

Migration Guide
Updated March, 2008

Note

Before using this information and the product it supports, read the general information under Appendix C, "Notices," on page 175.

Edition Notice

This document contains proprietary information of IBM. It is provided under a license agreement and is protected by copyright law. The information contained in this publication does not include any product warranties, and any statements provided in this manual should not be interpreted as such.

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

To order DB2 publications from DB2 Marketing and Sales in the United States or Canada, call 1-800-IBM-4YOU (426-4968).

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

© Copyright International Business Machines Corporation 2006, 2008. All rights reserved.

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

Contents

About this book v

Part 1. Migrating your DB2 environment 1

Chapter 1. Migration to DB2 Version 9.5 3

Chapter 2. Migration planning for your DB2 environment. 5

Planning migration for your DB2 servers 7

Planning migration for your clients. 9

Planning migration for your database applications and routines 10

Part 2. Migrating DB2 servers 13

Chapter 3. Migration for DB2 servers 15

Chapter 4. Migration essentials for DB2 servers 17

What gets migrated. 17

Migration restrictions for DB2 servers 18

Migration best practices for DB2 servers. 21

Disk space requirements for DB2 server migration 24

Support changes for 32-bit and 64-bit DB2 servers 25

DB2 server behavior changes 26

Deprecated or discontinued functionality that impacts DB2 server migration 35

Migration from non-DB2 relational database management systems 36

Chapter 5. Pre-migration tasks for DB2 servers 39

Verifying that your databases are ready for migration 40

Revoking the EXECUTE privilege on migrated routines from PUBLIC. 41

Backing up databases before migration 42

Backing up DB2 server configuration and diagnostic information 43

Increasing table space and log file sizes before migration 44

Changing raw devices to block devices (Linux) . . . 46

Migrating DB2 servers in a test environment . . . 48

 Creating database duplicates 49

 Taking a DB2 server offline before migration . . . 49

Chapter 6. Migrating a DB2 server (Windows) 51

Migrating instances. 52

Migrating the DB2 Administration Server (DAS) . . 54

Migrating databases 55

Chapter 7. Migrating a DB2 server (Linux and UNIX) 59

Migrating instances. 60

Migrating the DB2 Administration Server (DAS) . . 61

Migrating databases 63

Chapter 8. Migrating environments with specific characteristics. 65

Migrating DB2 32-bit servers to 64-bit systems (Windows). 65

Migrating to a new DB2 server. 66

Migrating partitioned database environments . . . 69

Migrating from a DB2 server with multiple DB2 copies 70

Migrating DB2 servers in Microsoft Cluster Server environments. 72

Migrating DB2 Data Links Manager environments 73

Migrating from XML Extender to native XML data store. 74

Chapter 9. Post-migration tasks for DB2 servers 77

Adjusting the log space size in migrated databases 78

Activating a database after migration. 79

Managing DB2 server behavior changes 80

Setting up security to manage database auditing in migrated databases. 81

Converting type-1 indexes to type-2 indexes in migrated databases. 82

Rebinding packages in migrated databases 83

Migrating explain tables 84

Ensuring system temporary table spaces page sizes meet requirements 84

Re-creating write-to-table event monitors 86

Verifying migration of DB2 servers 86

Chapter 10. Enabling new DB2 Version 9.5 functionality in migrated databases. 89

Chapter 11. Reversing DB2 server migration. 91

Part 3. Migrating clients 93

Chapter 12. Migration for clients. 95

Chapter 13. Migration essentials for clients 97

Migration best practices for clients. 98

Chapter 14. Pre-migration tasks for clients 101

Backing up client configuration information	101
Migrating clients in a test environment	102

Chapter 15. Migrating to Data Server Client (Windows) 103

Chapter 16. Migrating to Data Server Runtime Client (Windows) 105

Chapter 17. Migrating clients (Linux and UNIX) 107

Chapter 18. Post-migration tasks for clients 109

Recataloging nodes and databases using TCP/IP protocol	109
Verifying migration of clients	110

Part 4. Migrating applications and routines 113

Chapter 19. Migration for database applications and routines 115

Chapter 20. Migration essentials for database applications. 117

Changed APIs and data structures	122
Migration impact of DB2 command changes	124
Migration impact of SQL statement changes	127
Migration impact of system catalog changes	128

Chapter 21. Migration essentials for routines. 135

Chapter 22. Pre-migration tasks for database applications and routines . . 137

Chapter 23. Migrating database applications 139

Migrating embedded SQL applications	140
Migrating CLI applications	141
Migrating Java applications that use IBM Data Server Driver for JDBC and SQLJ.	142
Migrating Java applications that use DB2 JDBC Type 2 driver	144

Migrating ADO.NET applications	145
Migrating scripts	146
Migrating 32-bit database applications to run on 64-bit instances.	147

Chapter 24. Migrating routines 149

Migrating C, C++, and COBOL routines	150
Migrating Java routines	152
Migrating .NET CLR routines	153
Migrating SQL procedures	154
Migrating 32-bit external routines to run on 64-bit instances	155

Chapter 25. Post-migration tasks for database applications and routines . . 157

Chapter 26. Enabling new DB2 Version 9.5 functionality in database applications and routines 159

Part 5. Appendixes 161

Appendix A. Important references . . . 163

Appendix B. Overview of the DB2 technical information 165

DB2 technical library in hardcopy or PDF format	165
Ordering printed DB2 books	168
Displaying SQL state help from the command line processor	168
Accessing different versions of the DB2 Information Center	169
Displaying topics in your preferred language in the DB2 Information Center	169
Updating the DB2 Information Center installed on your computer or intranet server	170
DB2 tutorials	171
DB2 troubleshooting information	172
Terms and Conditions	172

Appendix C. Notices 175

Index 179

About this book

The Migration Guide describes the migration process and concepts for each component of your DB2® environment. These components are DB2 servers, DB2 clients, database applications, and routines.

Who should use this book

This book is intended for database administrators, system administrators, and system operators who need to migrate DB2 servers and DB2 clients. It is also intended for programmers and other users who need to migrate database applications and routines.

How this book is structured

This book contains information on how to create a migration plan and how to migrate each component of your DB2 environment:

- Part 1, “Migrating your DB2 environment,” on page 1
- Part 2, “Migrating DB2 servers,” on page 13
- Part 3, “Migrating clients,” on page 93
- Part 4, “Migrating applications and routines,” on page 113

Part 1. Migrating your DB2 environment

This part of the book contains the following chapters:

- Chapter 1, “Migration to DB2 Version 9.5,” on page 3
- Chapter 2, “Migration planning for your DB2 environment,” on page 5

Chapter 1. Migration to DB2 Version 9.5

Upgrading to a new release of DB2 database product might require migration of your DB2 environment components if you want them to run on the new release.

Your DB2 environment has several components such as DB2 servers, DB2 clients, database applications, and routines. Migrating these components requires an understanding of DB2 database products and their migration concepts. For example, if you have an existing DB2 environment with DB2 Version 9.1 or DB2 UDB Version 8 copies and you want to upgrade them to DB2 Version 9.5, then you must migrate your DB2 environment.

The migration process consists of all the tasks that you need to perform to have your DB2 environment running successfully on a new release. The migration of each of the components in your DB2 environment requires that you perform different tasks:

- Migrating DB2 servers involves migrating your existing instances and databases so that they can run in the new release.
- Migrating clients involves migrating your client instances to keep the configuration of your existing clients.
- Migrating database applications and routines involves testing them in the new release and modifying them only when you need to support changes in this new release.

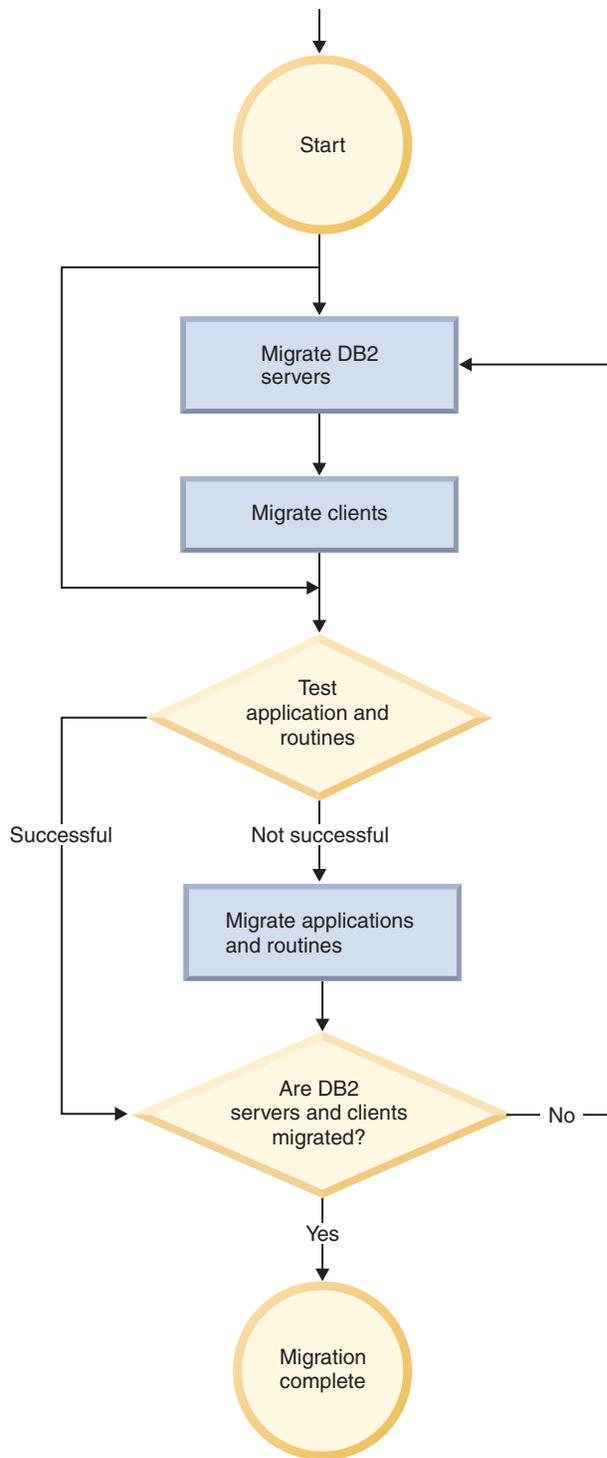
The following information is provided to document the migration process for DB2 Version 9.5:

- Migration overviews define migration concepts and describe the migration process for a component.
- Migration essentials include the details about migration support, restrictions and recommendations that you need to know to plan your migration strategy.
- Pre-migration tasks describe all the preparation tasks that you need to perform before migration.
- Migration tasks describe step by step the basic migration process for a component and how to migrate DB2 environments with special characteristics.
- Post-migration tasks describe all the tasks that you need to perform after migration to have your DB2 server running at the optimum level.

Chapter 2. Migration planning for your DB2 environment

Your environment has several components such as DB2 servers, DB2 clients, database applications, scripts, routines and tools. Planning your migration requires a thorough understanding of the migration process for each component in your environment.

