Migrate the DB2 Administration Server Command” in the *Command Reference*

Migrating to DB2 Version 8 64-bit (UNIX)

This topic lists the steps for migrating your database to DB2 Version 8 on UNIX 64-bit. DB2 Version 8 64-bit support exists on AIX, HP-UX, Linux, and Solaris Operating Environment.

Prerequisites:

- Before updating to DB2 Version 8 on 64-bit, migrate your databases. This database migration must be done after issuing the **db2imigr** command but before issuing the **db2iupdt –w64** command.

Restrictions:

- You cannot revert to a Version 7 32-bit instance after you have migrated to a Version 8 64-bit instance.
- The information in this topic does not apply to Linux. Use the backup and restore utilities to move from a Linux 32–bit DB2 Version 8 database to a Linux 64-bit DB2 Version 8 database.

Procedure:

To migrate from a DB2 Version 7 database to a DB2 Version 8 64-bit system:

1. Install DB2 Version 8 on your 64-bit system. Do not uninstall the previous version.
2. Run the **DB2DIR/bin/db2ckmig** command as the instance owner to ensure that your database can be migrated.
3. Back up your existing DB2 Version 7 database.
4. Stop your DB2 Version 7 instance.
5. Run the **DB2DIR/instance/db2imigr <instance_name>** command.
6. If your existing database is within a DB2 Version 7 32-bit instance on your 64-bit system, you need to update the instance to a 64-bit instance. To update your Version 7 32-bit instance to a Version 7 64-bit instance, use the **db2iupdt** command, specifying the **–w** parameter with a value of **64**. For example:

   ```
   DB2DIR/instance/db2iupdt -w 64 db2inst1
   ```

   where DB2DIR represents the DB2 Version 8 installation path on the appropriate platform.
7. Restart your instance.

You can use the backup and restore utilities to migrate your system. The backup and restore method supports moving from any of AIX, HP-UX, Solaris Operating Environment 32-bit DB2 Version 8 database to any of AIX, HP-UX, Solaris Operating Environment 64-bit Version 8 database; or Linux 32-bit DB2 Version 8 database to Linux 64-bit Version 8 database (Linux IA64 or Linux AMD64).

Related reference:

- “BACKUP DATABASE Command” in the *Command Reference*
- “RESTORE DATABASE Command” in the *Command Reference*
- “db2icrt - Create Instance Command” in the *Command Reference*
- “db2iupdt - Update Instances Command” in the *Command Reference*

Migrating to DB2 UDB Version 8 on a UNIX 64-bit system from DataJoiner 2.1.1 on a UNIX 32-bit system

DB2 Universal Database Version 8 on UNIX 64-bit systems is supported on AIX and the Solaris Operating Environment.

Prerequisites:

1. Ensure that the wrapper for the data source that you want to access is available in 64-bit mode.
2. Ensure that all data source client software is installed and configured on the 64-bit system.
3. If the wrapper is available in 64-bit mode, run the **djxlink** command to create 64-bit wrapper libraries in the DB2 base directories.
4. When you are migrating from DataJoiner, ensure that the total size for all log files is twice the backup file size of the database. Use the following formula to determine if you have sufficient log file space on the DataJoiner system:

   ```
   ( 4096 X logfilsiz X ( logprimary + logsecond) )  >  2 X (DataJoiner
                                                              database
                                                              backup file size)
   ```

   `4096` is the page size of the log file. `logfilsiz`, `logprimary`, and `logsecond` are database configuration parameters.

Restrictions:

- You cannot revert to a DataJoiner Version 2.1.1 32-bit instance after you migrate to a DB2 Universal Database Version 8 64-bit instance.
- DB2 Universal Database Version 8 on UNIX 64-bit system is not supported on Linux operating systems.

Procedure:

To migrate from a DataJoiner 2.1.1 32-bit database to a DB2 Universal Database Version 8 64-bit system:

1. Install DB2 Universal Database Version 8 and DB2 Information Integrator on the DataJoiner system. Do not remove the previous version.
2. Stop the DataJoiner Version 2.1.1 instance.
3. On the DataJoiner 2.1.1 32-bit system, issue the **/usr/opt/db2_08_01/bin/db2ckmig** command as the instance owner to ensure that the database can be migrated.
4. Back up the DataJoiner Version 2.1.1 database.
5. Install DB2 Universal Database Version 8 and DB2 Information Integrator on the 64-bit computer.
6. Create the 32-bit instance either by using the installer or by issuing the **db2icrt -w 32 -u <instance_name>** command.
7. Migrate the database by issuing the **db2 restore database** command:

```
>>──RESTORE──┬─DATABASE─┬──source-database-alias──────────────────────────>
             └─DB───────┘

>──┬───────────────────────────┬───────────────────────────────────────────><
   └─FROM──directory──device───┘
```

The DATABASE *source-database-alias* parameter is the alias of the database from which the backup was taken. The FROM *directory/device* parameter is the fully qualified path name of the directory or device on which the backup image resides. If the FROM parameter is omitted, the default value is the current working directory of the client computer. This target directory or device must exist on the database server.

8. Update the instance to a DB2 Universal Database Version 8 64-bit instance by issuing the **db2iupdt** command with the **–w** parameter set to **64**.

   For example:

   ```
   DB2DIR/instance/db2iupdt -w 64 db2inst1
   ```

   DB2DIR is the DB2 Universal Database Version 8 installation path on the appropriate operating system.
9. Restart the instance.

Troubleshooting:

If you receive error SQL2519N (which indicates that the database was restored but not migrated), migrate the database:

1. Issue the **db2 migrate database** command.

   ```
   >>──MIGRATE──┬─DATABASE─┬──database-alias───────────────────────────────>
                └─DB───────┘

   >──┬────────────────────────────────────────┬─────────────────────────────><
      └─USER──username──┬────────────────────┬─┘
                        └─USING──password────┘
   ```

   The DATABASE *database-alias* parameter is the alias of the database that you want to migrate to the currently installed version of the database manager. The USER *username* parameter is the user name that you want to use to migrate the database. The USING *password* parameter is the password for the user name.

   If the log file size is still not large enough, you receive the following error:

   ```
   SQL1704N  Database migration failed.  Reason code "3".
   ```

   Increase log file size, and issue the **db2 migrate database** command again.
2. Reset the `logfilsiz`, `logprimary`, and `logsecond` parameters.

Related tasks:

- "Migrating databases" on page 40
- "Migrating instances (UNIX)" on page 34

Related reference:

- "db2ckmig - Database Pre-migration Tool Command" in the *Command Reference*
- "db2imigr - Migrate Instance Command" in the *Command Reference*
- "db2icrt - Create Instance Command" in the *Command Reference*
- "db2iupdt - Update Instances Command" in the *Command Reference*

Migrating explain tables

The **migrate database** command does not migrate explain tables. If you want to migrate explain tables for a database that you migrated to DB2 version 8 or if you are migrating from DataJoiner, you must perform the extra step of migrating explain tables using the **db2exmig** command.

You may want to migrate explain tables if you wish to maintain explain table information that you previously gathered. If you do not want to maintain previously gathered explain table information, you can later recreate the explain tables and gather new information.

Procedure:

To migrate the explain tables, use the **db2exmig** command:

```
db2exmig -d dbname -e explain_schema [-u userid password]
```

where:

- *dbname* represents the database name. This parameter is required.
- *explain_schema* represents the schema name of the explain tables to be migrated. This parameter is required.
- *userid* and *password* represent the current user's ID and password. These parameters are optional.

The explain tables belonging to the user ID that is running **db2exmig**, or that is used to connect to the database, will be migrated. The explain tables migration tool will rename Version 6 or Version 7 tables, create a new set of tables using the EXPLAIN.DDL, and copy the contents of the old tables to the new tables. Finally, it will drop the old tables. The **db2exmig** command will preserve any user added columns on the explain tables.

Related concepts:

- "Explain tools" in the *Administration Guide: Performance*
- "The explain tables and organization of explain information" in the *Administration Guide: Performance*

Related tasks:

- "Migrating databases" on page 40

Migrating applications to DB2 Version 8

DB2® Version 8 supports the following DB2 versions for migration:

- DB2 Version 6
- DB2 Version 7.1
- DB2 Version 7.2
- DataJoiner® Version 2.1.1

When you migrate to a later version of DB2, your database and node directories are migrated automatically. To migrate from any other previous version of DB2, you must first migrate to one of the above supported versions, and then migrate from that version to DB2 version 8.

HP-UX

If you are migrating DB2 from HP-UX Version 10 or earlier to HP-UX Version 11, your DB2 programs must be re-precompiled with DB2 on HP-UX Version 11 (if they include embedded SQL), and must be re-compiled. This includes all DB2 applications, stored procedures, user-defined functions and user exit programs. As well, DB2 programs that are compiled on HP-UX Version 11 cannot run on HP-UX Version 10 or earlier. DB2 programs that are compiled and run on HP-UX Version 10 can connect remotely to HP-UX Version 11 servers.

Micro Focus COBOL

Any existing applications precompiled with DB2 Version 2.1.1 or earlier and compiled with Micro Focus COBOL should be re-precompiled with the current version of DB2, and then recompiled with Micro Focus COBOL. If these applications built with the earlier versions of the IBM® precompiler are not re-precompiled, there is a possibility of database corruption if abnormal termination occurs.

Related concepts:

- "Migration recommendations" on page 14
- "Migrating Java applications, routines, and applets" in the *Application Development Guide: Building and Running Applications*
- "Running applications on two versions of DB2" in the *Application Development Guide: Building and Running Applications*

Related tasks:

- "Migrating databases" on page 40
- "Migrating instances (UNIX)" on page 34
- "Migrating DB2 UDB (Windows)" on page 31
- "Migrating DB2 UDB (UNIX)" on page 33
- "Migrating applications from 32-bit to 64-bit environments" in the *Application Development Guide: Building and Running Applications*
- "Ensuring application portability" in the *Application Development Guide: Building and Running Applications*

Related reference:

- "Administrative APIs and application migration" in the *Administrative API Reference*
- "Migration restrictions" on page 13
- "Version 8 incompatibilities with previous releases" in the *Administration Guide: Planning*

Migrating databases

Prerequisites:

You require SYSADM authority.

Restrictions:

Migration is supported only from:

- DB2 Version 6.x or Version 7.x. (all platforms supported in Version 6.x and Version 7.x)

- DB2 DataJoiner Version 2.1.1 (AIX, Windows NT, and Solaris Operating Environment).

No database migration is required if the database has been migrated to a DB2 Version 8 FixPak level.

Procedure:

To migrate a DB2 database:

1. Migrate the database using the **db2 migrate database** command.

   DB2 MIGRATE DATABASE command

   ```
   >>──MIGRATE──┬─DATABASE─┬──database-alias──────────────────────────────>
                └─DB───────┘

   >──┬──────────────────────────────────┬─────────────────────────────────><
      └─USER──username──┬──────────────────┬─┘
                        └─USING──password──┘
   ```

   where:

   DATABASE *database-alias*
   Specifies the alias of the database to be migrated to the currently installed version of the database manager.

   USER *username*
   Identifies the user name under which the database is to be migrated.

   USING *password*
   The password used to authenticate the user name. If the password is omitted, but a user name was specified, the user is prompted to enter it.
2. Optional: Update statistics for local tables within the database. When database migration is complete, old statistics that are used to optimize query performance are retained in the catalogs. However, DB2 Version 8 has statistics that are modified or do not exist in DB2 Version 6 or DB2 Version 7. To take advantage of these statistics, you may want to execute the **runstats** command on tables, particularly those tables that are critical to the performance of your SQL queries.
3. Optional: Rebind packages. During database migration, all existing packages are invalidated. After the migration process, each package is rebuilt when it is used for the first time by the DB2 Version 8 database manager. You can run the **db2rbind** command to rebuild all packages stored in the database.
4. Optional: Revoke EXECUTE privileges on external stored procedures that contain SQL data access from PUBLIC. During database migration, EXECUTE privileges are granted to PUBLIC for all existing functions, methods, and external stored procedures. This will cause a security exposure for external stored procedures that contain SQL data access which allow users to access SQL objects for which they would not otherwise have privileges. Revoke the privileges by entering the **db2undgp - r** command.
5. Optional: Migrate DB2 Explain tables.
6. Optional: If you recorded configuration settings before migration, you might want to compare pre-migration configuration settings to current configuration settings to verify successful migration. Verify:
   - database configuration parameter settings

- database manger configuration parameter settings
- tablespaces records
- packages records

Note: During migration, the database configuration parameter *maxappls* is set to automatic. If you want it set to a different value, you should update it manually.

Related tasks:

- “Recording your system configuration settings before DB2 migration” on page 20
- “Migrating explain tables” on page 39

Related reference:

- “MIGRATE DATABASE Command” in the *Command Reference*
- “LIST DATABASE DIRECTORY Command” in the *Command Reference*
- “RESTORE DATABASE Command” in the *Command Reference*
- “db2rbind - Rebind all Packages Command” in the *Command Reference*

Migrating databases to a new system

Use this task to migrate DB2 Linux, UNIX, and Windows databases.

Prerequisites:

You must have SYSADM authorization.

You must backup your databases using the **db2 backup database** command.

If you are migrating from DataJoiner, your total log file size must be twice the size of the DataJoiner backup size of your database.

If you are migrating from DataJoiner, DB2 Relational Connect, or DB2 Life Sciences Data Connect, DB2 Information Integrator and DB2 Universal Database for Linux, UNIX, and Windows must be installed on the target system.

Restrictions:

Migration is supported only from:

- DB2 Version 6.x or Version 7.x. (all operating systems supported in Version 6.x and Version 7.x)
- DB2 DataJoiner V2.1.1 (AIX, Windows NT, and Solaris Operating Environment).

Procedure:

To migrate a database to a new system:

1. Migrate the database using the **db2 restore database** command.

   ### DB2 RESTORE DATABASE command

   ```
   ►►──RESTORE──┬─DATABASE─┬──source-database-alias─────────────────────────►
                └─DB───────┘
   ```

```
►─┬──────────────────────────┬────────────────────────────────────────►◄
  └─FROM──directory──device──┘
```

where:

DATABASE *source-database-alias*
: Specifies the alias of the database from which the backup was taken.

FROM *directory/device*
: The fully qualified path name of the directory or device on which the backup image resides. If the FROM parameter is omitted, the default value is the current working directory of the client system. This target directory or device must exist on the database server.

2. If the following error is returned, the database was restored but the database was not migrated.

```
SQL2519N  The database was restored but the restored database was not migrated
   to the current release.  Error "-1704" with tokens "3" is returned.
   SQLSTATE=57011
```

You must complete the following steps to migrate the database:

a. Increase the total size for all log files.
   - If you are migrating from DB2 Version 6 or Version 7, double the values of the `logfilsiz`, `logprimary`, and `logsecond` parameters.
   - If you are migrating from DataJoiner, the total size for all log files should be twice the backup file size of the database. Use the following formula to determine if you have sufficient log file space on your DataJoiner system:

     ```
     ( 4096 X logfilsiz X ( logprimary + logsecond) )  >
          2 X (DataJoiner database backup file size)
     ```

     `4096` is the page size of the log file `logfilsiz`. `logfilsiz`, `logprimary`, and `logsecond` are database configuration parameters.

b. Migrate the database using the **db2 migrate database** command.

DB2 MIGRATE DATABASE command

```
►►──MIGRATE──┬─DATABASE─┬──database-alias──────────────────────────────►
             └─DB───────┘

►─┬─────────────────────────────────────┬──────────────────────────────►◄
  └─USER──username──┬─────────────────┬─┘
                    └─USING──password─┘
```

where:

DATABASE *database-alias*
: Specifies the alias of the database to be migrated to the currently installed version of the database manager.

USER *username*
: Identifies the user name under which the database is to be migrated.

USING *password*
: The password used to authenticate the user name. If the password is omitted, but a user name was specified, the user is prompted to enter it.

c. If the log file size is still not large enough, the following error is returned. You must increase log file size and issue the **db2 migrate database** command again.

```
SQL1704N  Database migration failed.  Reason code "3".
```

d. Once migration is complete reset the `logfilsiz`, `logprimary`, and `logsecond` parameters.

3. Optional: Rebind packages. During database migration, all existing packages are invalidated. After the migration process, each package is rebuilt when it is used for the first time by the DB2 Version 8 database manager. You can run the **db2rbind** command to rebuild all packages stored in the database.
4. Optional: Revoke EXECUTE privileges on external stored procedures that contain SQL data access from PUBLIC. During database migration, EXECUTE privileges are granted to PUBLIC for all existing functions, methods, and external stored procedures. This will cause a security exposure for external stored procedures that contain SQL data access which allow users to access SQL objects for which they would not otherwise have privileges. Revoke the privileges by entering the **db2undgp - r** command.
5. Optional: Migrate DB2 Explain tables.
6. Optional: If you recorded configuration settings before migration, compare premigration configuration settings to the current configuration settings to verify that the migration was successful. You should verify:
   - Database configuration parameter settings
   - Database manger configuration parameter settings
   - Tablespaces records
   - Packages records

   Note: During migration, the database configuration parameter *maxappls* is set to automatic. If you want to set this parameter to a different value, you should update the *maxappls* parameter manually.

Chapter 5. After you migrate

This chapter describes the steps that you need to take after you migrate to DB2 Information Integrator from DataJoiner or from DB2 Version 7. You need to initiate general tasks and tasks that are specific to the data sources that you want to access.

This chapter includes the following topics:

- "Tasks after migrating to DB2 Information Integrator"
- "Accessing DB2 family data sources after migrating to DB2 Information Integrator" on page 52
- "Accessing Informix data sources after migrating to DB2 Information Integrator" on page 54
- "Accessing Microsoft SQL Server data sources after migrating to DB2 Information Integrator" on page 56
- "Accessing ODBC data sources after migrating to DB2 Information Integrator" on page 59
- "Accessing Oracle data sources after migrating to DB2 Information Integrator" on page 60
- "Accessing Sybase data sources after migrating to DB2 Information Integrator" on page 62
- "Accessing Teradata data sources on AIX after migrating to DB2 Information Integrator" on page 64
- "Accessing Documentum data sources after migrating to DB2 Information Integrator" on page 66

Tasks after migrating to DB2 Information Integrator

After you migrate to DB2 Information Integrator, you must complete the following tasks to ensure that you can access your data sources:

- "Restoring DataJoiner configuration settings"
- "Restoring DB2 UDB Version 7 federated systems configuration settings" on page 46
- "Changing the version of a data source" on page 46
- "Converting type-1 indexes to type-2 indexes" on page 46
- "Restoring connections to DB2 family data sources" on page 47
- "Rebinding packages" on page 47
- "Changing the wrapper used to access a data source" on page 47
- "Accessing XML data sources after migrating to DB2 Information Integrator" on page 50

Restoring DataJoiner configuration settings

If you migrated from IBM DataJoiner, you must restore your DataJoiner configuration settings after you migrate your instances and databases.

Locate the djenv.log file that you created before migrating to DB2 Information Integrator. Compare the variables that are listed in the djenv.log file with the variables in the db2dj.ini file, which is located in the $HOME/sqllib/cfg/ directory. If necessary, edit the db2dj.ini file and add any missing variables.

Restoring DB2 UDB Version 7 federated systems configuration settings

If you migrated from DB2 Universal Database Version 7, you must restore the configuration settings after you migrate your instances and databases.

Restore the copy of the db2dj.ini file that you created before you migrated to the $HOME/sqllib/cfg/ directory.

Changing the version of a data source

If you migrated from IBM DataJoiner, you can change the version of your data source to one that is supported by DB2 Information Integrator. You might want to change the data source version if:

- You migrated a DataJoiner server mapping that specified an invalid server version or a server version that DB2 Information Integrator does not support.
- You changed a remote server to a different version.

To change the server version to a version that DB2 Information Integrator supports, issue the ALTER SERVER statement. For example:

```
ALTER SERVER server_name VERSION number
```

server_name is the name of the data source, and *number* is the version number of the data source that is supported by DB2 Information Integrator.

By changing the server version, you ensure that DB2 Information Integrator can access the server name that you set up for the wrapper. The server version that you specify should match the version of your remote server. For example, if you alter the version of an Informix server definition to 9.3, then the remote Informix server should be version 9.3. If the versions do not match, you might get errors when you try to access a nickname in the server definition.

Converting type-1 indexes to type-2 indexes

After you migrate to DB2 Information Integrator, you can convert type-1 indexes to type-2 indexes for a local table in a DB2 Information Integrator database to minimize next-key locking.

1. To determine the index type, issue the following command:

   ```
   db2 load query table <table_name>
   ```

   <table_name> is the name of the table that you want to determine the index type for.

   If the index is type-1, the output for the table is:

   ```
   Tablestate:
    Normal
    Type-1 Indexes
   ```

   If the index is type-2, the output for the table is:

   ```
   Tablestate:
    Normal
   ```

   You do not need to convert a type-2 index.

2. To convert the index to type-2, issue the following command:

```
db2 reorg indexes all for table <table_name> allow [write | read | no]
 access convert
```

Alternatively, you can drop all indexes on the table and create them again.

Restoring connections to DB2 family data sources

If you need to access a remote DB2 family data source after you migrate to DB2 Information Integrator from either DB2 Universal Database Version 7 or from DataJoiner Version 2.1.1, you must use the bind utility to restore a connection from the server to the remote data source. Remote DB2 family data sources include DB2 Universal Database for UNIX and Windows, DB2 Universal Database for z/OS and OS/390, DB2 Universal Database for iSeries, and DB2 Server for VM and VSE.

If you do not restore the connection by using the bind utility, you receive a -805 error message.

To restore the connection to the data source:

1. Connect to the remote DB2 data source.
2. Issue the following commands:

   ```
   cd /home/instancename/sqllib/bnd
   db2 bind @db2cli.lst grant public
   ```

   instancename is the name of the instance that you want to access.
3. Disconnect from the remote data source.

Issue these commands for each remote DB2 family database that you want to access.

Rebinding packages

If you migrated only databases from DataJoiner, you do not need to rebind the packages.

You must rebind packages after you migrate. The command to rebind is

```
rebind package_name
```

If you migrate instances and databases from DataJoiner for AIX or from DataJoiner for Solaris, and do not rebind the packages you might receive an error message when you run the packages.

Changing the wrapper used to access a data source

DB2 Information Integrator Version 8 supported more than one wrapper for some data sources such as the SQLNET and NET8 wrappers for Oracle data sources and the DBLIB and CTLIB wrappers for Sybase data sources. For DB2 Information Integrator Version 8.2 and later, the SQLNET wrapper is not supported for Oracle data sources and the DBLIB wrapper is not supported for Sybase data sources.

The CT-Library protocol supports dynamic prepare and execute of SQL statements. Dynamic prepare and execute functionality allows applications that use the CT-Library protocol to prepare a statement one time and execute it many times by using different input parameters. Preparing a statement one time eliminates the need to recompile the statement for each input parameter change. Although the DB2 application might not take advantage of dynamic SQL, federated query processing uses dynamic SQL exclusively.

Procedure:

If you want to change the wrapper that you use to access a data source, you can make this change after you migrate.

To change the wrapper that you use:

1. Run the **db2look** command. You must run the **db2look** command before you drop the wrapper.
2. Drop the wrapper.
3. Register the new wrapper with the federated database.

When you drop a wrapper, the other objects dependent upon that wrapper are affected:

- All server definitions, user-defined functions mappings, and user-defined data type mappings that are dependent upon the wrapper are also dropped.
- Dropping all server definitions that are dependent on that wrapper affects the objects dependent on those server definitions. All user-defined function mappings, user-defined data type mappings, and user mappings that are dependent on the dropped server definitions are also dropped.
- All nicknames that are dependent on the dropped server definitions are also dropped. Dropping the nicknames that are dependent on the server definitions affects the objects dependent on those nicknames:
  – Any index specifications that are dependent on the dropped nicknames are also dropped.
  – Any federated views that are dependent on the dropped nicknames are marked inoperative.
- All applications that are dependent on the dropped objects and inoperative views are invalidated.

You can change the wrapper by either reissuing all the SQL statements manually, or you can use the **db2look** command to obtain the correct syntax necessary for each data source configuration statement.

The **db2look** command generates DDL statements from the metadata in the DB2 catalogs. If you use the **db2look** command, you can edit the statements with your specific information.

To change the wrapper for a data source:

1. Issue the **db2look** command to generate the necessary DDL statements from DB2 UDB. For example:
   ```
   db2look -d database_name -wrapper wrapper_name -e -o federated.sql
   ```
   This command creates an output file named `federated.sql` that contains the DDL statements for all the federated objects of the wrapper that you specify.
2. Edit the SQL statements in the `federated.sql` file to specify the new wrapper information:
   - If necessary, change the order of type mappings, nicknames, and transparent DDL statements. The correct order is:
     a. Create the wrapper.
     b. Create the server definitions.
     c. Create the user mappings.

d. Create the nicknames, including nicknames for remote tables that were created using transparent DDL.

e. Alter the nicknames to override the default mappings.

- Change the wrapper name. The wrapper name is on both the CREATE WRAPPER and CREATE SERVER statement. Use the default wrapper name so that you do not have to specify the library file name. If you do not use the default wrapper name, ensure that the wrapper library name is correct. The wrapper libraries are in the file system.

  On UNIX systems, the wrapper libraries are in the /sqllib/lib directory for the instance owner. On Windows systems, the wrapper libraries are in the \SQLLIB\BIN directory. The CREATE WRAPPER statement must specify the base library of the wrapper, which is the library name. The base library name does not include, ’F’ (fenced) or ’U’ (unfenced).
- If the AUTHORIZATION and PASSWORD parameters are used with the CREATE SERVER statement, the values that are associated with these parameters are empty in the SQL generated by the **db2look** command. You must add the correct values for these parameters.
- Manually enclose the remote data types in the CREATE TYPE MAPPING statements in double quotation marks (″) if you want the characters to be case sensitive. The remote data types in the generated CREATE TYPE MAPPING statements are not enclosed with double quotation marks.
- Some data type mappings for certain data sources in DB2 Information Integrator, Version 2 are not the same as they were in DB2 Information Integrator, Version 1. You can now use empty parentheses for parameterized data types. When you use **db2look** against a migrated database, empty parentheses are added not only against parameterized data types that do not have length but also against non-parameterized data types. If you have non-parameterized data types, remove the empty parentheses.
- Insert the correct passwords in the CREATE USER MAPPING statements.
- Check the SQL statements for any local views that are created by using the WITH CHECK OPTION clause. This clause is no longer supported. Remove the SQL statements that create these local views from the `federated.sql` file.

3. If the wrapper that you plan to drop accesses remote tables that you created by using transparent DDL, the nicknames for these remote tables are dropped when you drop the wrapper. You must create nicknames again for the remote tables. You can add the SQL for these nicknames to the `federated.sql` file.
4. Drop the wrapper.
5. By default, GRANT authorization is associated with the user ID of the user that is connected to the database. This user ID appears in the federated database system catalog. If you want a different user ID to appear in the catalog, you must connect to the database with the correct user ID before the GRANT statements are issued. Then issue the GRANT statements.
6. Add or modify the db2dj.ini file to include the client paths by performing the following steps:
   a. Before you begin to access data sources after migration, you must issue the djxlink script to create new libraries in the DB2 base directories and the **db2iupdt** command to update your instances.
   b. Recycle DB2 Universal Database by stopping and starting it.
   c. Modify db2dj.ini to include the client paths for the data source that you are using.
7. Issue the SQL statements in the `federated.sql` file to configure access to the data source. For example:

```
db2 -tvf federated.sql
```

8. Bind any application packages that were invalidated when you dropped the wrapper. Packages that can be invalidated contain static SQL that references nicknames that were dropped and recreated.
9. Recreate any federated views that were invalidated when you dropped the wrapper. Views that were invalidated are the views that reference nicknames that were dropped and recreated.

Accessing XML data sources after migrating to DB2 Information Integrator

The default column names generated for XML nicknames by the discover feature in the DB2 Control Center have changed in DB2 UDB Information Integrator, Version 8.2. If you drop a nickname for an XML data source that was created in a release prior to Version 8.2 and then create the nickname again in Version 8.2, the default column names will be different. You might need to alter the nickname column names or alter other objects, such as your SQL queries, that refer to the old column names.