First, you need to devise a strategy on how to approach the migration of your environment. You need to determine the order in which you are going to migrate each component. The characteristics of your environment and the information in migration essentials, especially the migration recommendations and restrictions, can help you determine your strategy. The following graphic depicts the recommended migration roadmap for the components in your environment:



The following is an example of a good *migration strategy* in which you test your database applications and routines and determine that they run successfully in DB2 Version 9.5:

1. Set up a DB2 Version 9.5 test server and create test databases.
2. Test your database applications and routines on a DB2 Version 9.5 test database to determine whether they run successfully. If your application requires a client, use a Version 9.5 client.

3. Migrate your DB2 servers and clients in a test environment. Determine what the migration issues are and how to resolve them. Use this information to adjust your migration plan.
4. Migrate your DB2 servers to DB2 Version 9.5 in your production environment. Ensure that they operate as expected.
5. Migrate your clients to DB2 Version 9.5 in your production environment. Ensure that your clients operate as expected.
6. Test your database applications and routines in the DB2 Version 9.5 migrated environment to determine whether they run as expected.
7. Make your migrated environment available to users.
8. Identify the use of deprecated features that will eventually become discontinued and new features that can improve the functionality and performance of your applications and routines. Plan how to modify your applications and routines.
9. Modify your database applications and routines as planned. Ensure that they run successfully in DB2 Version 9.5.

Once you have a strategy that will give you the outline for your migration plan, you can define the migration plan details for each component in your environment. A *migration plan* should include for each component:

- Migration prerequisites
- Pre-migration tasks
- Migration tasks
- Post-migration tasks

If you have previous migration plans, review them and compare them with the migration plan for DB2 Version 9.5. Include in your new plan any steps related to internal procedures to request access, software installation or other system services within your organization.

Review also the DB2 migration portal that provides access to additional resources and up-to-date information about the migration process as they become available at <http://www.ibm.com/support/docview.wss?rs=73&uid=swg21200005>. These resources include white papers and sample scripts for migration.

Finally, plan to remove the use of deprecated features and incorporate new features from DB2 Version 9.5. Although you are only required to remove the use of unsupported features, you should also plan to remove the use of deprecated features after migration because they will become unsupported in a future release. Also, you should take advantage of new features for your database products, applications, and routines to enhance functionality and improve performance.

Planning migration for your DB2 servers

Planning the migration of DB2 servers requires that you review all of the applicable migration prerequisites, pre-migration tasks, migration tasks and post-migration tasks.

Procedure

To create a migration plan for your DB2 servers:

1. Write the migration plan for DB2 servers, using all of the details that apply to your environment:

Table 1. Migration plan details for DB2 servers.

Migration plan	Details
Prerequisites	<p>Ensure that you:</p> <ul style="list-style-type: none"> • meet the installation requirements for DB2 database products described in <i>Quick Beginnings for DB2 Servers</i>. • resolve any support issues described in migration essentials for DB2 servers. • meet all prerequisites for the migration task and subtasks, especially obtaining root or Local Administrator access and required DB2 authorization.
Pre-migration tasks	<p>Include the following tasks:</p> <ul style="list-style-type: none"> • Migrate your DB2 server in a test environment to determine any migration issues • Verify that databases are ready for DB2 migration • Back up your databases • Back up configuration information • Increase table space and log file sizes • If the diaglevel database manager configuration parameter is set to 2 or less, set this parameter to 3 or higher. • Take the DB2 server offline for DB2 migration <p>In addition, check the list of pre-migration tasks for optional tasks that you might want to perform for your environment.</p>
Migration task	<p>You must include these steps:</p> <ul style="list-style-type: none"> • Install DB2 Version 9.5 • Migrate instances • Migrate the DAS • Migrate databases <p>Review the following migration tasks to determine the additional steps that are required to migrate your environment:</p> <ul style="list-style-type: none"> • Migrating a DB2 server (Windows[®]) • Migrating a DB2 server (Linux[®] and UNIX[®]) • Migrating environments with specific characteristics <p>Take note of the time required to migrate your databases.</p>

Table 1. Migration plan details for DB2 servers. (continued)

Migration plan	Details
Post-migration tasks	<ul style="list-style-type: none"> • Reset the <i>diaglevel</i> database manager configuration parameter to the value set before the migration • Adjust the log space size • Activate your database after migration • Manage changes in DB2 server behavior • Convert type-1 indexes to type-2 indexes in migrated databases • Rebind packages in migrated databases • Migrate DB2 Explain tables • Update the statistics on the system catalog tables. Update the statistics on the system catalog tables. • Verify that your DB2 server migration was successful • Back up your databases after the migration is complete <p>In addition, check the list of post-migration tasks for optional tasks that you might want to perform for your environment. Consider adding the following tasks to your migration plan:</p> <ul style="list-style-type: none"> • Tune your DB2 server once migration is completed. See “Developing a performance improvement process” in <i>Tuning Database Performance</i> • Remove the use of deprecated features in DB2 Version 9.5 • Enable new features in migrated databases, where appropriate, to improve performance at the DB2 server level. Review manageability, performance, and scalability enhancements in What’s New to determine what new features you might want to apply to your environment

2. If you must be able to reverse the migration, add details to the plan about the tasks required to reverse a DB2 server migration. These details should include any steps required in the migration task that enables you to reverse the migration.
3. Combine with the migration plan for other components such as clients, database applications, and routines to create an overall migration plan for your DB2 environment.

Planning migration for your clients

Planning the migration of clients requires that you review all of the applicable migration prerequisites, pre-migration tasks, migration tasks and post-migration tasks.

Procedure

To create a migration plan for your clients:

1. Write the migration plan for clients, using all the details that apply to your environment:

Table 2. Migration plan details for clients.

Migration plan	Details
Prerequisites	<p>Ensure that you:</p> <ul style="list-style-type: none"> • meet the installation requirements for DB2 database products described in <i>Quick Beginnings for DB2 Servers</i>. • resolve any support issues in migration essentials for clients including client and server connectivity. • meet all prerequisites for the migration task and subtasks, especially obtaining root or Local Administrator access and required DB2 authorization.
Pre-migration tasks	<ul style="list-style-type: none"> • Migrate your DB2 servers • Back up your client configuration information
Migration task	<p>You must include these steps:</p> <ul style="list-style-type: none"> • Install Version 9.5 client • Migrate client instance <p>Review the following migration tasks to determine the additional steps that are required to migrate your environment:</p> <ul style="list-style-type: none"> • Migrating to Data Server Client (Windows) • Migrating to Data Server Runtime Client (Windows) • Migrating clients (Linux and UNIX)
Post-migration tasks	<ul style="list-style-type: none"> • Recatalog nodes and databases that use NetBIOS and SNA protocols • Review changes in DB2 server behavior • Verify that migration for clients was successful

2. Combine with the migration plan for other components such as DB2 servers, database applications, and routines to create an overall migration plan.

Planning migration for your database applications and routines

Planning the migration of database applications and routines requires that you review all of the applicable pre-migration tasks, migration prerequisites, migration tasks, and post-migration tasks.

Procedure

To create a migration plan for your database applications and routines:

1. Write the migration plan for database applications, using all the details that apply to your environment:

Table 3. Migration plan details for database applications.

Migration plan	Details
Prerequisites	<p>Ensure that you:</p> <ul style="list-style-type: none"> • meet the installation requirements for DB2 database products described in <i>Quick Beginnings for DB2 Servers</i>. • meet the new development software requirements. • resolve any support issues in migration essentials for database applications during migration. • meet all prerequisites for the migration task and sub-tasks, especially obtaining required DB2 authorization.

Table 3. Migration plan details for database applications. (continued)

Migration plan	Details
Pre-migration tasks	<p>Include the following tasks:</p> <ul style="list-style-type: none"> • Migrate your client or install Version 9.5 application driver. • Test your database applications in a DB2 Version 9.5 testing environment. If your applications run successfully, the rest of migration steps are not required. <p>In addition, check the list of pre-migration tasks for optional tasks that you might want to perform for your environment. Even if your current operating system and development software are supported, consider including the following tasks to improve application performance:</p> <ul style="list-style-type: none"> • Upgrade your operating system to the latest supported level • Upgrade your development software to the latest supported level
Migration task	<p>You must include these steps:</p> <ul style="list-style-type: none"> • Modify your application code to support changes in DB2 Version 9.5 and to remove use of features that are discontinued in DB2 Version 9.5. • Modify your application to support changes specific to the development environment. • Rebuild all database applications after completing your modifications. • Test your database applications using DB2 Version 9.5. <p>Review the following migration tasks to determine the additional steps that are required by your development environment to migrate database applications:</p> <ul style="list-style-type: none"> • “Migrating embedded SQL applications” on page 140 • “Migrating CLI applications” on page 141 • “Migrating Java applications that use IBM Data Server Driver for JDBC and SQLJ” on page 142 • “Migrating Java applications that use DB2 JDBC Type 2 driver” on page 144 • “Migrating ADO.NET applications” on page 145 • “Migrating scripts” on page 146 • “Migrating 32-bit database applications to run on 64-bit instances” on page 147
Post-migration tasks	<p>Perform the recommended post-migration tasks for database applications, especially:</p> <ul style="list-style-type: none"> • Tune performance of your database applications and routines. • Remove the use of deprecated functionality in DB2 Version 9.5. • Implement the use of new features in DB2 Version 9.5 for application development where appropriate.

2. Write the migration plan for routines, using all the details that apply to your environment:

Table 4. Migration plan details for routines.

Migration plan	Details
Prerequisites	<p>Ensure that you:</p> <ul style="list-style-type: none"> • meet the new development software requirements. • resolve any support issues in migration essentials for routines during migration. • meet all prerequisites for the migration task and sub-tasks, especially obtaining required DB2 authorization.
Pre-migration tasks	<p>Include the following task:</p> <ul style="list-style-type: none"> • Test your routines in a DB2 Version 9.5 testing environment. If your routines run successfully, the rest of migration steps are not required. <p>In addition, check the list of pre-migration tasks for optional tasks that you might want to perform for your environment. Even if your development software is supported, consider upgrading your development software to the latest supported level.</p>
Migration task	<p>You must include these steps:</p> <ul style="list-style-type: none"> • Modify your routines to support changes in DB2 Version 9.5 and to remove use of features that are discontinued in DB2 Version 9.5. • Modify your routines to support changes specific to the development environment. • Rebuild all external routines after completing your modifications. • Retest your routines using DB2 Version 9.5. <p>Review the following migration tasks to determine the additional steps that are required by your development environment to migrate routines:</p> <ul style="list-style-type: none"> • “Migrating C, C++, and COBOL routines” on page 150 • “Migrating Java routines” on page 152 • “Migrating .NET CLR routines” on page 153 • “Migrating SQL procedures” on page 154 • “Migrating 32-bit external routines to run on 64-bit instances” on page 155
Post-migration tasks	<p>Perform the recommended post-migration tasks for routines, especially:</p> <ul style="list-style-type: none"> • Remove the use of deprecated functionality in DB2 Version 9.5 • Implement the new features in DB2 Version 9.5 for routines where appropriate

3. Combine with the migration plan for other components such as clients and DB2 servers to create an overall migration plan.