Related tasks:

- "Altering a nickname" in the *Federated Systems Guide*
- "Accessing DB2 family data sources after migrating to DB2 Information Integrator" on page 52
- "Accessing Documentum data sources after migrating to DB2 Information Integrator" on page 66
- "Accessing Informix data sources after migrating to DB2 Information Integrator" on page 54
- "Accessing Microsoft SQL Server data sources after migrating to DB2 Information Integrator" on page 56
- "Accessing ODBC data sources after migrating to DB2 Information Integrator" on page 59
- "Accessing Oracle data sources after migrating to DB2 Information Integrator" on page 60
- "Accessing Sybase data sources after migrating to DB2 Information Integrator" on page 62
- "Accessing Teradata data sources on AIX after migrating to DB2 Information Integrator" on page 64

Related reference:

- "db2iupdt - Update Instances Command" in the *Command Reference*
- "Altering a local type for a data source object – examples" in the *Federated Systems Guide*

Allowing write operations on large object data types after migrating to DB2 Information Integrator

After you migrate to DB2 Information Integrator, the large object data types of your remote data sources are mapped to CLOB, BLOB, or DBCLOB data types, depending on the data source that you are accessing.

With the exception of the Oracle NET8 wrapper, federated servers cannot write to DB2 CLOB , BLOB, or DBCLOB data types in nicknames.

If you need to write to these large object data types after you migrate to DB2 Information Integrator, you can alter the local CLOB and BLOB data type columns to VARCHAR and VARCHAR FOR BIT DATA data types.

You can alter CLOB and BLOB data types to VARCHAR data types with lengths up to 32672 characters. If the CLOB, BLOB, or DBCLOB data type contains data that is larger than 32672 characters, the data is truncated.

To enable replication of large object data types, you must allow write operations on these large object data type columns. DB2 replication requires insert and update privileges on the column values that it replicates to target tables.

Procedure:

To allow write operations on large object data types, issue the ALTER NICKNAME statement.

For example:

```
ALTER NICKNAME nickname1
      ALTER COLUMN column1
       LOCAL TYPE VARCHAR(32672)
```

You might receive SQL statement warnings when you attempt to alter large object data types depending on the size of the data and the data source that you are accessing.

You can allow write operations on and enable replication of large object data types for the following data sources:

DB2 UDB
The DRDA wrapper maps remote LONG VARCHAR and LONG VARCHAR FOR BIT DATA data types by default to local CLOB, BLOB, and DBCLOB data types. You can alter these local CLOB, BLOB, and DBCLOB data type columns to VARCHAR and VARCHAR FOR BIT data types with lengths up to 32672 characters.

Oracle The NET8 wrapper maps remote LONG and LONG RAW data types by default to local CLOB, BLOB, and DBCLOB data types. You can alter these local CLOB, BLOB, and DBCLOB data type columns to VARCHAR and VARCHAR FOR BIT DATA data types with lengths up to 32672 characters.

Informix
The INFORMIX wrapper maps TEXT data types and BYTE data types by default to local CLOB, BLOB, and DBCLOB data types. You can alter these local CLOB, BLOB, and DBCLOB data type columns to VARCHAR and VARCHAR FOR BIT DATA data types with lengths up to 32672 characters.

Sybase
The CTLIB wrapper maps TEXT data types and IMAGE data types by default to local CLOB, BLOB, and DBCLOB data types. You can alter these local CLOB, BLOB, and DBCLOB data type columns to VARCHAR and VARCHAR FOR BIT DATA data types with lengths up to 32672 characters.

Microsoft SQL Server
The DJXMSSQL3 wrapper and the MSSQLODBC3 wrapper map TEXT data types and IMAGE data types to local CLOB, BLOB, and DBCLOB data types. You can alter these local CLOB, BLOB, and DBCLOB data type columns to VARCHAR and VARCHAR FOR BIT DATA data types with lengths that are less than 32672 characters.

Teradata

Remote CHAR and VARCHAR data types with lengths that are greater than 32672 character are mapped by default to local CLOB, BLOB, and DBCLOB data types. Remote BYTE and VARBYTE data types with lengths that are greater than 32672 character are mapped by default to local BLOB data types. You can alter these local CLOB, BLOB, and DBCLOB data type columns to VARCHAR and VARCHAR FOR BIT DATA data types with lengths up to 32672 characters.

For example, a remote table that is named EMP is on a remote DB2 Universal Database data source. The EMP table contains a column named EMP_INFO with a remote data type of LONG VARCHAR. To use the remote table, create a nickname named EMPLOYEE that references the DB2 table EMP.

When you create a nickname for this remote table, the DRDA wrapper maps the data type of the EMP_INFO column by default to a local CLOB data type.

To allow write operations on and to enable replication of the local data in the EMP_INFO column, alter the data type of this column from CLOB to VARCHAR. For example:

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

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*

Accessing DB2 family data sources after migrating to DB2 Information Integrator

Use this procedure after you migrate to DB2 Information Integrator from IBM DataJoiner or from DB2 Universal Database Version 7.

Prerequisites:

Record the server names and the database alias names for the databases that you accessed before you migrated to DB2 Information Integrator.

Although the DB2 Information Integrator Version 8 DRDA wrapper ignores the NODE server option, a node entry must be catalogued in the node directory to catalog remote databases in the database directory.

DataJoiner and DB2 UDB Version 7 federated servers use the NODE option to match a local node directory entry with the DBNAME option. These options match a database alias, location, or remote database name at the remote data source. DB2 Information Integrator Version 8 federated servers only use the DBNAME option, which matches a database alias in the local database directory.

Procedure:

To access DB2 family data sources after you migrate to DB2 Information Integrator:

1. Issue the **list database directory** command to display the names of the databases that are defined in the federated server system database directory.

2. Issue the **list node directory** command to display the contents of the node directory.
3. Verify that the remote node names that displayed when you issued the **list database directory** command are valid node entries in the node directory.
4. If you migrated only the database, issue the **catalog tcpip node** command. This command adds the TCP/IP node entry to the node directory so that you can access the remote node.

   For example:

   ```
   CATALOG TCPIP NODE NODENAME1 REMOTE stlxxx1.abc.ibm.com SERVER 10110
   ```

   NODENAME1 is the name of the remote node. *stlxxx1.abc.ibm.com* is the host name where the target database resides. *10110* is the port number of the server database manager instance.
5. Catalog the database, whether you migrated instances and the database or whether you migrated only the database:
   a. Issue the **catalog database** command to catalog and store remote database information in the federated system database directory.

      For example:

      ```
      CATALOG DATABASE DB_NAME AS ALIAS1 AT NODE NODENAME1
      ```

      DB_NAME is the name of the remote database that you are cataloging in the federated server system database directory. *ALIAS1* is the alias for the database that you are cataloging. *NODENAME1* is the name of the node where this remote database resides.

      The name of the node that you specify must match the node name that you added in the **catalog tcpip node** command.

   b. If the remote database name is longer than eight characters, issue the **catalog dcs database** command. This command creates a DCS directory entry.

      For example:

      ```
      CATALOG DCS DATABASE ALIAS2 AS DB2_DATABASE_NAME
      ```

      The name of the alias, *ALIAS2* in this example, must match the name of the database alias that you are cataloging. *DB2_DATABASE_NAME* is the name of the target host database.

      Database names for DB2 UDB for z/OS and DB2 UDB for iSeries data sources are typically longer than eight characters.
6. Alter the server definition in the federated database system catalog to register the database name:
   a. Check the SYSCAT.SERVERS catalog view for the name of the server that you migrated.

   b. Issue the ALTER SERVER statement.

      For example:

      ```
      ALTER SERVER alias_server_name OPTIONS (SET DBNAME 'ALIAS1')
      ```

      alias_server_name is the alias for the remote server in the catalog view. *ALIAS1* is the alias for the database that you cataloged.
7. Verify that the migration was successful:
   a. Connect to the database.

   b. Issue SELECT, INSERT, UPDATE, and DELETE statements on the nicknames to verify that you can view and modify the data.

You can also open a pass-through session to the remote database, and issue a SELECT statement on a remote table.

For complete information on migrating your DB2 replication environment, see the *IBM DB2 Information Integrator Migration Guide: Migrating to SQL Replication Version 8* at http://www.ibm.com/software/data/integration/db2ii/support.html.

Related tasks:

- "Adding DB2 family data sources to a federated server" in the *IBM DB2 Information Integrator Data Source Configuration Guide*
- "Recording database information before migrating" on page 21

Related reference:

- "DataJoiner terms and their DB2 Information Integrator equivalents" on page 69
- "Migration reference information for DB2 family data sources" on page 71
- "Changes to default data type mappings for DB2 for z/OS and OS/390 data sources" on page 72
- "Changes to default data type mappings for DB2 for iSeries (AS/400) data sources" on page 73
- "Changes to default data type mappings for DB2 for Linux, UNIX, and Windows data sources" on page 74
- "Changes to default data type mappings for DB2 for VM and VSE data sources" on page 75

Accessing Informix data sources after migrating to DB2 Information Integrator

Use this procedure if you migrated the instance and database or only the database.

If you migrated only the database, follow all of the steps in this procedure.

If you migrated the instance and database on UNIX operating systems, you might need to perform only some of the steps in this procedure:

- Set the INFORMIXDIR environment variable.
- Run the djxlink script or the djxlinkInformix script.
- Recycle the DB2 UDB instance.
- Verify the migration.

If you migrated the instance and database, you might need to perform these steps depending on how you set the variables on your system and whether the djxlink script or the djxlinkInformix script ran when you installed DB2 Information Integrator.

Procedure:

To access Informix data sources after you migrate to DB2 Information Integrator from DataJoiner or from DB2 UDB Version 7:

1. Verify the connection to the Informix server. The connection information is in the sqlhosts file:
   - On UNIX operating systems, the configuration information is stored in the $INFORMIXDIR/etc/sqlhosts file.

- On Windows operating systems, the sqlhosts file is in the Windows registry: HKEY_LOCAL_MACHINE\SOFTWARE\Informix\SQLHOSTS.

2. Check the settings of the Informix environment variables:
   a. Find the db2dj.ini file:
      - On UNIX operating systems, the db2dj.ini file is in the /sqllib/cfg subdirectory.
      - On Windows operating systems, the db2dj.ini file is in the \sqllib\cfg subdirectory.

   b. Set the environment variables in the db2dj.ini file as necessary. These variables are:

      INFORMIXDIR

      The directory path where the Informix Client SDK software is installed. For example:

      ```
      INFORMIXDIR=/informix/csdk
      ```

      INFORMIXSERVER

      The name of the default Informix server. For example:

      ```
      INFORMIXSERVER=inf93
      ```

      The INFORMIXSERVER setting must be a valid entry in the sqlhosts file (on UNIX operating systems) or the SQLHOSTS registry key (on Windows operating systems). To get a value for the INFORMIXSERVER variable, look in the sqlhosts file. Select one of the dbservername values. The dbservername is the first value in each entry in the sqlhosts file. Although the Informix wrapper does not use the value of this variable, the Informix client requires that this variable be set.

      INFORMIXSQLHOSTS

      The path where the sqlhosts file resides. Set the INFORMIXSQLHOSTS environment variable only if the path that you are using for the sqlhosts file is different from the default path.
      - On UNIX operating systems, the default path is $INFORMIXDIR/etc.
      - On Windows operating systems, if the SQLHOSTS registry key does not reside on the local computer, then the INFORMIXSQLHOSTS value is the name of the Windows computer that stores the registry.

      The following example shows you how to set this variable to another path:

      ```
      INFORMIXSQLHOSTS=/informix/csdk/etc/my_sqlhosts
      ```

   c. Optional: Set the following environment variables if you want to change the default values for Informix code page conversion:
      - Set the CLIENT_LOCALE environment variable to the Informix locale. For example:

        CLIENT_LOCALE=*informix_client_locale_value*
      - Set the DB_LOCALE environment variable to the name of the Informix database locale. For example:

        DB_LOCALE=*informix_db_locale_value*
      - Set this environment variable if the Informix database uses a different code page than your client locale, and you want Informix to perform conversions between the two code pages.

- Set the DBNLS environment variable to 1 to ensure that the Informix software verifies that the DB_LOCALE setting matches the actual locale of the Informix database. For example:

  ```
  DBNLS=1
  ```

3. If the djxlink script or the djxlinkInformix script did not run when you installed DB2 Information Integrator on UNIX, run the djxlink script or the djxlinkInformix script. For more information on running the djxlink script, see the *IBM DB2 Information Integrator Installation Guide for Linux, UNIX, and Windows*.
4. If you changed any environment variable in the db2dj.ini file in step 2 or if you ran the djxlink script or the djxlinkInformix script in step 3, recycle the DB2 instance:
   a. Issue the **db2stop** command.
   b. If you ran the djxlink script or the djxlinkInformix script, issue the **db2iupdt** command.
   c. Issue the **db2start** command.

   By recycling the DB2 instance, you ensure that the environment variable changes are properly set on the federated server.
5. Verify that the migration was successful:
   a. Connect to the database.
   b. Issue SELECT, INSERT, UPDATE, and DELETE statements on the nicknames to verify that you can view and modify the data.

   You can also open a pass-through session to the remote database, and issue a SELECT statement directly on a remote table.

For complete information on migrating your DB2 replication environment, see the *IBM DB2 Information Integrator Migration Guide* at http://www.ibm.com/software/data/integration/db2ii/support.html.

Related tasks:

- "Adding Informix data sources to federated servers" in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Related reference:

- "db2iupdt - Update Instances Command" in the *Command Reference*
- "DataJoiner terms and their DB2 Information Integrator equivalents" on page 69
- "Migration reference information for Informix data sources" on page 78

Accessing Microsoft SQL Server data sources after migrating to DB2 Information Integrator

Use this procedure if you migrated the instance and database or only the database. If you migrated only the database, follow all of the steps in this procedure. If you migrated the instance and database on UNIX operating systems, you might need to initiate only some of the steps in this procedure. You might need to initiate the following steps:

- Run the djxlink script or the djxlinkMssql script.
- Recycle the DB2 instance.
- Verify the migration.