Part 2. Migrating DB2 servers

This part of the book contains the following chapters:

- Chapter 3, “Migration for DB2 servers,” on page 15
- Chapter 4, “Migration essentials for DB2 servers,” on page 17
- Chapter 5, “Pre-migration tasks for DB2 servers,” on page 39
- Chapter 6, “Migrating a DB2 server (Windows),” on page 51
- Chapter 7, “Migrating a DB2 server (Linux and UNIX),” on page 59
- Chapter 8, “Migrating environments with specific characteristics,” on page 65
- Chapter 9, “Post-migration tasks for DB2 servers,” on page 77
- Chapter 10, “Enabling new DB2 Version 9.5 functionality in migrated databases,” on page 89
- Chapter 11, “Reversing DB2 server migration,” on page 91

Chapter 3. Migration for DB2 servers

Upgrading to DB2 Version 9.5 requires that you migrate your DB2 servers.

Migrating your existing DB2 server requires that you install a DB2 Version 9.5 copy and then migrate all the instances and databases to be able to run them under the DB2 Version 9.5 copy.

You can directly migrate existing DB2 Version 9.1 or DB2 UDB Version 8 instances and databases to DB2 Version 9.5. Learn details, limitations about the migration process, and possible issues that you need to be aware of in the migration essentials section. Refer to the migrating a DB2 server tasks for details on how to migrate to DB2 Version 9.5.

On Windows operating systems, you have an option to automatically migrate an existing DB2 Version 9.1 or DB2 UDB Version 8 copy. If you choose to migrate your existing DB2 copy during installation, you only need to migrate your databases after installation.

If your DB2 servers are running on a release prior to DB2 UDB Version 8, you need to migrate them first to DB2 UDB Version 8, and then migrate to DB2 Version 9.5. It is recommended that you migrate to the latest fix pack of DB2 UDB Version 8.2. Refer to the DB2 UDB Version 8 migration roadmap for details on how to migrate to DB2 UDB Version 8.2.

Migration to DB2 Version 9.5 is supported for the following DB2 products:

- DB2 Enterprise Server Edition Version 9.1
- DB2 Workgroup Server Edition Version 9.1
- DB2 Personal Edition Version 9.1
- DB2 Express Edition Version 9.1
- DB2 Connect™ Enterprise Edition Version 9.1
- DB2 Connect Personal Edition Version 9.1
- DB2 Connect Unlimited Edition Version 9.1
- DB2 Connect Application Server Edition Version 9.1
- DB2 Client Version 9.1
- DB2 Runtime Client Version 9.1
- DB2 Query Patroller Version 9.1
- DB2 UDB Enterprise Server Edition Version 8
- DB2 UDB Workgroup Server Unlimited Edition Version 8
- DB2 UDB Workgroup Server Edition Version 8
- DB2 UDB Personal Edition Version 8
- DB2 Express Edition Version 8
- DB2 Universal Developer's Edition Version 8
- DB2 Personal Developer's Edition Version 8
- DB2 UDB Express Edition Version 8
- DB2 Connect Unlimited Edition Version 8
- DB2 Connect Enterprise Edition Version 8
- DB2 Connect Application Server Edition Version 8

- DB2 Connect Personal Edition Version 8
- DB2 Administration Client Version 8
- DB2 Application Development Client Version 8
- DB2 Runtime Client Version 8
- DB2 Query Patroller Version 8

For DB2 products not supported, refer to functionality deprecated or discontinued in DB2 database products that impacts migration.

Chapter 4. Migration essentials for DB2 servers

Migrating DB2 servers to DB2 Version 9.5 requires an understanding of migration concepts, migration restrictions, migration recommendations, and your DB2 server. Once you have a complete understanding of what migrating your DB2 server involves, you can create your own migration plan.

Consider the following factors to develop a complete understanding of migrating DB2 servers to DB2 Version 9.5:

- “What gets migrated”
- “Migration restrictions for DB2 servers” on page 18
- “Migration best practices for DB2 servers” on page 21
- “Disk space requirements for DB2 server migration” on page 24
- “Support changes for 32-bit and 64-bit DB2 servers” on page 25
- “DB2 server behavior changes” on page 26
- “Deprecated or discontinued functionality that impacts DB2 server migration” on page 35
- “Migration from non-DB2 relational database management systems” on page 36

What gets migrated

Learning what actions take place when you invoke the commands to migrate instances and databases give you a better understanding of the migration process for DB2 servers.

When the instance migration is called explicitly using the `db2imigr` command, or implicitly when you install DB2 Version 9.5 on Windows and select the migration action in Work with Existing, this command:

- Migrates an existing instance to a new instance under a DB2 Version 9.5 copy.
- Migrates instance profile registry variables. The global profile registry variables set by the user are not migrated.
- Migrates the database manager configuration (`dbm cfg`) file.
- Sets the `jdk_path` database manager (`dbm cfg`) parameter appropriately.
- Migrates the `db2audit.cfg` audit configuration file when the audit facility is enabled.
- Migrates the `SSLconfig.ini` SSL configuration file and migrates the instance profile registry setting “`DB2COMM=SSL`”.
- In a Microsoft® Cluster Server (MSCS) environment, defines a new resource type, updates all DB2 MSCS resources to use the new resource type, removes the old resource type and brings all resources online.

For a successful instance migration, it is essential that all files for all instances exist and that write access is granted. However you need to review migration restrictions for particular scenarios that are not supported.

When you access the database directory the first time, it is implicitly migrated if necessary. The database directory is accessed when you issue commands such as `LIST DATABASE DIRECTORY` or `MIGRATE DATABASE` command.

When the database migration is called explicitly using the MIGRATE DATABASE command, or implicitly using the RESTORE DATABASE command from a DB2 Version 9.1 or DB2 UDB Version 8 backup, the following database entities might be converted during database migration:

- Database configuration file
- Log file header
- Table root page
- Index root page
- Catalog tables
- Buffer pool files
- History file

Migration restrictions for DB2 servers

Before you start to migrate your DB2 server, you need to understand what the support for migration is and what the restrictions are.

What is supported?

- Migration is supported from DB2 Version 9.1 and DB2 UDB Version 8. If you have DB2 UDB Version 7 or earlier, you need to migrate to DB2 UDB Version 8 before migrating to DB2 Version 9.5.
- On Windows operating systems, there is a migrate action available to automatically migrate an existing DB2 UDB Version 8 or DB2 Version 9.1 copy during the installation of DB2 Version 9.5. This action automatically migrates all of your instances and your DB2 Administration Server (DAS) running on the existing DB2 copy, and uninstalls your existing DB2 copy and any add-on products installed in this copy. If you do not choose the migrate action, you must manually migrate your instances and your DAS after installation.
- On Linux and UNIX operating systems, you can install only a new copy of DB2 Version 9.5. You have to manually migrate your instances after installation. You can manually migrate your existing DAS.
- Instance bit size is determined by the operating system where DB2 Version 9.5 is installed, and support for 32-bit kernels and 64-bit kernels has changed.
- You can only migrate to a root installation of DB2 Version 9.5. Migration is not supported from a DB2 Version 9.1 or a DB2 UDB Version 8 copy to a non-root installation of DB2 Version 9.5.
- Migration is supported from a system with multiple DB2 copies of DB2 Version 9.1, DB2 UDB Version 8, or both levels. On Windows operating systems, you need to be aware of the restrictions on coexistence of previous versions of the DB2 database products. See “Updating DB2 copies (Windows)” in *Data Servers, Databases, and Database Objects Guide*.
- Migration is supported from a partitioned database environment with multiple database partitions.
- Restoring full database offline backups from DB2 UDB Version 8 or DB2 Version 9.1 is supported. However, rolling forward of logs from a previous level is not possible. Review “Backup and restore operations between different operating systems and hardware platforms” in *Data Recovery and High Availability Guide and Reference* for complete details about migration support using the RESTORE DATABASE command.

- When the DB2 Workload Manager feature is installed, a migrated database has three predefined default services classes: the default user class, the default maintenance class, and the default system class. After database migration, all connections belong to the default workload. Connections that belong to the default workload are mapped to the default user class. The default user class has only one service subclass: the default subclass. All activities from the connections in the default user class run in the default subclass.
- In migrated databases with the **RESTRICT_ACCESS** database configuration parameter set to YES, you must grant the USAGE privilege to non-DBADM users on SYSDEFAULTUSERWORKLOAD. Otherwise, these users are unable to submit any work to the database.
- Index extensions are migrated as part of the database migration. However, you might need to re-create your indexes if you migrated from a DB2 UDB Version 8 or a DB2 Version 9.1 32-bit instance to a DB2 Version 9.5 64-bit instance. Review the post-migration tasks for details.

What is unsupported?

DB2 Version 9.5 installation fails if the following situations exist:

- The operating system is not supported. You need to upgrade to a supported version of the operating system before you migrate to DB2 Version 9.5 or migrate to a new DB2 server that meets the operating system requirements. See “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers*.
- A 32-bit kernel is running on Linux and UNIX operating systems except for Linux on x86. A 64-bit kernel must be installed prior to installing DB2 Version 9.5.
- A copy of DB2 UDB Version 7 or earlier is installed.

The db2imigr command fails if the following situations exist:

- You do not have authorization to migrate the instance.
- The instance that you are trying to migrate is active. Run the db2stop command to stop the instance.
- The instance is already at DB2 Version 9.5 or later. Run the db2iupdt command to update to a different fix pack levels or copies of DB2 Version 9.5.
- You try to migrate from DB2 Version 9.5 back to DB2 UDB Version 8 or DB2 Version 9.1. Chapter 11, “Reversing DB2 server migration,” on page 91 is possible, however, you must follow the prerequisites and steps in this procedure.
- The type of instance that you are trying to migrate to the DB2 Version 9.5 copy is unsupported. The following table describes the migration support for each type of instance by DB2 database product:

Table 5. Instance migration support for DB2 Version 9.5 database products

Instance type	Node type	Migration support
client – default type for DB2 clients ¹	Client	<ul style="list-style-type: none"> • Migration to a client, a standalone, a wse, or an ese instance is supported.
standalone – default type for DB2 Personal Edition (PE)	Database server with local clients	<ul style="list-style-type: none"> • Migration to a standalone, a wse, or an ese instance is supported. • Migration to a client instance is unsupported.

Table 5. Instance migration support for DB2 Version 9.5 database products (continued)

Instance type	Node type	Migration support
wse – default type for DB2 Workgroup Server Edition (WSE)	Database server with local and remote clients	<ul style="list-style-type: none"> • Migration to a wse or an ese instance is supported. • Migration to a standalone instance creates a standalone instance²(Linux and UNIX only) • Migration to a client instance is unsupported.
ese – default type for DB2 Enterprise Server Edition (ESE)	Partitioned database server with local and remote clients or Enterprise Server Edition with local and remote clients	<ul style="list-style-type: none"> • Migration to an ese instance is supported. • Migration to a standalone or a wse instance from single database partition environments creates a standalone or wse instance²(Linux and UNIX only) • Migration to a client instance is unsupported.

Note:

Specify the type of instance with the **-s** option of the db2icrt command. If you do not specify the **-s** option, the instance is created using the highest level of instance type supported by the DB2 database product installed.