If you migrated the instance and database, you might need to initiate these steps if the djxlink script or the djxlinkMssql script did not run when you installed DB2 Information Integrator.

Data type mappings that are migrated from DB2 Information Integrator, Version 1: Some data type mappings for Microsoft SQL Server data sources in DB2 Information Integrator, Version 2 are not the same as they were in DB2 Information Integrator, Version 1. The following data type mappings in a Unicode federated databases have changed:

- NCHAR
- NVARCHAR
- NTEXT

All of the data type mappings from DB2 Information Integrator, Version 1 are migrated to DB2 Information Integrator, Version 2 as they currently are in DB2 Information Integrator, Version 1. However, if you drop a nickname for the Microsoft SQL Server data source that was created in DB2 Information Integrator, Version 1 and then recreate that nickname using DB2 Information Integrator, Version 2, the data type mapping changes to the new type mapping. If you need to use the data type mapping for NCHAR, NVARCHAR, or NTEXT from DB2 Information Integrator, Version 1, you must create forward type mapping or alter the column data type of the nickname.

Procedure:

To access Microsoft SQL Server data sources after you migrate to DB2 Information Integrator from DataJoiner or from DB2 UDB Version 7:

1. Check the settings of the environment variables for Microsoft SQL Server data sources:
   a. Locate the db2dj.ini file:
      - On UNIX operating systems, the db2dj.ini file is in the /sqllib/cfg subdirectory.
      - On Windows operating systems, the db2dj.ini file is in the \sqllib\cfg subdirectory.

   b. Set the environment variables in the db2dj.ini file as necessary. These variables are:

      ODBCINI

      Full path to the ODBC data source configuration file (.odbc.ini). The .odbc.ini file contains the list of ODBC data sources that can be accessed, the ODBC driver library to be used for each data source, and connection information for each data source such as hostname/ipaddress and listener port. For example:

      ```
      ODBCINI=/home/db2inst1/.odbc.ini
      ```

      DJX_ODBC_LIBRARY_PATH

      The directory path where the ODBC library files reside. For example:

      ```
      DJX_ODBC_LIBRARY_PATH=ODBC_home_directory/lib
      ```

      ODBC_home_directory is the directory path where the ODBC driver is installed.

      LD_LIBRARY_PATH

      On the Solaris Operating Environment, the directory path where the ODBC library files reside. For example:

```
LD_LIBRARY_PATH=ODBC_home_directory/lib
```

SHLIB_PATH
On HP-UX operating systems, the directory path where the ODBC library files reside. For example:

```
SHLIB_PATH=ODBC_home_directory/lib
```

2. On UNIX operating systems, set the DB2 profile registry variables. To set the variables, issue the **db2set** command.

   For example:

   ```
   db2set DB2LIBPATH=ODBC_home_directory/lib
   db2set DB2ENVLIST=LIBPATH
   ```

   ODBC_home_directory is the directory path where the ODBC driver is installed.
3. Check the Microsoft SQL Server configuration:
   - On UNIX operating systems, verify that the /opt/odbc/odbc.ini file is properly set up. Use the demoodbc tool provided with DataDirect Connect to test connectivity to the Microsoft SQL Server data source.
   - On Windows operating systems, verify that the ODBC System DSN is properly set up. Use a query tool that can access ODBC data sources, or use the connectivity test in the Microsoft SQL Server ODBC driver data source configuration window to test connectivity to the Microsoft SQL Server data source.
4. Verify that the migration was successful:
   a. Connect to the database.
   b. Issue SELECT, INSERT, UPDATE, and DELETE statements on the nicknames to verify that you can view and modify the data.

   You can also open a pass-through session to the remote database, and issue a SELECT statement on a remote table.

For complete information on migrating your DB2 replication environment, see the *IBM DB2 Information Integrator Migration Guide: Migrating to SQL Replication Version 8* at http://www.ibm.com/software/data/integration/db2ii/support.html.

Related tasks:

- “Adding Microsoft SQL Server data sources to a federated server” in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Related reference:

- “db2iupdt - Update Instances Command” in the *Command Reference*
- “Unicode default forward data type mappings - Microsoft SQL Server wrapper” in the *Federated Systems Guide*
- “DataJoiner terms and their DB2 Information Integrator equivalents” on page 69
- “Migration reference information for Microsoft SQL Server data sources” on page 80

Accessing ODBC data sources after migrating to DB2 Information Integrator

The ODBC wrapper can be used to access a data source if there is not an available data source-specific wrapper, such as NET8 for Oracle data sources, for the data source that you want to access.

Use this procedure if you migrated the instance and database or only the database.

Use this procedure only if you are accessing ODBC data sources on UNIX operating systems. No post-migration steps are required if you are accessing ODBC data sources on Windows operating systems.

Data type mappings that are migrated from DB2 Information Integrator, Version 1: Some data type mappings for ODBC data sources in DB2 Information Integrator, Version 2 are not the same as they were in DB2 Information Integrator, Version 1. The following data type mappings in a Unicode federated databases have changed:

- SQL_WCHAR
- SQL_WVARCHAR
- SQL_WLONGVARCHAR

All of the data type mappings from DB2 Information Integrator, Version 1 are migrated to DB2 Information Integrator, Version 2 as they currently are in DB2 Information Integrator, Version 1. However, if you drop a nickname for the ODBC data source that was created in DB2 Information Integrator, Version 1 and then recreate that nickname using DB2 Information Integrator, Version 2, the data type mapping changes to the new type mapping. If you need to use the data type mapping for SQL_WCHAR, SQL_WVARCHAR, or SQL_WLONGVARCHAR from DB2 Information Integrator, Version 1, you must create forward type mapping or alter the column data type of the nickname.

Prerequisites:

If you use transparent DDL to create remote tables on ODBC data sources through DataJoiner, you might receive errors with SELECT statements after you migrate to DB2 Information Integrator. The DataJoiner DATE data type is mapped to the ODBC DATETIME data type when you create a remote ODBC table through DataJoiner. To avoid these errors, change the local data type from DATE to TIMESTAMP in the federated database systems catalog after you migrate.

Procedure:

To access ODBC data sources after you migrate to DB2 Information Integrator from DataJoiner or from DB2 Universal Database Version 7:

1. If you migrated only the database, set the appropriate environment variables in the sqllib/cfg/db2dj.ini file.

   For example, if you are configuring a data source by using Data Direct Connect for ODBC, set the following environment variables:

   - ODBCINI
   - LD_LIBRARY_PATH (on the Solaris Operating Environment only)
   - SHLIB_PATH (on HP-UX operating systems only)

   On AIX, issue the **db2set** command to set the DB2LIBPATH environment variable.

2. Issue the ALTER WRAPPER statement to specify the MODULE option. You must specify the MODULE option to enable the ODBC wrapper on UNIX operating systems. For AIX operating systems, enclose the shared object member name in parentheses. For example:

   ```
   ALTER WRAPPER ODBC
    OPTIONS ( add MODULE '/opt/odbc/lib/libodbc.a(odbc.so)' );
   ```

   In this example, *libdb2rcodbc.a* is the file that contains the wrapper module. */opt/odbc/lib/libodbc.a* is the fully qualified path of the directory that contains the ODBC Driver Manager library on AIX. *(odbc.so)* is the appropriate member in the library. The wrapper option *MODULE* specifies the full path to the ODBC driver manager library.
3. Verify that the migration was successful:
   a. Connect to the database.
   b. Issue SELECT, INSERT, UPDATE, and DELETE statements on the nicknames to verify that you can view and modify the data.

   You can also open a pass-through session to the remote database, and issue a SELECT statement on a remote table.

Related tasks:

- "Adding ODBC data sources to a federated server" in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Related reference:

- "db2look - DB2 Statistics and DDL Extraction Tool Command" in the *Command Reference*
- "Unicode default forward data type mappings - ODBC wrapper" in the *Federated Systems Guide*
- "DataJoiner terms and their DB2 Information Integrator equivalents" on page 69
- "Migration reference information for ODBC data sources" on page 82

Accessing Oracle data sources after migrating to DB2 Information Integrator

Use this procedure if you migrated the instance and database or only the database.

If you migrated only the database, follow all of the steps in this procedure.

If you migrated the instance and database on UNIX operating systems, you might need to perform only some of the steps in this procedure. The steps that you might need to perform are:

- Running the djxlink script or the djxlinkOracle script
- Recycling the DB2 instance
- Verifying the migration

If you migrated the instance and database, you might need to perform these steps if the djxlink script or the djxlinkOracle script did not run when you installed DB2 Information Integrator.

Procedure:

To access Oracle data sources using the NET8 wrapper or the SQLNET wrapper after you migrate to DB2 Information Integrator from DataJoiner or from DB2 UDB Version 7:

1. Check the settings of the Oracle environment variables:
   a. Locate the db2dj.ini file:
      - On UNIX operating systems, the db2dj.ini file is in the /sqllib/cfg subdirectory.
      - On Windows operating systems, the db2dj.ini file is in the \sqllib\cfg subdirectory.

   b. Set the environment variables in the db2dj.ini file as necessary. These variables are:

      ORACLE_HOME
      The fully qualified directory path where the Oracle client software is installed. For example:
      ```
      ORACLE_HOME=/usr/oracle/8.1.7
      ```
      /usr/oracle/8.1.7 is fully qualified path of the Oracle home directory. This environment variable is required.

      ORACLE_BASE
      The root of the Oracle client directory tree. For example:
      ```
      ORACLE_BASE=oracle_root_directory
      ```

      ORA_NLS
      The directory where location-specific data is stored for the version of Oracle that you are using. For example:
      ```
      ORACLE_NLS33=oracle_root_directory/ocommon/nls/admin/data
      ```
      In this example, the location-specific data is stored in the *oracle_root_directory*/ocommon/nls/admin/data directory on UNIX federated servers that access Oracle 8.1 data sources.

      TNS_ADMIN
      The directory where the tnsnames.ora file is located. Set this parameter only if the location of the tnsnames.ora is different from the default location.
      - On UNIX operating systems, the default location for the tnsnames.ora file is the /etcdirectory. This parameter is required only if the tnsnames.ora file that is used is not in the $ORACLE_HOME/network/admin directory or the /etc directory.
      - On Windows operating systems, the default location for the tnsnames.ora file is the \NETWORK\ADMIN directory.

      For example:
      ```
      TNS_ADMIN=tnsnames.ora_directory
      ```

2. For UNIX operating systems only: Update the .profile file of the DB2 instance with the Oracle environment variable by issuing the following command:
   ```
   export ORACLE_HOME=oracle_home_directory
   ```
   oracle_home_directory is the directory where the Oracle client software is installed.

3. Perform the DB2 instance .profile by entering:
   ```
   . .profile
   ```

4. If the djxlink script or the djxlinkOracle script did not run when you installed DB2 Information Integrator on UNIX operating systems, run the djxlink script or the djxlinkOracle script.
5. If you changed any environment setting in the preceding steps or if you ran only the djxlink script or the djxlinkOracle script in step 4, recycle the DB2 instance:
   a. Issue the **db2stop** command.
   b. If you ran the djxlink script or the djxlinkOracle script, issue the **db2iupdt** command.
   c. Issue the **db2start** command.

   By recycling the DB2 instance, you ensure that the environment variable changes are properly set on the federated server.
6. Set up and test the Oracle client configuration files (tnsnames.ora and sqlnet.ora). Use Oracle sqlplus to test connectivity to the Oracle server.
7. Verify that the migration was successful:
   a. Connect to the database.
   b. Issue SELECT, INSERT, UPDATE, and DELETE statements on the nicknames to verify that you can view and modify the data.

   You can also open a pass-through session to the remote database, and issue a SELECT statement on a remote table.

For complete information on migrating your DB2 replication environment, see the *IBM DB2 Information Integrator Migration Guide: Migrating to SQL Replication Version 8* at http://www.ibm.com/software/data/integration/db2ii/support.html.

Related tasks:

- "Adding Oracle data sources to a federated server" in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Related reference:

- "db2iupdt - Update Instances Command" in the *Command Reference*
- "DataJoiner terms and their DB2 Information Integrator equivalents" on page 69
- "Migration reference information for Oracle data sources" on page 76
- "Changes to default data type mappings for Oracle SQLNET" on page 76
- "Changes to default data type mappings for Oracle NET8" on page 77

Accessing Sybase data sources after migrating to DB2 Information Integrator

Use this procedure if you migrated the instance and database or only the database.

If you migrated only the database, follow all of the steps in this procedure.

If you migrated the instance and database on UNIX systems, you must do step 1 to restore the interfaces file, and you might need to do additional steps.

If you migrated the instance and database, you must recycle the instance in step 6 and verify the migration in step 7 after you set the SYBASE_OCS environment variable or after you run the djxlink script or the djxlinkSybase script.

Procedure:

To access Sybase data sources after you migrate to DB2 Information Integrator from DataJoiner or from DB2 Universal Database Version 7:

1. From the federated database instance, restore the copy of the interfaces file that you made before you migrated to DB2 Information Integrator. Save this restored file in the $HOME/sqllib directory.
2. Check the settings of the environment variables for Sybase data sources:
   a. Locate the db2dj.ini file:
      - On UNIX systems, the db2dj.ini file is in the /sqllib/cfg subdirectory.
      - On Windows operating systems, the db2dj.ini file is in the \sqllib\cfg subdirectory.

   b. Set the environment variables in the db2dj.ini file as necessary. These variables are:

      SYBASE
      The directory path where the Sybase Open Client software is installed. For example:
      ```
      SYBASE=/sybase
      ```
      SYBASE_OCS
      If you are using Sybase Open Client Version 12 or later, set the SYBASE_OCS environment variable to the name of the OCS directory. For example:
      ```
      SYBASE_OCS=OCS-12_5
      ```
3. Update the .profile file of the DB2 instance with the SYBASE and SYBASE_OCS environment variables by issuing the following commands:
   ```
   export SYBASE=sybase_home_directory
   export SYBASE_OCS=sybase_ocs_directory
   export PATH=$SYBASE/$SYBASE_OCS/bin:$PATH
   ```
   sybase_home_directory is the directory where the Sybase Open Client software is installed. *sybase_ocs_directory* is the name of the OCS directory and is required if you are using Sybase Open Client Version 12 or later.
4. Initiate the DB2 instance .profile by entering:
   ```
   . .profile
   ```
5. If the djxlink script or the djxlinkSybase script did not run when you installed DB2 Information Integrator on UNIX systems, run the djxlink script or the djxlinkSybase script.
6. If you changed any environment setting in the preceding steps, or if you ran only the djxlink script or the djxlinkSybase script in step 5, recycle the DB2 instance:
   a. Issue the **db2stop** command.
   b. If you ran the djxlink script or the djxlinkSybase script, issue the **db2iupdt** command.
   c. Issue the **db2start** command.