1. The highest level for each DB2 database product is the default instance type as indicated in Table 5 on page 19 ordered from lower to higher-level. Each instance type supports instance types of a lower-level. For example, the ese instance type supports wse, standalone, and client. You can use the db2icrt command with the **-s** option to create instances of a lower-level.
 2. Database manager configuration parameters have default values for the created instance. Previous database manager configuration settings are not retained. If the configuration parameters are available in the new instance, after migration, you can restore previous settings. Avoid migrating from a higher-level instance type to a lower-level instance type if possible.
- The db2ckmig command fails and causes the db2imigr command to fail. The db2imigr command calls the db2ckmig command to verify whether catalogued local databases are ready for migration to DB2 Version 9.5.
 - DB2 Data Links Manager Version 8 is installed on the DB2 server. DB2 Data Links Manager is unsupported in DB2 Version 9.5. You can migrate to a standard DB2 Version 9.5 instance without the DB2 Data Links Manager functionality.
 - DB2 Data Warehouse Manager Version 8 and any extensions are installed on the DB2 server. DB2 Data Warehouse Manager is unsupported in DB2 Version 9.5. However, when you run the db2imigr command, the error message that is generated includes instructions on how to migrate to a standard DB2 Version 9.5 instance without the DB2 Data Warehouse Manager functionality.

The MIGRATE DATABASE command fails if the following situations exist:

- You do not have authorization to migrate the database.
- A catalogued database does not exist.

- Database migration encounters any of the problems described in the reason codes of error message “SQL1704N” in *Message Reference, Volume 2*.
- User-defined distinct types (UDTs) are encountered with the names ARRAY, BINARY, DECFLOAT, VARBINARY, or XML. You must drop these UDTs and re-create them with different names before database migration.
- User objects use the system-defined DATALINK data type. You must drop or alter these objects before database migration. Also, if you installed DB2 Net Search Extender (NSE) on your DB2 server, you must drop the UDFs created by NSE for Data Links support.
- A database enabled as a high availability disaster recovery (HADR) standby database.

Migration best practices for DB2 servers

Consider the following best practices when planning your DB2 server migration.

Review changes in existing DB2 database product functionality

Changes in existing functionality introduced in DB2 Version 9.5 can potentially impact your applications, scripts, maintenance processes, and any other aspects related your DB2 server migration process. If you are migrating from DB2 UDB Version 8, the changes in existing functionality introduced in DB2 Version 9.1 can also have an impact. You need to review these changes and plan how to address these changes before the migration. Migrating in a test environment allows you to learn about possible issues, evaluate the impact on your environment and find a resolution.

Perform hardware and operating system upgrades prior to DB2 database product migration

The support for UNIX, Linux and Windows operating systems has changed in DB2 Version 9.5. Review the “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers* to determine whether your operating system version is supported and if you need to upgrade your operating system before installing DB2 Version 9.5. Note that newer versions of operating systems can also bring new hardware requirements.

Even when you are not required but decide to upgrade, performing hardware and operating system upgrades separately from DB2 database product migration simplifies problem determination if you encounter migration difficulties. If you upgrade your software or hardware prior to a DB2 database product migration, ensure that your system is operating as expected before attempting the migration process.

If you have a DB2 Version 9.1 copy on Windows XP or Windows 2003, first apply a fix pack that supports Windows Vista before you upgrade the operating system to Windows Vista to ensure that your DB2 copy performs as expected after the operating system upgrade. The support for Windows Vista starts from DB2 Version 9.1 Fix Pack 2. If you have a DB2 UDB Version 8 copy on Windows XP or Windows 2003, first migrate to DB2 Version 9.5 and then upgrade the operating system to Windows Vista.

If you have a DB2 UDB Version 8.1 32-bit copy on Linux on POWER™, you need to update your current DB2 copy to DB2 UDB Version 8.1 FixPak 7 or higher and then upgrade your operating system to SUSE Linux Enterprise Server (SLES) 10 before installing DB2 Version 9.5.

Upgrade 32-bit Linux operating systems to 64-bit

If you are migrating to DB2 Version 9.5 32-bit database product on Linux operating systems, the new multithreaded architecture brings new restrictions due to the 32-bit virtual memory address limit such as:

- Agent private memory for all agent threads is now allocated within a single process. The process memory space might not be large enough to allocate the aggregate of all private memory for all agents. You might need to reduce the number of agents configured.
- Support for multiple databases is limited because all database shared memory segments for all databases are allocated in a single process memory space. You can reduce the memory usage for each database so that you can activate all databases successfully. However, the database server performance is impacted.

Consider migrating to DB2 Version 9.5 64-bit database product instead, to avoid running into any of the 32-bit kernel limitations.

Devise a plan to reverse a migration

There is no utility to reverse a migration from DB2 Version 9.5 to DB2 Version 9.1 or DB2 UDB Version 8. To reverse a database migration, you must re-create instances under DB2 Version 9.1 or DB2 UDB Version 8, and restore your DB2 Version 9.1 or DB2 UDB Version 8 database backups. Chapter 11, “Reversing DB2 server migration,” on page 91 to learn all the required steps.

Perform pre-migration tasks

There are several pre-migration tasks that you should execute for a successful migration, such as backing up DB2 configuration parameters settings, increasing table spaces and log files, and verifying that databases are ready for migration.

Migrate DB2 servers first

As you upgrade your environment to DB2 Version 9.5 from DB2 Version 9.1 or DB2 UDB Version 8, if you migrate your DB2 clients to DB2 Version 9.5 before you migrate all of your DB2 servers to DB2 Version 9.5, there are some restrictions and limitations such as support of new DB2 database product features, network protocols, and connectivity.

To avoid these known restrictions and limitations, migrate all of your DB2 servers to DB2 Version 9.5 before you migrate any of your DB2 clients to DB2 Version 9.5. These restrictions and limitations are not associated with DB2 Connect.

Migrate database applications and routines

If you migrate your DB2 server, you might also need to migrate your database applications and routines to support changes for 64-bit instances, SQL stored procedures, Java™ Virtual Machine (JVM), and development software.

Chapter 20, “Migration essentials for database applications,” on page 117 and Chapter 21, “Migration essentials for routines,” on page 135 describe the factors that can impact your database application migration or routine migration. Review these factors and make any necessary changes to your database applications and routines to ensure that they run after the migration to DB2 Version 9.5.

In a migration testing environment, you can test and verify that your database applications and routines run successfully in DB2 Version 9.5 to find out if you need to migrate them. You can also migrate your database applications and routines before you migrate your production environment.

Benchmark DB2 server performance

Run a number of performance tests before migrating your DB2 server. The db2batch benchmark tool helps you to collect elapsed and CPU times for running queries. You can use this tool to develop performance tests. Record the exact environment conditions where you run your tests.

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 that might occur.

Enable autonomic computing features

DB2 Version 9.5 enables several autonomic computing features when you create a database, such as automatic agent configuration and real-time statistics.

However, when you migrate your database to DB2 Version 9.5, agent configuration is not automatic and real-time statistics are not enabled. You should consider enabling these new features to gain performance and manageability improvements.

If you are migrating from DB2 UDB Version 8, DB2 Version 9.5 enables additional autonomic computing features introduced in Version 9.1 when you create a database:

- Automatic execution of the configuration advisor.
- Enablement of automatic storage.
- Enablement of the **auto_runstats** and **self_tuning_mem** database configuration parameters.

Refer to Enabling new DB2 Version 9.1 features in migrated databases for details.

Migration of an SQL replication environment

Migrating an SQL replication environment from DB2 Version 9.1 or DB2 UDB Version 8 requires that you prepare to migrate your DB2 servers, run pre-migration tasks for migration of an SQL replication environment, migrate DB2 servers, and use the migration tools to convert your SQL replication environment to DB2 Version 9.5.

For complete information on migrating your SQL replication environment, see the *Migrating to Replication Version 9.5 guide* available at <http://www.ibm.com/support/docview.wss?rs=71&uid=swg27009474>.

Migration of DB2 Spatial Extender

If you had DB2 Spatial Extender installed and you migrated your spatially-enabled databases to DB2 Version 9.5, refer to *Migrating to DB2 Spatial Extender Version 9.5* in *Spatial Extender and Geodetic Data Management Feature User's Guide and Reference* for migration details specific to DB2 Spatial Extender.

Migration of a Microsoft Cluster Server environment

In a Microsoft Cluster Server (MSCS) environment, you should install DB2 Version 9.5 as a new copy and then run the db2imigr command to migrate

the MSCS instance. Refer to “Migrating DB2 servers in Microsoft Cluster Server environments” on page 72 for details.

Disk space requirements for DB2 server migration

You need to be aware that the migration process requires additional disk space. Ensure that you have enough free disk space to complete this process successfully. The following disk space recommendations are applicable for migrating to DB2 Version 9.5.

Table space information files

The files SQLSPCS.1 and SQLSPCS.2 contain table space information. During migration from DB2 UDB Version 8 to DB2 Version 9.5, these files grow to four times their previous size but the total data size on disk does not exceed the new size of the SQLSPCS.1 and SQLSPCS.2 files. For example, if you have two files whose size totals 512 KB before migration, you will need at least 2 MB of free disk space.

System catalog and system temporary table spaces

Ensure that you have sufficient free space on the system catalog and the system temporary table spaces for the databases that you are migrating. System catalog table space is required for both old and new database catalogs during migration. The amount of free 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)

Increasing the total size to twice the total of used space is recommended. In other words the amount of free space should be at least the same as the current amount of used space.

Temporary table space (TEMPSPACE1 is the default name)

Increasing the total size to twice the total size of the system catalog table space is recommended.

For the system catalog table space, free pages should be equal to or greater than used pages. Total pages for the system temporary table space should be twice the amount of total pages for the system catalog table space.

To increase the amount of free space on your System Managed Space (SMS) table spaces, free sufficient disk space on the corresponding file systems or increase the size of your file systems if you are using a volume manager.

To increase the amount of free space on your Database Managed Space (DMS) table spaces, you can increase the size of existing containers. You can also add additional containers although this might trigger data rebalancing. You can reduce the size of the containers after migration.

Log file space

The database migration process makes changes to system catalog objects. All changes to a system catalog object are performed in a single transaction and need adequate log space to contain this transaction. If there is insufficient log space, this transaction is rolled back and migration does not complete successfully.

To ensure sufficient log file space is available, you can set the **logsecond** database configuration parameter to twice the current value of **logprimary** and **logsecond** if the file system containing the log files has enough disk

free space to increase this parameter. If you already have available a large log file space, it might not be necessary to increase this parameter. Also on partitioned database environments, you only need to increase the log space in the catalog database partition server.

You must update these database configuration parameters values before you migrate the instance to DB2 Version 9.5, because you will not be able to update these database configuration parameters until you issue the `MIGRATE DATABASE` command. If this command fails because there is insufficient log file space, then you can set these database configuration parameters to higher values and then re-issue the `MIGRATE DATABASE` command.

The new database configuration parameter settings for log space can be restored to their original value after the migration is complete.

Index space

Each index on every populated table requires one additional page per index to use the following functionality:

- Real-time statistics.
- Deferred cleanup roll out for MDC tables.
- Index rebuilt on a populated table.

If you have a limited amount of free disk space for indexes, you can get the error message `SQL0289N` that indicates the table space is full. Ensure that you have enough free pages in the corresponding index table space to account for one additional page per index on populated tables before:

- Populating tables in new databases created in DB2 Version 9.5, real-time statistics are enabled by default in these newly created databases.
- Enabling deferred cleanup roll out by setting `DB2_MDC_ROLLOUT` to `DEFER`, or when `DB2WORKLOAD` is set to `SAP`.
- Reorganizing or recreating indexes on populated tables.