   By recycling the DB2 instance, you ensure that the environment variable changes are properly set on the federated server.
7. Verify that the migration was successful:
   a. Connect to the database.
   b. Issue SELECT, INSERT, UPDATE, and DELETE statements on the nicknames to verify that you can view and modify the data.

You can also open a pass-through session to the remote database, and issue a SELECT statement on a remote table.

For complete information on migrating your DB2 replication environment, see the *IBM DB2 Information Integrator Migration Guide: Migrating to SQL Replication Version 8* at http://www.ibm.com/software/data/integration/db2ii/support.html.

Related tasks:

- "Adding Sybase data sources to a federated server" in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Related reference:

- "db2iupdt - Update Instances Command" in the *Command Reference*
- "DataJoiner terms and their DB2 Information Integrator equivalents" on page 69
- "Migration reference information for Sybase data sources" on page 79

Accessing Teradata data sources on AIX after migrating to DB2 Information Integrator

Use this procedure if you migrated the instance and database or only the database.

If you migrated only the database, follow all of the steps in this procedure.

If you migrated the instance and the database, you might need to perform only some of the steps in this procedure. The steps that you might need to perform are:

- Running the djxlink script or the djxlinkTeradata script
- Recycling the DB2 instance
- Verifying the migration

If you migrated the instance and database, you might need to perform these steps if the djxlink script or the djxlinkTeradata script did not run when you installed DB2 Information Integrator.

Prerequisites:

If you already use the Teradata wrapper, you can upgrade to Teradata release V2R5 with either of the following procedures:

- Modifying your existing Teradata wrapper configuration.
- Recreating your entire Teradata wrapper configuration of servers, user mappings, type mappings, function mappings, and nicknames.

To modify your existing Teradata wrapper configuration:

1. Change the Teradata server product version to 2.5 on all of your Teradata servers by using the ALTER SERVER statement:
   ```
   ALTER SERVER MYTERASERVER VERSION 2.5
   ```
2. Remove the TUPLE_ID_TYPE nickname option on all of your Teradata nicknames by using the ALTER NICKNAME statement:
   ```
   ALTER NICKNAME TERANICKNAME OPTIONS (DROP TUPLE_ID_TYPE)
   ```

   To get a list of all nicknames that need to be changed, use the following SQL statement to return nickname object names, server object names, and server versions for all nicknames that are associated with a TERADATA server type:

```
SELECT T.tabname, S.servername, S.serverversion
  FROM syscat.servers AS S, syscat.taboptions AS T
   WHERE T.option = 'SERVER' AND S.servername = T.setting
   AND S.servertype = 'TERADATA' AND S.serverversion
   IN ( '2.3', '2.4', '3', '4' )
```

To recreate your entire Teradata wrapper configuration:

1. Use the **db2look** command to get the correct syntax for each data source configuration statement. The **db2look** command generates DDL statements from the metadata in the DB2 catalog.
2. In the **db2look** command output file, change all references to the Teradata product version to `2.5` for all server objects.
3. Drop the Teradata wrapper.

   When you drop a wrapper, all server definitions, user-defined function mappings, and user-defined data type mappings that are dependent on the wrapper are dropped. All user-defined function mappings, nicknames, user-defined data type mappings, and user mappings that are dependent on the dropped server definitions are also dropped. All packages dependent on the dropped objects and inoperative views are invalidated.
4. If necessary, drop the type and function mappings that are associated with the wrapper you dropped. The DROP statement drops only the objects that are explicitly or implicitly associated with a wrapper. If other objects are associated with the name of the wrapper that you dropped, you must manually drop those objects.
5. Issue the SQL statements in the **db2look** output file to reconfigure access to Teradata. For example, if the name of the output file is teradata.sql , issue the following command:

   ```
   db2 -tvf teradata.sql
   ```
6. Bind any application packages that were invalidated when you dropped the wrapper.

Procedure:

To access Teradata data sources after you migrate to DB2 Information Integrator from DataJoiner:

1. Check the settings of the environment variables for Teradata data sources:
   a. Locate the db2dj.ini file in the sqllib/cfg subdirectory.
   b. Set the environment variables in the db2dj.ini file as necessary. These variables are:

      COPERR
      The directory in which the errmsg.txt file resides. For example:

      ```
      COPERR=coperr_directory
      ```

      COPLIB
      The directory in which the Teradata shared library file, libcliv2.so, resides. For example:

      ```
      COPLIB=coplib_directory
      ```

      The libcliv2.so file and the errmsg.txt are usually in the same directory. The default directory is /usr/lib on AIX operating systems.
2. If the djxlink script or the djxlinkTeradata script did not run when you installed DB2 Information Integrator, run the djxlink script or the djxlinkTeradata script.

3. If you changed any environment variable in the db2dj.ini file in step 1 or if you ran the djxlink script or the djxlinkTeradata script in step 2, recycle the DB2 instance:
   a. Issue the **db2stop** command.
   b. If you ran the djxlink script or the djxlinkTeradata script, issue the **db2iupdt** command.
   c. Issue the **db2start** command.

   By recycling the DB2 instance, you ensure that the environment variable changes are properly set on the federated server.
4. Use the Teradata BTEQ query tool to test your connection to the Teradata server.
5. Verify that the migration was successful:
   a. Connect to the database.
   b. Issue SELECT, INSERT, UPDATE, and DELETE statements on the nicknames to verify that you can view and modify the data.

   You can also open a pass-through session to the remote database, and issue a SELECT statement on a remote table.

Related tasks:

- "Adding Teradata data sources to a federated server" in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Related reference:

- "db2iupdt - Update Instances Command" in the *Command Reference*
- "DataJoiner terms and their DB2 Information Integrator equivalents" on page 69
- "Migration reference information for Teradata data sources" on page 83

Accessing Documentum data sources after migrating to DB2 Information Integrator

After you migrate to DB2 Information Integrator, you need to verify environment settings and make the Documentum library available to the federated server before you can access your Documentum data sources.

Procedure:

To access Documentum data sources after you migrate to DB2 Information Integrator:

1. Verify that the DOCUMENTUM or DMCL_CONFIG environment variable settings in the `db2dj.ini` file are the same as the settings were before migration.
2. The following examples show the settings for these variables on federated servers that use UNIX operating systems:

   ```
   DOCUMENTUM=<path>
   ```

   ```
   DMCL_CONFIG=<path>/dmcl.ini
   ```

   `<path>` is the fully qualified directory that contains the `dmcl.ini` file that you want to use.

   The default path to the location of the Documentum `dmcl.ini` file is `/pkgs/documentum`. If both lines are included, the DMCL_CONFIG setting is used (on Windows operating systems, a back slash is used instead of the forward slash to define the location of the `dmcl.ini` file).

On AIX and Solaris Operating Environment, the `db2dj.ini` file is located in `$HOME/sqllib/cfg`. On Windows, the `db2dj.ini` file is in `x:\sqllib \cfg`.

3. Make the Documentum client library available to the federated server by copying the client library to the appropriate directory. It is also possible to create a symbolic link for the client library to the appropriate directory.

Related tasks:

- "Adding Documentum data sources to a federated server" in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Related reference:

- "Documentum wrapper library files" in the *IBM DB2 Information Integrator Data Source Configuration Guide*

Chapter 6. Reference information for migrating from DataJoiner Version 2.1.1

This chapter contains reference information that you can use after you migrate from DataJoiner Version 2.1.1 to DB2 Information Integrator, and includes the following topics:

- “General migration reference information”
- “Migration reference information for DB2 family data sources” on page 71
- “Migration reference information for Oracle data sources” on page 76
- “Migration reference information for Informix data sources” on page 78
- “Migration reference information for Sybase data sources” on page 79
- “Migration reference information for Microsoft SQL Server data sources” on page 80
- “Changes to default data type mappings for Microsoft SQL Server data sources” on page 81
- “Migration reference information for ODBC data sources” on page 82
- “Changes to default data type mappings for ODBC data sources” on page 82
- “Migration reference information for Teradata data sources” on page 83

General migration reference information

This section provides information about conceptual and terminology differences between the DataJoiner and DB2 Information Integrator products, and includes the following topics:

- “DataJoiner terms and their DB2 Information Integrator equivalents”
- “DataJoiner catalog conversion” on page 70

DataJoiner terms and their DB2 Information Integrator equivalents

The following table lists common DataJoiner terms and their DB2 Information Integrator equivalents.

Table 2. DataJoiner terms and their DB2 Information Integrator equivalents

DataJoiner term	DB2 Information Integrator term
Data Access Module (DAM)	Wrapper
DataJoiner catalog	Global catalog (sometimes referred to as the federated database system catalog)
DataJoiner database	Federated database
DataJoiner server	Federated server
Protocol	Wrapper
Remote table	Data source object (includes tables, views, synonyms, and other nonrelational objects)
Server mapping	Server definition

Related concepts:

- “The federated database” in the *Federated Systems Guide*
- “Wrappers and wrapper modules” in the *Federated Systems Guide*
- “Server definitions and server options” in the *Federated Systems Guide*
- “The federated server” in the *Federated Systems Guide*
- “The federated database system catalog” in the *Federated Systems Guide*

DataJoiner catalog conversion

In the DataJoiner environment, the tables in the DataJoiner catalog contain information about the data source objects.

In DB2 Information Integrator, the tables in the federated database system catalog contain information about the data source objects. The DB2 federated database system catalog is referred to as the *global catalog*.

The following table lists all data source objects in DataJoiner and DB2 Information Integrator, and the location of the corresponding table in the DataJoiner and global catalogs.

Table 3. Data source objects and catalog information tables in DataJoiner and DB2 Information Integrator

Data source object	DataJoiner catalog table	Global catalog table
Column	syscat.columns	syscat.columns
Column option	Not applicable	syscat.coloptions
Function mappings[1]	syscat.server_functions	syscat.funcmappings
Indexes on nicknames	syscat.indexes	syscat.indexes
Nickname	syscat.tables	syscat.tables
Nickname option	Not applicable	syscat.taboptions
Pass-through privileges	syscat.passthru_auth	syscat.passthruauth
Packages	syscat.packages	syscat.packages
Package dependency	syscat.packagedep	syscat.packagedep
Reverse type mappings	syscat.revtypemappings	syscat.revtypemappings
Server	syscat.servers	syscat.servers
Server option	syscat.server_options	syscat.serveroptions
Stored procedure nickname	syscat.procedures	Not applicable
Type mappings	syscat.server_datatypes	syscat.typemappings
User mappings	syscat.remoteusers	syscat.useroptions
Wrapper	Not applicable	syscat.wrappers

1

Not all information that is stored in a single table in the DataJoiner catalog is placed in the corresponding single table in the global catalog. In the global catalog, data source object information can be stored in multiple tables that relate to the same data source object. For example, all server information is stored in the syscat.servers table in the DataJoiner catalog. In the global catalog, all server information is stored the syscat.servers and syscat.serveroptions tables.

[1] In DB2 Information Integrator, default function mappings for a given source type are stored in the wrapper library. In DataJoiner, default function mappings for a given source type are stored in the syscat.server_functions catalog table.

The maximum length of the data source object identifiers also differs between DataJoiner and DB2 Information Integrator. The following table lists all data source object identifiers and their maximum lengths in DataJoiner and DB2 Information Integrator.

Table 4. Data source object identifier lengths in DataJoiner and DB2 Information Integrator

Data source object identifier	DataJoiner length	DB2 Information Integrator length
Column name	18	30
Function mapping name	18	128
Index name	8	18
Nickname	18	128
Package name	8	8
Server name	18	128
Schema name	8	128
Type mapping name	18	18
Wrapper name	Not applicable	128

Related concepts:

- "Wrappers and wrapper modules" in the *Federated Systems Guide*
- "Server definitions and server options" in the *Federated Systems Guide*
- "User mappings" in the *Federated Systems Guide*
- "Nicknames and data source objects" in the *Federated Systems Guide*
- "Nickname column options" in the *Federated Systems Guide*
- "Data type mappings" in the *Federated Systems Guide*
- "Function mappings" in the *Federated Systems Guide*
- "Index specifications" in the *Federated Systems Guide*
- "The federated database system catalog" in the *Federated Systems Guide*

Migration reference information for DB2 family data sources

This topic describes the differences between DataJoiner protocols and the corresponding DB2 Information Integrator wrappers for all DB2 family data sources that are supported in DB2 Information Integrator.

DB2 Information Integrator supports the following DB2 family data sources:

- DB2 for Linux, UNIX, and Windows
- DB2 for z/OS and OS/390
- DB2 for iSeries (AS/400)
- DB2 for VM and VSE

In DataJoiner, the following three protocols support all DB2 family data sources:

- drda
- drdaIP
- db2ra

In DB2 Information Integrator, the three DataJoiner protocols correspond to the single DRDA wrapper. The DRDA wrapper is used for all DB2 family data sources.

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Changes to default data type mappings for DB2 for z/OS and OS/390 data sources" on page 72
- "Changes to default data type mappings for DB2 for iSeries (AS/400) data sources" on page 73
- "Changes to default data type mappings for DB2 for Linux, UNIX, and Windows data sources" on page 74
- "Changes to default data type mappings for DB2 for VM and VSE data sources" on page 75

Changes to the default data type mappings for DB2 family data sources

This section provides specific information about the differences between DataJoiner and DB2 Information Integrator for the default data type mappings for each DB2 family data source.

Changes to default data type mappings for DB2 for z/OS and OS/390 data sources

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for DB2 for z/OS and OS/390 data sources. The data mappings that remain unchanged between the two products are not shown.