Support changes for 32-bit and 64-bit DB2 servers

DB2 Version 9.5 provides support for 32-bit operating systems on Linux on x86 and Windows operating systems, and 64-bit operating systems on UNIX, Linux and Windows operating systems. Check installation requirements for details about supported architectures on each operating system.

You cannot specify the bit size for the instance when you create or migrate an instance. The bit size for new instances is determined by the operating system where DB2 Version 9.5 is installed. The following table summarizes the DB2 Version 9.5 bit size support that is available for each of the following operating systems:

Table 6. DB2 Version 9.5 32-bit and 64-bit support available per operating system.

Operating systems	DB2 Version 9.5 support available
<ul style="list-style-type: none"> • 32-bit Windows on x86 and X64 (Using DB2 Version 9.5 32-bit product) • 32-bit Linux on x86 	<ul style="list-style-type: none"> • 32-bit instances only • 32-bit DB2 server, client, and GUI tools packages • 32-bit IBM® Software Development Kit (SDK) for Java

Table 6. DB2 Version 9.5 32-bit and 64-bit support available per operating system. (continued)

Operating systems	DB2 Version 9.5 support available
<ul style="list-style-type: none"> • 64-bit kernels of AIX®, HP-UX, or Solaris • 64-bit Windows on X64 • 64-bit Linux kernel on x86-64, POWER, and zSeries® 	<ul style="list-style-type: none"> • 64-bit instances • 32-bit and 64-bit DB2 libraries available • 64-bit DB2 server and client • 64-bit applications and routines • 32-bit client side application support • 32-bit fenced stored procedures/UDFs only (non-Java) • Java fenced Stored Procedures/UDFs • 64-bit IBM SDK for Java

The changes in 32-bit and 64-bit support can have an impact in your applications depending on the shared library path that you indicated when you linked the DB2 libraries to your applications. If you specified the DB2 installation path, the applications fail to run because the DB2 Version 9.5 copy has a different installation path. However, if you linked the libraries using the library path under the instance home directory, your applications will run successfully in the following cases:

- If you have 32-bit instances and you migrate to DB2 Version 9.5 on a 32-bit system. You can only migrate to 32-bit instances on 32-bit Windows or 32-bit Linux on x86.
- If you have 64-bit instances and you migrate to DB2 Version 9.5 on a 64-bit system. You can only migrate to a 64-bit instance on a 64-bit system.

If you have 32-bit instances and you migrate to DB2 Version 9.5 on a 64-bit system, you need to manage incompatibilities due to the shared library path specification and discontinued features to run successfully your applications and routines. Table 6 on page 25 summarizes the details on the available support. For example, 32-bit unfenced stored procedures in any supported language except Java are not supported. By simply dropping and recreating these stored procedures as fenced you can resolve this issue quickly.

DB2 server behavior changes

Changes to DB2 registry variables, configuration parameters, database physical design characteristics, and database authorities and privileges can result in DB2 server behavior changes that might impact migration.

As a general rule, instance profile variables that you set in your DB2 profile registry or your system environment retain their values after instance migration. Some global profile registry variables, such as **DB2SYSTEM** and **DB2PATH**, are set by the DB2 installation procedure or instance migration. However, the global profile registry variables that you set by running the `db2set` command with the **-g** option are not migrated. Therefore, you need to define them after migration.

Existing database and database manager configuration parameters also, as a general rule, retain their values after migration. However, the default values assigned to new parameters or the new default values assigned to existing parameters could impact the behavior or performance of your applications.

The following tables describe in detail the migration impact of all of the changes to variables, database and database manager configuration parameters, physical design characteristics of databases, and database authorities and privileges:

- New registry variables
- Changes to existing registry variables
- Deprecated and discontinued variables
- Changes to existing database manager configuration parameters
- Deprecated database manager configuration parameters
- New database configuration parameters
- Changes to existing database configuration parameters
- Deprecated and discontinued database configuration parameters
- Changes to physical design characteristics of databases
- Changes to authorities and privileges

If you are migrating from DB2 UDB Version 8, you must be aware of all of the changes to variables, database and database manager configuration parameters, and physical design characteristics of databases between DB2 UDB Version 8 and DB2 Version 9.1 that might also impact your migration. The details are described in <http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.uprun.doc/doc/r0022380.htm>.

New registry variables

The following table describes the migration impact of the default values of new registry variables:

Table 7. New registry variables

Name	Migration impact
DB2FODC	This registry variable allows you to control what information is collected for problem determination by extending the functionality previously available in the DB2FFDC registry variable. After migration, DB2FODC is not set and the database manager uses any value set in DB2FFDC . If you assign a value to DB2FODC , the database manager ignores any value set in DB2FFDC .
DB2_SET_MAX_CONTAINER_SIZE	If the DB2WORKLOAD registry variable is set to SAP, DB2_SET_MAX_CONTAINER_SIZE is set 20 GB to limit the size of individual containers for automatic storage or database managed table spaces with the automatic resizing functionality enabled. Otherwise, the default value is OFF, which indicates that there is no limit.

Table 7. New registry variables (continued)

Name	Migration impact
DB2_SYSTEM_MONITOR_SETTINGS	<p>This registry variable allows you to controls a set of parameters which allow you to modify the behavior of various aspects of DB2 monitoring.</p> <p>In migrated instances, this variable is not set, which indicates:</p> <ul style="list-style-type: none"> • On Linux on RHEL4 and SLES9, the CPU usage is not read. • On Linux on RHEL5 and SLES10, the newer method of reading CPU usage returns only the user CPU usage value. • On any other operating system, the method of reading CPU usage returns both system and user CPU usage times. This is same method used in previous releases. <p>If you need to use the method of reading CPU usage that returns both system and user CPU usage times on Linux, perform one of the following actions:</p> <ul style="list-style-type: none"> • On Linux on RHEL4 and SLES9, set DISABLE_CPU_USAGE to TRUE. • On Linux on RHEL5 and SLES10, set OLD_CPU_USAGE to TRUE.
DB2_UPDDBC_CFG_SINGLE_DBPARTITION	<p>In migrated instances, this variable is not set, which indicates that the DB2 CLP commands and the DB2 APIs to update or reset the database configuration apply changes to all database partitions in a partitioned database environment. To maintain the same behavior as in previous releases, set this registry variable to TRUE to indicate that the DB2 CLP commands and DB2 APIs apply only to the local database partition or the database partition that is set by the DB2NODE registry variable:</p> <pre>db2set DB2_UPDDBC_CFG_SINGLE_DBPARTITION=TRUE</pre> <p>This variable setting has no influence when you call the ADMIN_CMD procedure to update or reset the database configuration.</p>

Changes to existing registry variables

The following table describes the migration impact of changes to existing registry variables:

Table 8. Changes to existing registry variables

Name	Migration impact
DB2CLIINIPATH	<p>If you choose the migrate action when installing DB2 Version 9.5 on Windows operating systems, DB2CLIINIPATH is set to the location of the existing db2cli.ini configuration file for the DB2 copy that you want to migrate. However, if you migrate your instances after installing a DB2 copy and you want to maintain the same location for your configuration file, set this variable to this location.</p>

Table 8. Changes to existing registry variables (continued)

Name	Migration impact
DB2_MDC_ROLLOUT	<p>If the DB2WORKLOAD registry variable is set to SAP, DB2_MDC_ROLLOUT is set to DEFER at instance migration to give SAP installations deferred index cleanup rollout by default. You can change this default by issuing the SET CURRENT MDC ROLLOUT MODE statement. Also, configuration of this registry variable is now dynamic, which means that when you change its value, any new compilations of a statement use the new value. To enable deferred index cleanup rollout, issue the following command:</p> <pre>db2set DB2_MDC_ROLLOUT=DEFER</pre> <p>See “Disk space requirements for DB2 server migration” on page 24 for information about additional index space needed by deferred index cleanup rollout.</p>
DB2_MMAP_READ and DB2_MMAP_WRITE	<p>For AIX operating systems, if you are running a level that is lower than AIX Version 5.3 Technology Level 6, the database manager ignores the value of these registry variables and considers memory mapped I/O as disabled. This requirement is due to a limitation with memory mapped I/O in threaded environment which is addressed starting in AIX 5.3 TL06. If you had memory mapped I/O enabled before migration or want to enable it after migration, you need to upgrade your AIX operating system to least AIX 5.3 TL06 or higher.</p>
DB2_USE_DB2JCCT2_JROUTINE	<p>By default in DB2 Version 9.5, this registry variable is not set and this means that the default JDBC driver to run JDBC routines is the IBM Data Server Driver for JDBC and SQLJ. By default in previous releases, this registry variable was not set but that meant that the default JDBC driver was the DB2 JDBC Type 2 driver. Refer to “Migrating Java routines” on page 152 for details about how to manage this change.</p>

Deprecated and discontinued registry variables

You should remove the use of registry variables that are deprecated because the functionality associated to the variable is obsolete or has been replaced by new functionality. Also, remove the use of discontinued registry variables as they do not have the intended effect. Refer to *Deprecated registry variables* and *Discontinued registry variables* in *What's New* to determine the migration impact of deprecated and discontinued registry variables.

Changes to existing database manager configuration parameters

The following table describes the migration impact of changes to database manager configuration parameters:

Table 9. Changes to existing database manager configuration parameters

Name	Migration impact
agent_stack_sz	<p>In DB2 Version 9.1, this parameter was only available on instances on Windows operating systems. This parameter now has new default values for migrated instances on Linux and UNIX operating systems. If your applications receives an insufficient stack error SQL00973N or you get a critical error in the db2diag.log file because of STACK OVERFLOW error, increase the value of this parameter.</p>
comm_bandwidth	<p>The setting for comm_bandwidth is not modified by instance migration. However, after migration, if you set the comm_bandwidth parameter to -1 to tune this parameter, the database manager attempts to detect the network link speed and sets this parameter to that speed. If the database manager fails to detect the link speed, it sets this parameter to 100, which is analogous to Gigabit Ethernet speed.</p> <p>DB2 Version 9.5 uses enhanced algorithms to determine a value for the comm_bandwidth parameter, the query compiler uses this value to determine an optimal access plan. You need to rebind database packages with static SQL so that the cost of the access plans use the new comm_bandwidth settings. Prior to rebinding database packages, consider updating statistics using the RUNSTATS command.</p>
fenced_pool	<p>The setting for fenced_pool is not modified by instance migration. However, configuration of this parameter is now dynamic and the maximum value is 64000. If you decrease this parameter value by issuing the UPDATE DBM CFG with the IMMEDIATE clause after migration, the database manager stops caching db2fmp threads or processes. If you increase this parameter value, the database manager starts caching more db2fmp threads and processes when they are created. Also, you can now set the fenced_pool to AUTOMATIC. If you set the fenced_pool and max_coordagents parameters to AUTOMATIC, the database manager increases the number of db2fmp threads and processes cached limited by the maximum value of the max_coordagents parameter. See the row in this table about max_coordagents and num_poolagents for more details.</p>
instance_memory	<p>Now, this configuration parameter refers to all private and shared memory heap allocations for the entire database partition. In migrated databases, this parameter is set to AUTOMATIC due to the parameter's change in meaning. The AUTOMATIC setting allows you to account for the increase in memory usage and changes in the memory model and allocate up to a value between 75% and 95% of the physical memory configured on the system divided by the number of configured local database partitions in the instance.</p>
java_heap_sz	<p>The setting for java_heap_sz is not modified by instance migration. However, the default value for HP-UX operating systems have increased to 4096. If the current value of this parameter is less than 4096, consider increasing this parameter to the new default value.</p>