Table 5. DB2 for z/OS and OS/390 forward default data type mapping changes

DB2 data type	DataJoiner data type	Federated data type
BLOB	-	BLOB
CHAR	LONG VARCHAR	-
CHAR	-	VARCHAR
CLOB	-	CLOB
DBCLOB	-	DBCLOB
LONGVAR	LONG VARCHAR	CLOB
LONGVARG	LONG VARGRAPHIC	DBCLOB
VARCHAR	LONG VARCHAR	-
VARCHAR	-	VARCHAR
VARGRAPHIC	LONG VARGRAPH	-
VARGRAPHIC	-	VARGRAPHIC

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for DB2 for z/OS and OS/390 data sources. The data mappings that remain unchanged between the two products are not shown.

Table 6. DB2 for z/OS and OS/390 reverse default data type mapping changes

Federated data type	DB2 data type used in DataJoiner	DB2 data type used in federated systems
BLOB	LONGVAR	BLOB
CLOB	LONGVAR	CLOB
DBCLOB	-	DBCLOB
LONG VARCHAR	LONG VAR	-
LONG VARGRAPHIC	LONG VARG	-

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Migration reference information for DB2 family data sources" on page 71

Changes to default data type mappings for DB2 for iSeries (AS/400) data sources

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for DB2 for iSeries (AS/400) data sources. The data mappings that remain unchanged between the two products are not shown.

Table 7. DB2 for iSeries (AS/400) forward default data type mapping changes

DB2 data type	DataJoiner data type	Federated data type
BLOB	-	BLOB
CHAR	LONG VARCHAR	-
CHAR	-	VARCHAR
CLOB	-	CLOB
DBCLOB	-	DBCLOB
LONGVARCH	-	CLOB
LONGVARG	LONG VARGRAPHIC	DBCLOB
VARCHAR	LONG VARCHAR	-
VARCHAR	-	VARCHAR
VARGRAPHIC	LONG VARG	-
VARGRAPHIC	-	VARGRAPHIC

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for DB2 for iSeries (AS/400) data sources. The data mappings that remain unchanged between the two products are not shown.

Table 8. DB2 for iSeries (AS/400) reverse default data type mapping changes

Federated data type	DB2 data type used in DataJoiner	DB2 data type used in federated systems
BLOB	BLOB	BLOB
CLOB	CLOB	CLOB

Table 8. DB2 for iSeries (AS/400) reverse default data type mapping changes (continued)

Federated data type	DB2 data type used in DataJoiner	DB2 data type used in federated systems
DBCLOB	DBCLOB	DBCLOB
LONG VARCHAR	LONG VARCH	CLOB
LONG VARGRAPHIC	LONG VARG	DBCLOB

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Migration reference information for DB2 family data sources" on page 71

Changes to default data type mappings for DB2 for Linux, UNIX, and Windows data sources

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for DB2 for Linux, UNIX, and Windows data sources. The data mappings that remain unchanged between the two products are not shown.

Table 9. DB2 for UNIX and Windows forward default data type mapping changes

DB2 data type	DataJoiner data type	Federated data type
BLOB	-	BLOB
CHAR	LONG VARCHAR	-
CHAR	VARCHAR	-
CHAR	-	CHAR
CLOB	-	CLOB
DBCLOB	-	DBCLOB
LONG VARCHAR	LONG VARCHAR	CLOB
LONG VARGRAPHIC	LONG VARGRAPHIC	DBCLOB
VARCHAR	LONG VARCHAR	-
VARCHAR	-	VARCHAR
VARGRAPHIC	LONG VARGRAPHIC	-
VARGRAPHIC	-	VARGRAPHIC

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for DB2 for UNIX and Windows data sources. The data mappings that remain unchanged between the two products are not shown.

Table 10. DB2 for UNIX and Windows reverse default data type mapping changes

Federated data type	DB2 data type used in DataJoiner	DB2 data type used in Federated systems
BLOB	BLOB	BLOB
CLOB	CLOB	CLOB
DBCLOB	DBCLOB	DBCLOB

Table 10. DB2 for UNIX and Windows reverse default data type mapping changes (continued)

Federated data type	DB2 data type used in DataJoiner	DB2 data type used in Federated systems
LONG VARCHAR	LONG VAR	CLOB
LONG VARGRAPHIC	LONG VARGRAPHIC	DBCLOB

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Migration reference information for DB2 family data sources" on page 71

Changes to default data type mappings for DB2 for VM and VSE data sources

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for DB2 for VM and VSE data sources. The data mappings that remain unchanged between the two products are not shown.

Table 11. DB2 for VM and VSE forward default data type mappings (not all columns are shown)

DB2 data type	DataJoiner data type	Federated data type
BLOB	-	BLOB
CLOB	-	CLOB
DBCLOB	-	DBCLOB
CHAR	VARCHAR	-
CHAR	LONG VARCHAR	-
VARCHAR	LONG VARCHAR	-
VARCHAR	-	VARCHAR
LONGVARCHAR	LONG VARCHAR	CLOB
VARGRAPH	LONG VARGRAPHIC	-
LONGVARG	LONG VARGRAPHIC	DBCLOB
VARGRAPHIC	-	VARGRAPHIC

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for DB2 for VM and VSE data sources. The data mappings that remain unchanged between the two products are not shown.

Table 12. DB2 for VM and VSE reverse default data type mappings (not all columns are shown)

Federated data type	DB2 data type used in DataJoiner	DB2 data type used in Federated systems
BLOB	-	BLOB
CLOB	-	CLOB
DBCLOB	-	DBCLOB

Table 12. DB2 for VM and VSE reverse default data type mappings (not all columns are shown) (continued)

Federated data type	DB2 data type used in DataJoiner	DB2 data type used in Federated systems
LONG VARCHAR	LONG VARCHAR	-
VARGRAPHIC	LONG VARG	-
VARGRAPHIC	-	-

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Migration reference information for DB2 family data sources" on page 71

Migration reference information for Oracle data sources

This topic describes the differences between DataJoiner protocols and the corresponding DB2 Information Integrator wrappers for Oracle data sources.

The following table lists the DataJoiner protocols for accessing Oracle data sources and the corresponding DB2 Information Integrator wrapper names.

Table 13. Oracle data source protocol names and their corresponding wrapper names

DataJoiner protocol name	DB2 Information Integrator wrapper name
sqlnet	SQLNET
net8	NET8

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Changes to default data type mappings for Oracle SQLNET" on page 76
- "Changes to default data type mappings for Oracle NET8" on page 77

Changes to the default data type mappings for Oracle data sources

This section provides specific information about the differences between DataJoiner and DB2 Information Integrator for the default data type mappings for Oracle data sources.

Changes to default data type mappings for Oracle SQLNET

Important: The information in this topic does not apply to DB2 Information Integrator 8.2 and later. The SQLNET wrapper is not supported for accessing Oracle data sources in DB2 Information Integrator Version 8.2.

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for the Oracle SQLNET wrapper. The data mappings that remain unchanged between the two products are not shown.

Table 14. Oracle SQLNET forward default data type mapping changes

Oracle data type	DataJoiner data type	Federated data type
FLOAT	DOUBLE	REAL

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for the Oracle SQLNET wrapper. The data mappings that remain unchanged between the two products are not shown.

Table 15. Oracle SQLNET reverse default data type mapping changes

Federated data type	Oracle data type used in DataJoiner	Oracle data type used in federated systems
LONG VARCHAR	LONG	-
LONG VARCHAR	LONG RAW	-
REAL	-	FLOAT (length: 63)
VARCHAR*	LONG RAW	RAW
VARCHAR*	LONG RAW	-
VARCHAR	LONG	VARCHAR2

* The upper and lower lengths are different for these two data type mappings. The lengths are described in another topic. See the Related reference links, below.

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Migration reference information for Oracle data sources" on page 76

Changes to default data type mappings for Oracle NET8

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for the Oracle NET8 wrapper. The data mappings that remain unchanged between the two products are not shown.

Table 16. Oracle NET8 forward default data type mapping changes

Oracle data type	DataJoiner data type	Federated data type
BLOB	-	BLOB
CLOB	-	CLOB
LONG	CLOB	CLOB
LONG RAW	BLOB	BLOB
NCHAR(1-127)*	-	GRAPHIC
NCHAR(128-2000)**	-	VARGRAPHIC
NCLOB **	-	DBCLOB
NVARCHAR2**	-	VARGRAPHIC
TIMESTAMP	-	TIMESTAMP

* Requires Oracle client and server version 9 or later

** Requires Oracle client and server version 9 or later and local UNICODE codepage

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for the Oracle NET8 wrapper. The data mappings that remain unchanged between the two products are not shown.

Table 17. Oracle NET8 reverse default data type mapping changes

Federated data type	Oracle data type used in DataJoiner	Oracle data type used in federated systems
BLOB	LONG RAW	BLOB
CLOB	LONG	CLOB
DBCLOB**	-	NCLOB
FLOAT	-	FLOAT
GRAPHIC**	-	NCHAR
LONG VARCHAR	LONG	-
LONG VARCHAR	LONG RAW	-
REAL	-	FLOAT (length: 63)
VARCHAR*	LONG RAW	RAW
VARCHAR*	LONG RAW	-
VARCHAR	LONG	VARCHAR2
VARGRAPHIC**	-	NVARCHAR2
* The upper and lower lengths are different for these two data type mappings. The lengths are described in another topic. See the Related reference links, below.		
** Requires Oracle client and server version 9 or later and local UNICODE codepage		

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Migration reference information for Oracle data sources" on page 76

Migration reference information for Informix data sources

This topic describes the differences between DataJoiner protocols and the corresponding DB2 Information Integrator wrapper for Informix data sources.

The following table lists the Informix data sources that are supported in DB2 Information Integrator, the DataJoiner protocol names for each data source, and the corresponding DB2 Information Integrator wrapper name.

Table 18. Informix data source protocol names and their corresponding wrapper names

Data source	DataJoiner protocol name	DB2 Information Integrator wrapper name
Informix Version 7	informix7	INFORMIX
Informix Version 9	informix9	INFORMIX

The rest of this topic provides specific information about the differences in data type mappings between DataJoiner and DB2 Information Integrator.

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for all Informix data sources. The data mappings that remain unchanged between the two products are not shown.

Table 19. Informix forward default data type mapping changes

Informix data type	DataJoiner data type	Federated data type
CHAR	LONG VARCHAR	VARCHAR
NCHAR	LONG VARCHAR	VARCHAR
SMALLFLOAT	DOUBLE	REAL

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for all Informix data sources. The data mappings that remain unchanged between the two products are not shown.

Table 20. Informix reverse default data type mapping changes

Federated data type	Informix data type used in DataJoiner	Informix data type used in federated systems
VARCHAR	TEXT[1]	TEXT[2]
VARCHAR	-	LVARCHAR[3]
VARCHAR	-	TEXT[4]

Notes:

1. This type mapping has a length that ranges from 255 to 4000.
2. This type mapping is only valid with Informix Version 7, and has a length that ranges from 255 to 32672.
3. This type mapping is only valid with Informix Versions 8 and 9, and has a length that ranges from 255 to 2048.
4. This type mapping is only valid with Informix Versions 8 and 9, and has a length that ranges from 2049 to 36672.

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*

Migration reference information for Sybase data sources

The name of the Sybase data source protocol that is supported in DataJoiner remains unchanged in DB2 Information Integrator. The `ctlib` protocol becomes the CTLIB wrapper. The `dblib` protocol is not supported in DB2 Information Integrator Version 8.2 and later.

The rest of this topic provides specific information about the differences in data type mappings between DataJoiner and DB2 Information Integrator.

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for Sybase data sources. The data mappings that remain unchanged between the two products are not shown.

Table 21. Sybase forward default data type mapping changes

Sybase data type	DataJoiner data type	Federated data type
BINARY	LONG VARCHAR	VARCHAR
CHAR	LONG VARCHAR	VARCHAR
NCHAR	LONG VARCHAR	VARCHAR
NVARCHAR	LONG VARCHAR	VARCHAR
REAL	DOUBLE	REAL
VARBINARY	LONG VARCHAR	VARCHAR
VARCHAR	LONG VARCHAR	VARCHAR

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for Sybase data sources. The data mappings that remain unchanged between the two products are not shown.

Table 22. Sybase reverse default data type mapping changes

Federated data type	Sybase data type used in DataJoiner	Sybase data type used in federated systems
VARCHAR	CHAR	VARCHAR
VARCHAR	BINARY	VARBINARY
VARCHAR	BINARY	IMAGE
LONG VARCHAR	TEXT	-
LONG VARCHAR	IMAGE	-

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*

Migration reference information for Microsoft SQL Server data sources

This topic describes the differences between DataJoiner protocols and the corresponding DB2 Information Integrator wrappers for Microsoft SQL Server data sources.

The following table lists the DataJoiner protocols for accessing Microsoft SQL Server data sources and the corresponding DB2 Information Integrator wrapper names.

Table 23. Microsoft SQL Server data source protocol names and their corresponding wrapper names

Data source	DataJoiner protocol name	Default DB2 Information Integrator wrapper name
Microsoft SQL Server on UNIX	• For AIX: mssqlodbc and djxsqlnk • For the Solaris Operating Environment: mssqlodbc or djxsqlnk	MSSQLODBC3
Microsoft SQL Server on Windows	djxmssql	DJXMSSQL3

Related reference:

- “Default forward data type mappings” in the *Federated Systems Guide*
- “Default reverse data type mappings” in the *Federated Systems Guide*
- “Changes to default data type mappings for Microsoft SQL Server data sources” on page 81
- “Migration reference information for ODBC data sources” on page 82

Changes to default data type mappings for Microsoft SQL Server data sources

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for all Microsoft SQL Server data sources. In data types where the same remote data type maps to different DB2 data types, the mapping used depends on the length.

The data mappings that remain unchanged between the two products are not shown.

Table 24. Microsoft SQL Server forward default data type mapping changes

Microsoft SQL Server data type	DataJoiner data type	Federated data type
binary	LONG VARCHAR	VARCHAR
char	LONG VARCHAR	VARCHAR
float	DOUBLE	REAL
floatn	DOUBLE	REAL
nchar	CHARACTER	CHARACTER
nchar	CHARACTER	VARCHAR
nchar	VARCHAR	VARCHAR
nchar	LONG VARCHAR	-
nvarchar	VARCHAR	VARCHAR
nvarchar	LONG VARCHAR	-
real	DOUBLE	REAL
varbinary	LONG VARCHAR	VARCHAR
varchar	LONG VARCHAR	VARCHAR

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for all Microsoft SQL Server data sources. The data mappings that remain unchanged between the two products are not shown.