Table 9. Changes to existing database manager configuration parameters (continued)

Name	Migration impact
max_coordagents and num_poolagents	<p>During instance migration, the max_coordagents parameter is set to the value of the maxagents parameter if the pre-migration value was -1 and the num_poolagents parameter is set to the value of the maxagents parameter divided by 2 if the pre-migration value was -1. Otherwise, the settings for the max_coordagents and num_poolagents parameters do not change. If you change the value of the maxagents parameter, the value of the max_coordagents and num_poolagents parameters do not change.</p> <p>After migration, unless you need a limit that cannot be exceeded, set the max_coordagents, max_connections, num_poolagents, and fenced_pool parameters to AUTOMATIC, as shown in the following examples, to ensure that the number of agents and connections are not limited by the values of memory parameters:</p> <pre>db2 UPDATE DBM CFG USING max_coordagents AUTOMATIC db2 UPDATE DBM CFG USING max_connections AUTOMATIC db2 UPDATE DBM CFG USING num_poolagents AUTOMATIC db2 UPDATE DBM CFG USING fenced_pool AUTOMATIC</pre>
max_connections	<p>The setting for max_connections is not modified by instance migration. If the limit indicated by the value of this parameter can be exceeded, set it to AUTOMATIC after migration. See the row in this table about max_coordagents and num_poolagents for more details.</p>
mon_heap_sz	<p>In migrated databases, this parameter is now set to AUTOMATIC to indicate a limit that could be exceeded and to maintain the same behavior as in previous releases. Any other value now represents a limit that cannot be exceeded. See Table 12 on page 33 for information about other database configuration parameters that are also set to AUTOMATIC.</p>
num_initagents	<p>The setting for num_initagents is not modified by instance migration. However, this parameter has a new range of values from 0 to 64000. If the num_poolagents parameter value is not AUTOMATIC or is greater than the num_initagents, the database manager starts the number of idle agents indicated by the num_initagents parameter value when you start your instance. Otherwise, the database manager starts the number of idle agents indicated by the num_poolagents parameter value.</p>

Deprecated database manager configuration parameters

The following table describes the migration impact of deprecated database manager configuration parameters:

Table 10. Deprecated database manager configuration parameters

Name	Migration impact
agentpri	<p>This database manager configuration parameter is deprecated. Use DB2 service classes to adjust agent priority.</p>
maxagents and maxcagents	<p>These database manager configuration parameters are deprecated. The GET DB CFG command output does not show these parameters, and using the UPDATE DB CFG command with these parameters has no effect. See the row in Table 9 on page 30 about max_coordagents and num_poolagents for more details regarding changes to database manager agents.</p>

Table 10. Deprecated database manager configuration parameters (continued)

Name	Migration impact
query_heap_sz	This database manager configuration parameter was used in previous releases to support Version 7 clients using DB2RA protocol. This parameter is deprecated because Version 7 clients are not supported in Version 9.5.

New database configuration parameters

The following table describes the migration impact of the default values of new database configuration parameters:

Table 11. New database configuration parameters

Name	Migration impact
appl_memory	This new configuration parameter is set by default to AUTOMATIC, meaning that all application memory requests are allowed if the total amount of memory allocated by the database partition is within the limits of the instance_memory configuration parameter setting.
auto_stmt_stats	You can enable or disable the automatic collection of real-time statistics at statement compilation time by using this configuration parameter. In migrated databases, this feature is disabled. To enable this feature in migrated databases, set this parameter to ON, and increase the value of the catalogcache_sz parameter. Refer to Chapter 10, "Enabling new DB2 Version 9.5 functionality in migrated databases," on page 89 for details.
decflt_rounding	This configuration parameter indicates the rounding mode for decimal floating-point operations. The default value is ROUND_HALF_EVEN. Determine if the default rounding mode is suitable for your decimal floating-point operations. Changing the value of this parameter can return different results.
enable_xmlchar	During database migration, enable_xmlchar is set to YES. This setting allows the use of host variables or parameter markers for CHAR, VARCHAR, CLOB, and LONG VARCHAR data types in the XMLPARSE function. This use could lead to substitution characters in the input or output host variables, and certain queries can return a result set different than expected if you do not use escape characters in your queries. The character substitution does not happen if you use host variables of XML type. If you want to block the use of the CHAR, VARCHAR, CLOB, and LONG VARCHAR data types in the XMLPARSE function, set this variable to NO.

Changes to existing database configuration parameters

The following table describes the migration impact of changes to existing database configuration parameters:

Table 12. Changes to existing database configuration parameters

Name	Migration impact
applheapsz	In migrated databases, applheapsz is set to AUTOMATIC to account for changes to the DB2 memory model. In previous releases, this parameter indicated the amount of memory for each database agent. Now, this parameter indicates the total amount of memory for an application. The new XSLTRANSFORM function transforms a given XML document into another document as defined by a given XSL stylesheet document. All documents involved in the transformation are stored in memory allocated from the application heap. If applheapsz is not set to AUTOMATIC, increase it to a minimum of 3000 for large XML documents.
database_memory	Now, you can set this parameter to AUTOMATIC on HP-UX, Solaris, and Linux operating systems. See database_memory for details about the AUTOMATIC setting.
dbheap and stat_heap_sz	In migrated databases, these parameters are set to AUTOMATIC to account for changes to the DB2 memory model.
maxfilop	This parameter now indicates the maximum number of file handles that can be open for a database. In previous releases, this parameter indicated the maximum number of file handles that could be open for each database agent. During database migration, the maxfilop parameter value is changed to the DB2 Version 9.5 default value due to this change in meaning. See maxfilop for a list of the new default values for each operating system.
sortheap	The OLAP functions now use sort heap memory instead of application heap to provide a higher limit for memory resources. To avoid tuning this parameter for the new OLAP function memory requirement, set it to AUTOMATIC as shown in the following example: db2 UPDATE DB CFG FOR <i>database-name</i> USING sortheap AUTOMATIC db2 UPDATE DB CFG FOR <i>database-name</i> USING self_tuning_mem ON
sheapthres_shr	The default value for this configuration parameter is 5000. If the setting for this parameter is less than 5000 and you plan to use the LOAD command on tables with XML type columns, increase the sheapthres_shr parameter value to a minimum of 5000.
stmheap	You can now set stmheap to AUTOMATIC. Although this parameter retains its value after migration, consider setting stmheap to AUTOMATIC to account for changes to the DB2 memory model. This setting might cause changes to access plans generated by the SQL compiler.

Deprecated and discontinued database configuration parameters

The following table describes the migration impact of deprecated and discontinued database configuration parameters:

Table 13. Deprecated and discontinued database configuration parameters

Name	Migration impact
app_group_mem_sz , groupheap_ratio , and app_ctl_heap_sz	These database configuration parameters are deprecated due to changes in the DB2 memory model. The GET DB CFG command output does not show these parameters, and using the UPDATE DB CFG command with these parameters has no effect.

Table 13. *Deprecated and discontinued database configuration parameters (continued)*

Name	Migration impact
logretain and userexit	These parameters are deprecated and have been replaced by the logarchmeth1 configuration parameter since DB2 UDB Version 8.2. If you set logretain to RECOVERY or userexit to ON, logarchmeth1 is automatically set to LOGRETAIN or USEREXIT. After migration, start using logarchmeth1 instead of logretain and userexit before they become discontinued.
estore_seg_sz and num_estore_segs	These parameters are discontinued because the extended storage capability is also discontinued.
numsegs	This parameter is deprecated. You can now specify multiple containers for SMS table spaces in the CREATE DATABASE command.
priv_mem_thresh	This parameter is discontinued. The use of this parameter is not required because the database manager now uses a multithreaded architecture.

Changes to physical design characteristics of databases

The following table describes the migration impact of changes in physical design characteristics of databases:

Table 14. *Changes to physical design characteristics of databases*

Command	Migration impact
CREATE TABLESPACE and CREATE DATABASE	Starting in DB2 Version 9.5 on certain platforms, the default for I/O operations is NO FILE SYSTEM CACHING when you create a table space without specifying whether I/O operations are cached at the file system level. The default was FILE SYSTEM CACHING prior to DB2 Version 9.5. Specify the FILE SYSTEM CACHING clause when you create a table space or a database if you still want to use caching for I/O operations. See “File system caching configurations” in <i>Data Servers, Databases, and Database Objects Guide</i> for details about platforms where FILE SYSTEM CACHING remains as the default option.

Changes to authorities and privileges

The following table describes the migration impact of changes in authorities and privileges:

Table 15. *Changes to authorities and privileges*

Name	Migration impact
SECADM authority	During database migration, the instance configuration settings are used to create an audit policy for each database. If the audit facility is enabled for the instance, the audit policies are associated with the databases to enable auditing. You must grant the SECADM privilege to the users that manage the audit facility. See “Setting up security to manage database auditing in migrated databases” on page 81 for details.
SYSADM authority	Although SYSADM authority can grant and revoke SECADM authority, it does not have any of the abilities of the SECADM authority.

Table 15. Changes to authorities and privileges (continued)

Name	Migration impact
USAGE privilege on a workload	<p>For migrated databases where you set the RESTRICT_ACCESS database configuration parameter to YES, grant the USAGE privilege to non-DBADM users on the SYSDEFAULTUSERWORKLOAD workload. Otherwise, non-DBADM users are unable to submit any work to the databases.</p> <pre>GRANT USAGE ON WORKLOAD SYSDEFAULTUSERWORKLOAD TO USER user-name</pre>

Deprecated or discontinued functionality that impacts DB2 server migration

You should be aware of functionality that is deprecated or discontinued in DB2 Version 9.5 that can impact the migration of your DB2 server. Also, you should be aware of the DB2 products that are no longer supported because migration from these products to DB2 Version 9.5 is unsupported.

To deal with these functionality changes, you must perform additional tasks before or after migration. The majority of these tasks are pre-migration or post-migration tasks for DB2 servers. The following list describes changes that are not included in the pre-migration and post-migration tasks for DB2 servers:

Address Windowing Extensions (AWE) buffer pools

Support for AWE buffer pools using the **DB2_AWE** registry variable is discontinued. If you want to allocate more memory than the virtual addressable memory limit on 32-bit Windows operating systems or you are using AWE buffer pools, consider migrating to a DB2 Version 9.5 64-bit database product.

db2undgpc command

The db2undgpc command is not supported in DB2 Version 9.5. During the database migration to DB2 UDB Version 8, the EXECUTE privilege was granted to PUBLIC for all existing functions, methods, and external stored procedures. The db2undgpc revokes the EXECUTE privilege on all these routines.

If you did not run this command after you migrated your databases to DB2 UDB Version 8 or DB2 Version 9.1, you should run it before you migrate your database to DB2 Version 9.5. Refer to “Revoking the EXECUTE privilege on migrated routines from PUBLIC” on page 41 for details.