Table 25. Microsoft SQL Server reverse default data type mapping changes

Federated data type	Microsoft SQL server data type used in DataJoiner	Microsoft SQL server data type used in federated systems
LONG VARCHAR bit data: N	text	-
VARCHAR bit data: N	char	varchar
VARCHAR bit data: N	text	varchar

Table 25. Microsoft SQL Server reverse default data type mapping changes (continued)

Federated data type	Microsoft SQL server data type used in DataJoiner	Microsoft SQL server data type used in federated systems
VARCHAR bit data: N	-	varchar
VARCHAR bit data: N	-	text
VARCHAR bit data: Y	image	varbinary
VARCHAR bit data: Y	-	varbinary
VARCHAR bit data: Y	-	image

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Migration reference information for Microsoft SQL Server data sources" on page 80

Migration reference information for ODBC data sources

This topic describes the differences between DataJoiner protocol and the corresponding DB2 Information Integrator wrapper for ODBC data sources.

Any data source that can use ODBC driver 3.0 or later is a potential ODBC data source. For example, Microsoft SQL Server can use the ODBC driver.

The DataJoiner protocol Generic supports ODBC data sources. In DB2 Information Integrator, the Generic protocol becomes the ODBC wrapper.

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*
- "Changes to default data type mappings for Microsoft SQL Server data sources" on page 81
- "Changes to default data type mappings for ODBC data sources" on page 82

Changes to default data type mappings for ODBC data sources

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for ODBC data sources. The data mappings that remain unchanged between the two products are not shown.

Table 26. ODBC forward default data type mapping changes

ODBC data type	DataJoiner data type	Federated data type
SQL_BIGINT	DECIMAL	BIGINT
SQL_CHAR	LONG VARCHAR	VARCHAR
SQL_CHAR	LONG VARCHAR	-
SQL_BINARY	LONG VARCHAR	VARCHAR FOR BIT DATA
SQL_BINARY	LONG VARCHAR	-
SQL_VARCHAR	LONG VARCHAR	VARCHAR

Table 26. ODBC forward default data type mapping changes (continued)

ODBC data type	DataJoiner data type	Federated data type
SQL_VARCHAR	LONG VARCHAR	-
SQL_VARBINARY	LONG VARCHAR	VARCHAR FOR BIT DATA
SQL_VARBINARY	LONG VARCHAR	-
SQL_WCHAR	-	CHAR
SQL_WCHAR	-	VARCHAR
SQL_WVARCHAR	-	VARCHAR
SQL_WLONGVARCHAR	-	CLOB

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for ODBC data sources. The data mappings that remain unchanged between the two products are not shown.

Table 27. ODBC reverse default data type mapping changes

Federated data type	ODBC data type used in DataJoiner	ODBC data type used in Federated systems
BLOB	-	SQL_LONGVARBINARY
BIGINT	-	SQL_BIGINT
CHAR	-	SQL_CHAR
CHAR FOR BIT DATA	-	SQL_BINARY
CLOB	-	SQL_LONGVARCHAR
DATE	-	SQL_TYPE_DATE
DECIMAL	-	SQL_DECIMAL
DOUBLE	-	SQL_DOUBLE
FLOAT	-	SQL_FLOAT
INTEGER	-	SQL_INTEGER
NUMERIC	-	SQL_DECIMAL
REAL	-	SQL_REAL
SMALLINT	-	SQL_SMALLINT
TIME	-	SQL_TYPE_TIME
TIMESTAMP	-	SQL_TYPE_TIMESTAMP
VARCHAR	-	SQL_VARCHAR
VARCHAR FOR BIT DATA	-	SQL_VARBINARY

Related reference:

- “Default forward data type mappings” in the *Federated Systems Guide*
- “Default reverse data type mappings” in the *Federated Systems Guide*
- “Migration reference information for ODBC data sources” on page 82

Migration reference information for Teradata data sources

This topic describes the differences between DataJoiner protocol and the corresponding DB2 Information Integrator wrapper for Teradata data sources.

The DataJoiner protocol teradata becomes the corresponding DB2 Information Integrator wrapper teradata.

The rest of this topic provides specific information about the differences in data type mappings between DataJoiner and DB2 Information Integrator.

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default forward data type mappings for the Teradata data source. The data mappings that remain unchanged between the two products are not shown. Detailed information for each data type mapping (for example, lengths) is described in another topic. See the Related reference links.

Table 28. Teradata forward default data type mapping changes

Teradata data type	DataJoiner data type	Federated data type
BYTE	VARCHAR / LONG VARCHAR FOR BIT DATA	VARCHAR FOR BIT DATA / BLOB
CHAR	VARCHAR / LONG VARCHAR	VARCHAR / CLOB
VARBYTE	VARCHAR FOR BIT DATA / LONG VARCHAR FOR BIT DATA	VARCHAR FOR BIT DATA / BLOB
VARCHAR	VARCHAR / LONG VARCHAR	VARCHAR / CLOB
VARGRAPHIC	VARGRAPHIC / LONG VARGRAPHIC	VARGRAPHIC / DBCLOB

The following table lists the differences between DataJoiner and DB2 Information Integrator in the default reverse data type mappings for the Teradata data source. The data mappings that remain unchanged between the two products are not shown. Detailed information for each data type mapping (for example, lengths) is described in another topic. See the Related reference links.

Table 29. Teradata reverse default data type mapping changes

Federated type name	Teradata data type used in DataJoiner	Teradata data type used in Federated systems
BLOB	-	VARCHAR
CLOB	-	VARBYTE
DBCLOB	-	VARGRAPHIC
LONG VARCHAR	VARCHAR	-
LONG VARCHAR FOR BIT DATA	VARBYTE	-
LONG VARGRAPHIC	VARGRAPHIC	-

Related reference:

- “Default forward data type mappings” in the *Federated Systems Guide*
- “Default reverse data type mappings” in the *Federated Systems Guide*

Chapter 7. Reference information for migrating from DB2 Version 7

This chapter contains reference information that you can use after you migrate from DB2 Version 7 to DB2 Information Integrator.

DB2 Version 7 migration reference information

This topic provides information about wrapper library name changes, related libraries, and data type mapping differences when migrating from DB2 Version 7 to DB2 Information Integrator.

Wrapper library name changes

The wrapper library names in DB2 Information Integrator are different than the wrapper library names in DB2 Version 7. The following table lists the DB2 Information Integrator wrapper library names.

Table 30. DB2 Information Integrator wrapper library names and related libraries

Wrapper	Windows wrapper library names (related libraries)	AIX wrapper library names (related libraries)
NET8	db2net8.dll (db2net8F.dll, db2net8U.dll)	libdb2net8.a (libdb2net8F.a, libdb2net8U.a)
CTLIB	db2ctlib.dll	libdb2ctlib.a
MSSQLODBC3	db2mssql3.dll (db2mssql3F.dll, db2mssql3U.dll)	libdb2mssql3.a (libdb2mssql3F.a, libdb2mssql3U.a)
DRDA	db2drda.dll (db2drdaF.dll, db2drdaU.dll)	libdb2drda.a (libdb2drdaF.a, libdb2drdaU.a)
INFORMIX	db2informix.dll (db2informixF.dll, db2informixU.dll)	libdb2informix.a (libdb2informixF.a, libdb2informixU.a)
OLEDB	db2oledb.dll	

Data type mapping between DB2 Version 7 and DB2 Information Integrator

There are a few differences between the default data type mapping in DB2 Version 7 and DB2 Information Integrator Version 8. For a complete set of default forward and reverse data type mappings, see the topics ″Default forward data type mappings″ and ″Default reverse data type mappings″.

The default forward data type mapping changes are listed in the following table:

Table 31. Default forward data type mapping changes

Data source	Data type	DB2 Version 7 data type	DB2 Information Integrator data type
DB2 for z/OS and OS/390	CHAR(255) for bit data	VARCHAR	
DB2 for iSeries	GRAPHIC(127)	VARGRAPHIC	GRAPHIC
Informix	interval	DEC(19,5)	VARCHAR(25)
	boolean	SMALLINT	CHAR(1)

The default reverse data type mapping changes are listed in the following table:

Table 32. Default reverse data type mapping changes

Data source	Federated data type	Data source data type in DB2 Version 7	Data source data type in DB2 Information Integrator
Sybase	GRAPHIC	nchar	Not supported
	VARGRAPHIC	nvarchar	Not supported

Related reference:

- "Default forward data type mappings" in the *Federated Systems Guide*
- "Default reverse data type mappings" in the *Federated Systems Guide*

DB2 Information Integrator documentation

This topic provides information about the documentation that is available for DB2 Information Integrator. The tables in this topic provide the official document title, form number, and location of each PDF book. To order a printed book, you must know either the official book title or the document form number. Titles, file names, and the locations of the DB2 Information Integrator release notes and installation requirements are also provided in this topic.

This topic contains the following sections:

- Accessing DB2 Information Integrator documentation
- Documentation for replication function on z/OS
- Documentation for event publishing function for DB2 Universal Database on z/OS
- Documentation for event publishing function for IMS and VSAM on z/OS
- Documentation for event publishing and replication function on Linux, UNIX, and Windows
- Documentation for federated function on z/OS
- Documentation for federated function on Linux, UNIX, and Windows
- Documentation for enterprise search on Linux, UNIX, and Windows
- Release notes and installation requirements

Accessing DB2 Information Integrator documentation

All DB2 Information Integrator books and release notes are available in PDF files from the DB2 Information Integrator Support Web site at www.ibm.com/software/data/integration/db2ii/support.html.

To access the latest DB2 Information Integrator product documentation, from the DB2 Information Integrator Support Web site, click on the Product Information link, as shown in Figure 1 on page 88.

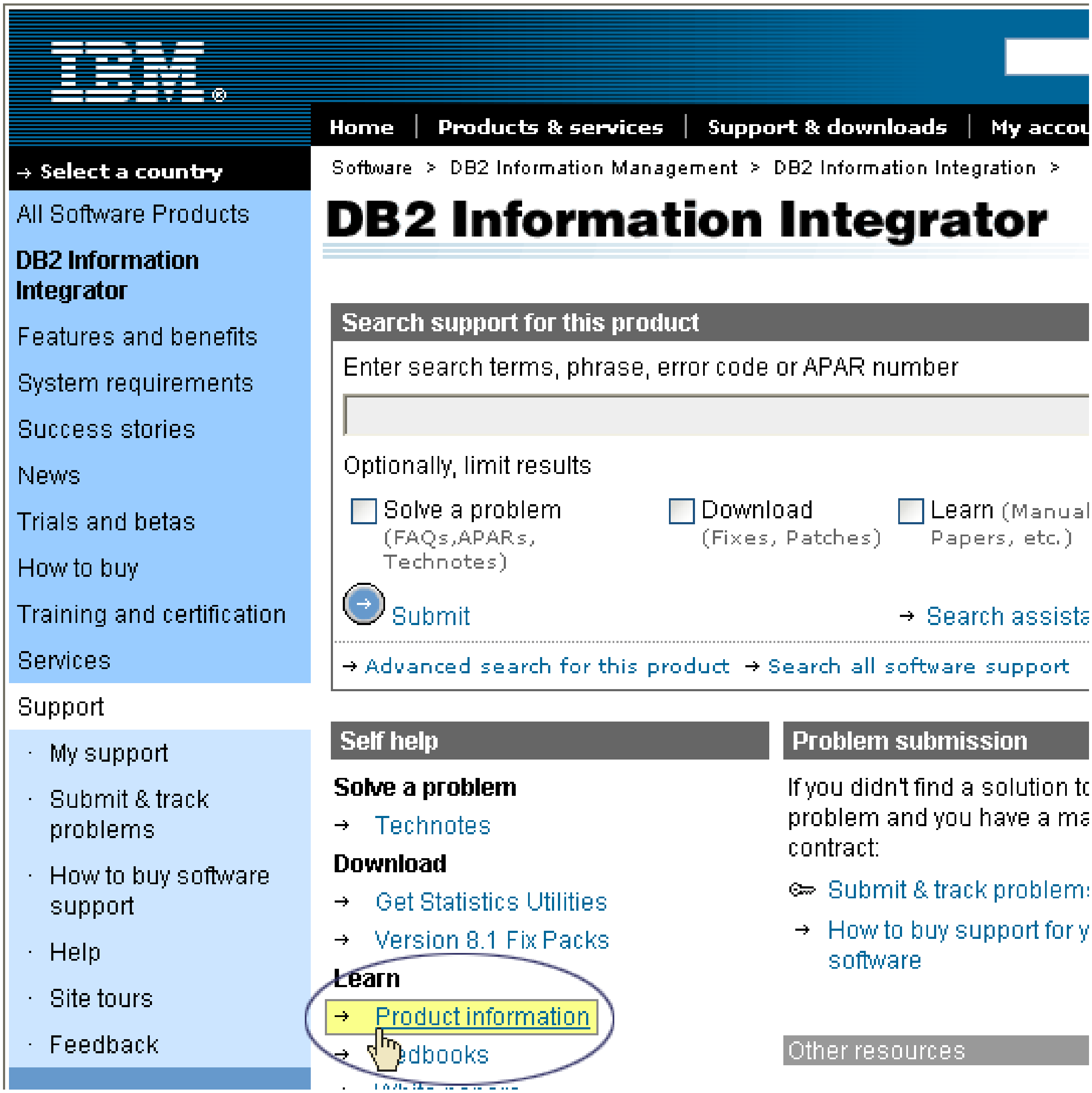
IBM
Home
Products & services
Support & downloads
My accou
→ Select a country
Software > DB2 Information Management > DB2 Information Integration >
DB2 Information Integrator
All Software Products
DB2 Information Integrator
Features and benefits
System requirements
Success stories
News
Trials and betas
How to buy
Training and certification
Services
Support
· My support
· Submit & track problems
· How to buy software support
· Help
· Site tours
· Feedback
Search support for this product
Enter search terms, phrase, error code or APAR number
Optionally, limit results
□ Solve a problem (FAQs,APARs, Technotes)
□ Download (Fixes, Patches)
□ Learn (Manual Papers, etc.)
Submit
→ Search assista
→ Advanced search for this product
→ Search all software support
Self help
Solve a problem
→ Technotes
Download
→ Get Statistics Utilities
→ Version 8.1 Fix Packs
Learn
→ Product information
→ Redbooks
Problem submission
If you didn't find a solution to problem and you have a ma contract:
Submit & track problems
→ How to buy support for y software
Other resources

Figure 1. Accessing the Product Information link from DB2 Information Integrator Support Web site

You can access the latest DB2 Information Integrator documentation, in all supported languages, from the Product Information link:

- DB2 Information Integrator product documentation in PDF files
- Fix pack product documentation, including release notes
- Instructions for downloading and installing the DB2 Information Center for Linux, UNIX, and Windows
- Links to the DB2 Information Center online

Scroll though the list to find the product documentation for the version of DB2 Information Integrator that you are using.

The DB2 Information Integrator Support Web site also provides support documentation, IBM Redbooks, white papers, product downloads, links to user groups, and news about DB2 Information Integrator.

You can also view and print the DB2 Information Integrator PDF books from the *DB2 PDF Documentation* CD.

To view or print the PDF documentation:

1. From the root directory of the *DB2 PDF Documentation* CD, open the index.htm file.
2. Click the language that you want to use.
3. Click the link for the document that you want to view.

Documentation about replication function on z/OS

Table 33. DB2 Information Integrator documentation about replication function on z/OS

Name	Form number	Location
ASNCLP Program Reference for Replication and Event Publishing	N/A	DB2 Information Integrator Support Web site
Introduction to Replication and Event Publishing	GC18-7567	DB2 Information Integrator Support Web site
Migrating to SQL Replication	N/A	DB2 Information Integrator Support Web site
Replication and Event Publishing Guide and Reference	SC18-7568	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Replication Installation and Customization Guide for z/OS	SC18-9127	DB2 Information Integrator Support Web site
SQL Replication Guide and Reference	SC27-1121	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Tuning for Replication and Event Publishing Performance	N/A	DB2 Information Integrator Support Web site
Tuning for SQL Replication Performance	N/A	DB2 Information Integrator Support Web site
Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS	N/A	• In the DB2 Information Center, **Product Overviews** > **Information Integration** > **DB2 Information Integrator overview** > **Problems, workarounds, and documentation updates** • DB2 Information Integrator Installation launchpad • DB2 Information Integrator Support Web site • The *DB2 Information Integrator* product CD

Documentation about event publishing function for DB2 Universal Database on z/OS

Table 34. DB2 Information Integrator documentation about event publishing function for DB2 Universal Database on z/OS

Name	Form number	Location
ASNCLP Program Reference for Replication and Event Publishing	N/A	DB2 Information Integrator Support Web site
Introduction to Replication and Event Publishing	GC18-7567	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Replication and Event Publishing Guide and Reference	SC18-7568	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Tuning for Replication and Event Publishing Performance	N/A	DB2 Information Integrator Support Web site
Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS	N/A	• In the DB2 Information Center, **Product Overviews** > **Information Integration** > **DB2 Information Integrator overview** > **Problems, workarounds, and documentation updates** • DB2 Information Integrator Installation launchpad • DB2 Information Integrator Support Web site • The *DB2 Information Integrator* product CD

Documentation about event publishing function for IMS and VSAM on z/OS

Table 35. DB2 Information Integrator documentation about event publishing function for IMS and VSAM on z/OS

Name	Form number	Location
Client Guide for Classic Federation and Event Publisher for z/OS	SC18-9160	DB2 Information Integrator Support Web site
Data Mapper Guide for Classic Federation and Event Publisher for z/OS	SC18-9163	DB2 Information Integrator Support Web site
Getting Started with Event Publisher for z/OS	GC18-9186	DB2 Information Integrator Support Web site
Installation Guide for Classic Federation and Event Publisher for z/OS	GC18-9301	DB2 Information Integrator Support Web site
Operations Guide for Event Publisher for z/OS	SC18-9157	DB2 Information Integrator Support Web site

Table 35. DB2 Information Integrator documentation about event publishing function for IMS and VSAM on z/OS (continued)

Name	Form number	Location
Planning Guide for Event Publisher for z/OS	SC18-9158	DB2 Information Integrator Support Web site
Reference for Classic Federation and Event Publisher for z/OS	SC18-9156	DB2 Information Integrator Support Web site
System Messages for Classic Federation and Event Publisher for z/OS	SC18-9162	DB2 Information Integrator Support Web site
Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS	N/A	DB2 Information Integrator Support Web site
Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS	N/A	DB2 Information Integrator Support Web site

Documentation about event publishing and replication function on Linux, UNIX, and Windows

Table 36. DB2 Information Integrator documentation about event publishing and replication function on Linux, UNIX, and Windows

Name	Form number	Location
ASNCLP Program Reference for Replication and Event Publishing	N/A	DB2 Information Integrator Support Web site
Installation Guide for Linux, UNIX, and Windows	GC18-7036	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Introduction to Replication and Event Publishing	GC18-7567	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Migrating to SQL Replication	N/A	DB2 Information Integrator Support Web site
Replication and Event Publishing Guide and Reference	SC18-7568	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
SQL Replication Guide and Reference	SC27-1121	DB2 Information Integrator Support Web site
Tuning for Replication and Event Publishing Performance	N/A	DB2 Information Integrator Support Web site
Tuning for SQL Replication Performance	N/A	DB2 Information Integrator Support Web site

Table 36. DB2 Information Integrator documentation about event publishing and replication function on Linux, UNIX, and Windows (continued)

Name	Form number	Location
Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS	N/A	• In the DB2 Information Center, **Product Overviews** > **Information Integration** > **DB2 Information Integrator overview** > **Problems, workarounds, and documentation updates** • DB2 Information Integrator Installation launchpad • DB2 Information Integrator Support Web site • The *DB2 Information Integrator* product CD

Documentation about federated function on z/OS

Table 37. DB2 Information Integrator documentation about federated function on z/OS

Name	Form number	Location
Client Guide for Classic Federation and Event Publisher for z/OS	SC18-9160	DB2 Information Integrator Support Web site
Data Mapper Guide for Classic Federation and Event Publisher for z/OS	SC18-9163	DB2 Information Integrator Support Web site
Getting Started with Classic Federation for z/OS	GC18-9155	DB2 Information Integrator Support Web site
Installation Guide for Classic Federation and Event Publisher for z/OS	GC18-9301	DB2 Information Integrator Support Web site
Reference for Classic Federation and Event Publisher for z/OS	SC18-9156	DB2 Information Integrator Support Web site
System Messages for Classic Federation and Event Publisher for z/OS	SC18-9162	DB2 Information Integrator Support Web site
Transaction Services Guide for Classic Federation for z/OS	SC18-9161	DB2 Information Integrator Support Web site
Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS	N/A	DB2 Information Integrator Support Web site

Documentation about federated function on Linux, UNIX, and Windows

Table 38. DB2 Information Integrator documentation about federated function on Linux, UNIX, and Windows

Name	Form number	Location
Application Developer's Guide	SC18-7359	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site

Table 38. DB2 Information Integrator documentation about federated function on Linux, UNIX, and Windows (continued)

Name	Form number	Location
C++ API Reference for Developing Wrappers	SC18-9172	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Data Source Configuration Guide	N/A	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Federated Systems Guide	SC18-7364	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Guide to Configuring the Content Connector for VeniceBridge	N/A	DB2 Information Integrator Support Web site
Installation Guide for Linux, UNIX, and Windows	GC18-7036	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Java API Reference for Developing Wrappers	SC18-9173	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Migration Guide	SC18-7360	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Wrapper Developer's Guide	SC18-9174	• *DB2 PDF Documentation* CD • DB2 Information Integrator Support Web site
Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS	N/A	• In the DB2 Information Center, **Product Overviews** > **Information Integration** > **DB2 Information Integrator overview** > **Problems, workarounds, and documentation updates** • DB2 Information Integrator Installation launchpad • DB2 Information Integrator Support Web site • The *DB2 Information Integrator* product CD

Documentation about enterprise search function on Linux, UNIX, and Windows

Table 39. DB2 Information Integrator documentation about enterprise search function on Linux, UNIX, and Windows

Name	Form number	Location
Administering Enterprise Search	SC18-9283	DB2 Information Integrator Support Web site
Installation Guide for Enterprise Search	GC18-9282	DB2 Information Integrator Support Web site
Programming Guide and API Reference for Enterprise Search	SC18-9284	DB2 Information Integrator Support Web site
Release Notes for Enterprise Search	N/A	DB2 Information Integrator Support Web site

Release notes and installation requirements

Release notes provide information that is specific to the release and fix pack level for your product and include the latest corrections to the documentation for each release.

Installation requirements provide information that is specific to the release of your product.

Table 40. DB2 Information Integrator Release Notes and Installation Requirements

Name	File name	Location
Installation Requirements for IBM DB2 Information Integrator Event Publishing Edition, Replication Edition, Standard Edition, Advanced Edition, Advanced Edition Unlimited, Developer Edition, and Replication for z/OS	Prereqs	• The *DB2 Information Integrator* product CD • DB2 Information Integrator Installation Launchpad
Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS	ReleaseNotes	• In the DB2 Information Center, **Product Overviews** > **Information Integration** > **DB2 Information Integrator overview** > **Problems, workarounds, and documentation updates** • DB2 Information Integrator Installation launchpad • DB2 Information Integrator Support Web site • The *DB2 Information Integrator* product CD
Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS	N/A	DB2 Information Integrator Support Web site

Table 40. DB2 Information Integrator Release Notes and Installation Requirements (continued)

Name	File name	Location
Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS	N/A	DB2 Information Integrator Support Web site
Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS	N/A	DB2 Information Integrator Support Web site
Release Notes for Enterprise Search	N/A	DB2 Information Integrator Support Web site

To view the installation requirements and release notes that are on the product CD:

- On Windows operating systems, enter:

  ```
  x:\doc\%L
  ```

  x is the Windows CD drive letter and *%L* is the locale of the documentation that you want to use, for example, en_US.
- On UNIX operating systems, enter:

  ```
  /cdrom/doc/%L/
  ```

  cdrom refers to the UNIX mount point of the CD and *%L* is the locale of the documentation that you want to use, for example, en_US.

Accessibility

Accessibility features help users with physical disabilities, such as restricted mobility or limited vision, to use software products successfully. The following list specifies the major accessibility features in DB2® Version 8 products:

- All DB2 functionality is available using the keyboard for navigation instead of the mouse. For more information, see "Keyboard input and navigation."
- You can customize the size and color of the fonts on DB2 user interfaces. For more information, see "Accessible display."
- DB2 products support accessibility applications that use the Java™ Accessibility API. For more information, see "Compatibility with assistive technologies" on page 98.
- DB2 documentation is provided in an accessible format. For more information, see "Accessible documentation" on page 98.

Keyboard input and navigation

Keyboard input

You can operate the DB2 tools using only the keyboard. You can use keys or key combinations to perform operations that can also be done using a mouse. Standard operating system keystrokes are used for standard operating system operations.

For more information about using keys or key combinations to perform operations, see Keyboard shortcuts and accelerators: Common GUI help.

Keyboard navigation

You can navigate the DB2 tools user interface using keys or key combinations.

For more information about using keys or key combinations to navigate the DB2 Tools, see Keyboard shortcuts and accelerators: Common GUI help.

Keyboard focus

In UNIX® operating systems, the area of the active window where your keystrokes will have an effect is highlighted.

Accessible display

The DB2 tools have features that improve accessibility for users with low vision or other visual impairments. These accessibility enhancements include support for customizable font properties.

Font settings

You can select the color, size, and font for the text in menus and dialog windows, using the Tools Settings notebook.

For more information about specifying font settings, see Changing the fonts for menus and text: Common GUI help.

Non-dependence on color

You do not need to distinguish between colors in order to use any of the functions in this product.

Compatibility with assistive technologies

The DB2 tools interfaces support the Java Accessibility API, which enables you to use screen readers and other assistive technologies with DB2 products.

Accessible documentation

Documentation for DB2 is provided in XHTML 1.0 format, which is viewable in most Web browsers. XHTML allows you to view documentation according to the display preferences set in your browser. It also allows you to use screen readers and other assistive technologies.

Syntax diagrams are provided in dotted decimal format. This format is available only if you are accessing the online documentation using a screen-reader.

Related concepts:

- "Dotted decimal syntax diagrams" in the *Infrastructure Topics (DB2 Common Files)*

Related tasks:

- "Keyboard shortcuts and accelerators: Common GUI help"
- "Changing the fonts for menus and text: Common GUI help"

Notices

This information was developed for products and services offered in the U.S.A. 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 106-0032, 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 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 Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

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

COPYRIGHT LICENSE:

This information contains sample application programs, in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. 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 IBM's application programming interfaces.

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:

IBM
AIX
DataJoiner
DB2
DB2 Universal Database
DRDA
IMS
Informix
iSeries
Lotus
OS/390
UNIX
Windows
z/OS

The following terms are trademarks or registered trademarks of other companies:

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

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

Intel, Intel Inside (logos), MMX and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

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

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

Index

Contacting IBM

To contact IBM customer service in the United States or Canada, call 1-800-IBM-SERV (1-800-426-7378).

To learn about available service options, call one of the following numbers:

- In the United States: 1-888-426-4343
- In Canada: 1-800-465-9600

To locate an IBM office in your country or region, see the IBM Directory of Worldwide Contacts on the Web at www.ibm.com/planetwide.

Product information

Information about DB2 Information Integrator is available by telephone or on the Web.

If you live in the United States, you can call one of the following numbers:

- To order products or to obtain general information: 1-800-IBM-CALL (1-800-426-2255)
- To order publications: 1-800-879-2755

On the Web, go to www.ibm.com/software/data/integration/db2ii/support.html. This site contains the latest information about:

- The technical library
- Ordering books
- Client downloads
- Newsgroups
- Fix packs
- News
- Links to Web resources

Comments on the documentation

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

- Send your comments using the online readers' comment form at www.ibm.com/software/data/rcf.
- Send your comments by e-mail to comments@us.ibm.com. 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 title, a table number, or a page number).

Printed in USA

SC18-7360-01