Network Information Services (NIS and NIS+)

NIS and NIS+ support for user authentication is deprecated on Linux and UNIX operating systems. You can use Lightweight Directory Access Protocol (LDAP) instead. DB2 Version 9.5 supports LDAP-based authentication and group lookup functionality through the use of LDAP security plug-in modules. The LDAP security plug-in modules, samples, and documentation on the configuration and usage of plug-ins are available for download from https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?lang=en_US&source=swg-dm-db2ldap.

Raw logs

The use of raw devices for database logging has been deprecated since DB2 Version 9.1 and will be removed in a future release. You should use a file system instead of a raw device. Using a file system with non-buffered I/O capabilities enabled, such as Concurrent I/O (CIO) or Direct I/O (DIO), can give you performance comparable to that of using raw devices. The following example illustrates how to change the **newlogpath** parameter setting to a file system directory:

```
db2 UPDATE DATABASE CONFIGURATION USING newlogpath /disk2/newlogdir
```

The new setting does not become effective until the database is in a consistent state and all users are disconnected from the database. The database manager moves the logs to the new location after the first user connects to the database.

Migration from DB2 UDB Version 8

If you are migrating from DB2 UDB Version 8, review *Deprecated or discontinued functionality in DB2 Version 9.1* to learn about additional possible impacts on the migration of your DB2 server.

Migration from non-DB2 relational database management systems

Migrating from a non-DB2 relational database management system is a more complex process than migrating from a DB2 database product. Therefore, you should carefully determine what the migration process entails and create a porting plan.

The porting plan should include tasks such as, converting your database objects to create the equivalent database objects in a DB2 database, moving the actual data to the new DB2 database and porting your database applications. Porting your applications refers to converting SQL statements, modifying interface calls, and converting any database specific code to access DB2 databases.

The most common approaches to converting database application code are manual conversion, dynamic call translation, and automated conversion. In general, conversion tools take source code as input and translate data management calls to equivalent SQL calls. Information from the source and target database, as well as program code, is used to build the new SQL statements.

The IBM Migration Toolkit (MTK) is a conversion tool that is designed to migrate data and the query and procedure language from source database management systems such as Informix[®] Dynamic Server, Informix Extended Parallel Server (XPS), Microsoft SQL Server, Oracle, and Sybase Enterprise to DB2 database products. MTK runs on AIX, Linux, Solaris, and Windows operating systems. The only language supported is English. MTK is available as a complementary download from the IBM Migration Toolkit Web page.

The most important and frequently accessed resources that IBM offers to assist in all aspects of migration from a non-DB2 relational database management systems are as follows:

- The IBM DB2 porting Web site can help you to find the information that you need to port your application and its data from other database management systems. This Web site describes the common migration steps and provides resources including tools and education. Additional resources are provided for IBM customers and IBM Business Partners.

- The worldwide IBM Innovation Centers for Business Partners offer a wide range of complimentary workshops and technical seminars. Visit the training resources page to find out details and schedules for workshops such as:
 - Migration to DB2 - Toolkits, hints and tips workshop designed for users that want to learn about toolkits and other resources designed to help you port your applications to DB2 database products.
 - DB2 enablement and porting workshop designed for ISVs and IBM Business Partners who plan to sell or port applications to DB2 database products. You can bring your existing database application to the workshop.
- The IBM Virtual Innovation Center™ (VIC) is an online knowledge and enablement center that provides educational courses, live mentoring, online technical support, solution roadmaps, client simulations, answers to FAQs, case studies, and discussion forums.
- The DB2 Migrate Now! end-to-end offering for strategic IBM Business Partners that includes migration tool kits, complementary online education, information, sales teams and other resources to assist you in planning and implementing your migration to DB2 products from Oracle, Sybase, and Microsoft SQL server.
- The developerWorks® Information Management Web site offers technical resources for DB2 Information Management software. It features product information, downloads, learning resources, support, and communities. On this Web site you can find many articles and tutorials that can help you to learn about the features of DB2 database products and how to use them in your applications.

Chapter 5. Pre-migration tasks for DB2 servers

Before you migrate your DB2 server, review the migration essentials for DB2 servers, including recommendations, restrictions, and disk space requirements to identify the changes or restrictions that can affect your migration. You must be ready to address any issues prior to migration in order to have a successful migration.

Prepare for the migration of your DB2 servers by performing the following tasks:

1. If you use distributed transactions involving DB2 databases, ensure that the databases to be migrated do not contain any indoubt transactions by using the LIST INDOUBT TRANSACTIONS command to get a list of indoubt transactions and to interactively resolve any indoubt transactions.
2. Verify that databases are ready for DB2 migration to identify any problems before the actual migration. You must resolve them before you proceed with the migration.
3. Revoke the EXECUTE privilege from PUBLIC on functions and procedures to maintain secure database access.
4. Optional: Stop HADR on the primary and standby databases. You can migrate only the primary database.
5. Back up your databases to be able to migrate them to a new migrated system or restore them in the original pre-migration system.
6. Back up configuration and diagnostic information to have a record of your current configuration that you can compare with the configuration after the migration. You can also use this information to create new instances or databases using the same configuration that you had before migration.
7. Archive all of the DB2 log files, either for SQL replication if the log files are needed by the Capture program or for high availability disaster recovery (HADR) replication if the log files are needed to create a standby database.
8. Review the disk space requirements to ensure that you have enough free disk space, temporary table space and log space for the migration and increase table space and log file sizes if necessary. Depending on the number of database objects, you might require more log space to perform the migration.
9. Windows only: If you obtained customized code page conversion tables from the DB2 support service, you need to backup all of the files in the DB2OLD\conv directory where DB2OLD is the location of your existing DB2 Version 9.1 or DB2 UDB Version 8 copy. You do not need to backup standard code page conversion tables. Migrating your DB2 Version 9.1 or DB2 UDB Version 8 copy removes these tables because standard code page tables are contained in a DB2 Version 9.5 library.
10. Linux only: Change raw devices to block devices.
11. Optional: Migrate your DB2 server in a test environment to identify migration issues and to verify that applications, scripts, tools and routines work as expected before migrating your production environment.
12. In DB2 Version 9.5, all significant migration events are logged in the db2diag.log file when the **diaglevel** database manager configuration parameter is set to 3 (default value) or higher. If this parameter is set to 2 or less, set this parameter to 3 or higher before migration. See "Setting the diagnostic log file error capture level" in *Troubleshooting Guide*.
13. Take the DB2 server offline for migration.

Verifying that your databases are ready for migration

Before you migrate your databases, it is important to use the `db2ckmig` command to verify that your databases are ready for migration.

This command verifies that all the following conditions are true:

- A catalogued database actually exists.
- A database is not in an inconsistent state.
- A database is not in a backup pending state.
- A database is not in a restore pending state.
- A database is not in roll-forward pending state.
- Table spaces are in a normal state.
- A database does not contain user-defined types (UDTs) with the name `ARRAY`, `BINARY`, `DECFLOAT`, `VARBINARY` and `XML`.
- A database does not contain the system-defined `DATALINK` data type.
- A database does not have orphan rows in system catalog tables that would cause database migration to fail.
- A database enabled as an HADR primary database allows successful connections.
- An HADR database role is not standby.
- If `SYSCATSPACE` is a DMS table space and the automatic resizing functionality is not enabled, `SYSCATSPACE` has at least 50% free pages of total pages.

A database must pass all of these checks to succeed at the migration process.

The `db2imigr` calls the `db2ckmig` command. The `db2imigr` fails if the `db2ckmig` command finds any of the conditions listed above are not true, and returns the error code `DBI1205E` described in *Message Reference, Volume 2*.

Prerequisites

- Ensure that you have `SYSADM` authority.
- Ensure that all the local databases that you want to migrate are cataloged.
- On Linux or UNIX operating systems, you need to install a DB2 Version 9.5 copy to be able to run the `db2ckmig` command. Ensure that you meet the installation requirements for DB2 database products. See “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers*.

Restriction

In a partitioned database environment to verify that your databases are ready for migration, you must run the `db2ckmig` command on each database partition.

Procedure

To verify that your databases are ready for migration:

1. Log on to the DB2 server as the DB2 instance owner that you want to migrate.
2. Stop the instance by running the `db2stop` command.
3. From a DB2 command line prompt move to the appropriate directory:
 - On UNIX or Linux operating systems, change to the `$DB2DIR/bin` directory where `DB2DIR` is the location that you specified during the DB2 Version 9.5 installation.

- On Windows operating system, you need to insert the DB2 Version 9.5 product CD in the drive and change to the \db2\Windows\utilities directory.
4. Run the db2ckmig command to verify that the databases that are owned by the current instance are ready to be migrated and to generate a log file.

```
db2ckmig sample -l db2ckmig.log -u adminuser -p password
db2ckmig was successful. Database(s) can be migrated.
```

where *sample* is the database name and db2ckmig.log is the log file created in the current directory that includes details on errors and warnings.

Each time you issue this command, it overwrites the existing log file. You can rename the log file to avoid losing the error details. You must correct these errors before you migrate.

When the db2imigr command runs the db2ckmig command, the log file specified is the migration.log file in the instance home directory for Linux and UNIX or in the current directory for Windows.

5. If you have external unfenced routines on Linux and UNIX that have no dependency on the DB2 engine library in your database, the db2ckmig command returns the warning message SQL1349W and generates a file in the same directory as the log file called alter_unfenced_<database-name>.lst containing a list of all the external unfenced routines that will be redefined as FENCED and NOT THREADSAFE when you migrate the database. See “Migrating C, C++, and COBOL routines” on page 150 for details on how to safely run your routines in the new multithreaded database manager.
6. Ensure that the log file for db2ckmig command contains the following text: Version of DB2CKMIG being run: VERSION 9.5. This text confirms that you are running the correct level of the db2ckmig command.
7. Start the instance by running the db2start command.

Revoking the EXECUTE privilege on migrated routines from PUBLIC

During the database migration to DB2 UDB Version 8, the EXECUTE privilege was granted to PUBLIC for all existing functions, methods, and external stored procedures. If you want to revoke this privilege from PUBLIC for all these routines, run the db2undgp command to revoke the EXECUTE privilege on all these routines.

If you ran the db2undgp command after you migrated your databases to DB2 UDB Version 8 or DB2 Version 9.1, you do not have to run this command again after your databases are migrated to DB2 Version 9.5. However, if you did not run this command, you should run it before you migrate your database to DB2 Version 9.5. The db2undgp command is not supported in DB2 Version 9.5.

To revoke the EXECUTE privilege on migrated routines from PUBLIC:

1. Run the db2undgp. In the following example, the option -o creates a file that contains all the REVOKE statements needed to remove the EXECUTE privilege from PUBLIC:

```
db2undgp -d sample -o revoke.db2
```

You can review or edit this file to remove any specific statements when you want to keep the EXECUTE privilege granted to PUBLIC for any routine.

2. Grant the EXECUTE privilege to specific users on all your routines. The following statement shows how to grant this privilege on all functions under a specific schema:

```
db2 GRANT EXECUTE ON FUNCTION schema-name.* to USERID
```

3. Run all your routines as a user granted with the EXECUTE privilege to ensure they run successfully.

Backing up databases before migration

Before you start the migration process to DB2 Version 9.5, it is strongly recommended that you perform a full *offline* database backup. If an error occurs during the migration process, you need full database backups to recover and migrate your databases.

Once you migrate your instances to DB2 Version 9.5, you cannot backup databases until you migrate them.

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

To perform an offline full back up for each of your local databases:

1. Disconnect all applications and users from the database. To get a list of all database connections for the current instance, issue the LIST APPLICATIONS command. If all applications are disconnected, this command returns the following message:

```
db2 list applications
SQL1611W No data was returned by the Database System Monitor.
SQLSTATE=00000
```

To disconnect all applications and users, use the FORCE APPLICATION command:

```
db2 force application all
```

2. Backup your database using the BACKUP DATABASE command. The following is an example for UNIX operating systems:

```
db2 BACKUP DATABASE sample USER arada USING password TO backup-dir
```

where *sample* is the database alias, the username is *arada*, the password is *password*, and the directory to create back up files is *backup-dir*.

In partitioned database environments, . See "Backing up partitioned databases" in *Data Recovery and High Availability Guide and Reference*.

If you performed a full *offline* database backup recently and you cannot perform another one before migration, you can perform an incremental *offline* database backup instead. Refer to "Migrating to a new DB2 server" on page 66 for details on how to migrate your database using an incremental *offline* database backup.

3. Optional: Test the integrity of a backup image to ensure that the image can be restored using the db2ckbkp Check Backup command. The following is an example on UNIX operating systems:

```
cd backup-dir
db2ckbkp SAMPLE.0.arada.NODE0000.CATN0000.20051014114322.001
```

[1] Buffers processed: #####

Image Verification Complete - successful.

Backing up DB2 server configuration and diagnostic information

Backing up your settings for database and database manager configuration parameters before DB2 server migration, allows you to verify DB2 server behavior after migration, and to re-create instances and databases.

In addition, you can collect information from your DB2 servers about the database system catalogs, DB2 registry variables settings, explain table data, and diagnostic information that can help in problem determination if you encounter any post-migration differences in the database manager behavior or performance.

Prerequisite

You must have SYSADM authority in order to execute all of the following tasks, although some tasks require lesser authority privileges or none.

Procedure

To back up your DB2 server configuration and diagnostic information:

1. Run the `db2support` command, for all your databases that you are going to migrate in all your instances, to collect information from your DB2 servers. This command allows you to collect information on the database system catalog, database and database manager configuration parameters settings, DB2 registry variables settings, explain table data, and diagnostic information required by DB2 support in case of problems.

```
db2support output-directory -d database-name -cl 0
```

The `-cl 0` parameter collects the database system catalog, database and database manager configuration parameters settings, DB2 registry variables settings. The information collected is stored on a compressed zip file in output directory. A summary report in HTML format is included. You need to run this command for all your databases.

It is important that you keep this zip file after you complete the migration for several months. The information in the zip file can help in quickly resolving any performance issues with the new release.

2. Back up the information about all the packages for your applications associated with each database. Use the following command to list packages associated with your databases and redirect the command output to a file:

```
db2 LIST PACKAGES FOR SCHEMA schema-name  
SHOW DETAIL > /migration/sample_pckg.txt
```

The `FOR SCHEMA` clause allows you to list all packages for a specific schema, if your application has several schemas you need to repeat this command for each schema name or use `FOR ALL` clause.

3. If you enabled the audit facility, back up the audit configuration of your instances by issuing the following command:

```
db2audit describe > audit_instance-name.cfg
```

If you have multiple instances, repeat this command for each instance.

4. "Back up all your external routines" See "Backup and restore of external routine library and class files" in *Administrative Routines and Views*. The following example shows how to backup all external routines created using the default path in UNIX operating systems:

```
cp -R $INSTHOME/sql1lib/function $INSTHOME/routine_backup
```

Where INSTHOME is set to the home directory of the instance owner. If you have specified a full path that is not under the default routines path when you created your external routines in the database, you do not need to back up your routines but you must ensure the existing libraries remain on the current location.

- Optional: The db2support command HTML report includes the database manager configuration parameter settings for the instance that owns the specified database. You can use the GET DATABASE MANAGER CONFIGURATION command to back up your settings for database manager configuration parameters and redirect the command output to a file to save these settings for each instance:

```
db2 GET DBM CFG > dbm_instname.cfg
```

where *instname* is the instance name.

- Optional: The db2support command HTML report includes the database configuration parameter settings for the specified database. You can use the GET DATABASE CONFIGURATION command to *back up your settings for database configuration parameters* and redirect the command output to a file to save these settings for each database:

```
db2 GET DB CFG FOR database_alias  
SHOW DETAIL > db_database_alias.cfg
```

where *database_alias* is the database alias and the **SHOW DETAIL** clause displays the values calculated by the database manager when configuration parameters are set to AUTOMATIC.

Database configuration parameters can be the same on each database partition in a partitioned database environment. If they are not the same, back up the database configuration parameter settings for each database partition.

- Optional: The db2support command generates a file with the output of the db2look command for the specified database. However if you need additional information not present in the generated DDL file, you can use this command to save the DDL information for your databases and the statements to re-create your database objects:

```
db2look -d sample -e -o sample_tbs.db2 -l -x
```

- Optional: The db2support command HTML report includes the environment and registry variable settings for the instance that owns the specified database. You can use the db2set command to back up your DB2 profile registry variables settings and redirect the command output to a file to save these settings:

```
db2set -all > reg_instname.txt
```

If you set DB2 environment variables, use the appropriate system command to list environment variables and their values. For example, on AIX you can issue the following command:

```
set |grep DB2 > env_instname.txt
```

When possible, use the output from the set command and run the db2set command to set these environment variables as registry variables in the DB2 profile registry.

Increasing table space and log file sizes before migration

Before you start migrating your DB2 server, you must ensure that you have a sufficient amount of free space on your system catalog table space and temporary table space, and enough log space to migrate your databases.

Prerequisite

Ensure that you have SYSCTRL or SYSADM authority to be able to increase the size of table spaces and log space.

Restriction

Additional considerations are required in partitioned database environments to increase table space sizes, since table spaces span across database partitions. Also, you only need to increase the log space in the catalog database partition server.

Procedure

To increase the size of your table spaces and log space:

1. Connect to the database you want to migrate:
`db2 CONNECT TO sample`
2. Determine your table space disk usage using the following command:
`db2 LIST TABLESPACES SHOW DETAIL`

Collect the number of total pages, used pages, free pages and page size. Refer to the following table for a summary of the information obtained from the previous command:

Table 16. Table space information for sample database

Table space	Type	Total pages	Used pages	Free pages	Page size
SYSCATSPACE	SMS	8172	8172	N/A	4086
TEMPSPACE1	SMS	10	10	N/A	4086

3. Increase the size of the system catalog table spaces.
 - If you have an SMS table space, ensure that you have at least the same amount of used pages available as free disk space; in this example, about 32 MB.
 - If you have a DMS table space and the number of used pages is greater than the number of free pages, use the following formula to calculate the number of pages to increase per container:
$$\text{number_of_pages} = (\text{used_pages} - \text{free_pages}) / \text{number_of_containers_in_SYSCATSPACE}$$

Then use the following command to increase the size of all containers in the system catalog table space:

```
db2 "ALTER TABLESPACE SYSCATSPACE EXTEND (ALL number_of_pages)"
```

4. Increase the size of the temporary table spaces.
 - If you have an SMS table space you only need to ensure that you have at least twice the amount of total pages for the system catalog table space in free disk space; in this example, about 64 MB.
 - If you have a DMS table space, use the following formula to calculate the number of pages to increase per container:

$$\text{number_of_pages} = (\text{number_of_total_pages_in_SYSCATSPACE}) / \text{number_of_containers_in_TEMPSPACE1}$$

Use the following command to increase the size of all containers in the temporary table space:

```
db2 "ALTER TABLESPACE TEMPSPACE1 EXTEND (ALL number_of_pages)"
```

- If you have a DMS table space with AUTORESIZE enabled and MAXSIZE is set to NONE, ensure that you have at least twice the amount of total pages for the system catalog table space in free disk space. If MAXSIZE is set to an integer value, ensure that this value is at least twice the amount of total pages. The following query returns the current size (amount of total pages in bytes) and MAXSIZE of the table space TEMPSPACE1 in the SAMPLE database:

```
db2 "SELECT TBSP_CURRENT_SIZE, TBSP_MAX_SIZE
      FROM table(SNAP_GET_TBSP_PART('SAMPLE', -1)) T
      WHERE TBSP_NAME = 'TEMPSPACE1'"
```

If TBSP_MAX_SIZE is less than twice the value of TBSP_CURRENT_SIZE, then you need to increase MAXSIZE using the ALTER TABLESPACE statement:

```
db2 "ALTER TABLESPACE TEMPSPACE1
      MAXSIZE (<TBSP_CURRENT_SIZE*2/1024>) K"
```

The automatic resizing of table spaces is available since DB2 UDB Version 8 FixPak 9.

5. Determine the current log space size using the GET DATABASE CONFIGURATION command. The following example shows how to record the values for *logfilsiz*, *logprimary*, and *logsecond* database configuration parameters on Linux and UNIX operating systems:

```
db2 GET DB CFG FOR sample |grep '(LOG[FPS]') | tee logsize.txt
Log file size (4KB)                (LOGFILSIZ) = 1000
Number of primary log files         (LOGPRIMARY) = 3
Number of secondary log files      (LOGSECOND) = 2
```

6. Increase your log space size using the following commands:

```
db2 UPDATE DB CFG FOR sample using LOGSECOND
      (current_value of LOGPRIMARY + current_value of LOGSECOND) * 2
```

If you already have a large log space, you might not need to increase it.

7. Optional: Enable infinite active log instead of increasing the log space, by setting *logsecond* to -1 and enabling archive logging. Infinite active logging allows an active unit of work to span the primary logs and archive logs, effectively allowing a transaction to use an infinite number of log files. You should be aware that if the migration fails, the time to rollback the transactions will depend on how many archived logs need to be retrieved. The following command shows an example on how to enable archive logging to disk and infinite logging:

```
db2 UPDATE DB CFG FOR sample using LOGARCHMETH1 DISK:archive-dir
db2 UPDATE DB CFG FOR sample using LOGSECOND -1
```

where *archive-dir* is the directory to archive the log files.

Although these are dynamic parameters, all applications must disconnect from this database before the new values become effective.

Changing raw devices to block devices (Linux)

Changing raw (character) devices to block devices on Linux operating systems is required before you migrate to DB2 Version 9.5.

The previous raw I/O method that required binding the block device to a raw (character) device using the raw utility is deprecated since DB2 Version 9.1, and

will be removed in a future release of DB2 database product. This raw I/O method is also deprecated in the Linux operating system and will be removed in a future release of Linux.

The block device method uses Direct I/O to achieve an equivalent performance compared to using the raw (character) device method.

Prerequisite

Ensure the database is offline in order to relocate the containers or change the log file path.

Restriction

In a partitioned database environment, the `db2relocatedb` command must be run against every database partition that requires changes. A different configuration file must be supplied for each database partition, and must include the `NODENUM` value of the database partition being changed.

Procedure

1. Perform a full offline backup of your database.
2. Shut down your database. Also consider putting the database in quiesce mode using the `QUIESCE DATABASE` command as shown in the following example:

```
db2 CONNECT TO sample
db2 QUIESCE DATABASE DEFER FORCE CONNECTIONS
db2 DEACTIVATE DATABASE database-alias
```

3. Use the raw `-a` system command to see which raw bindings you defined. This information will help you determine the block device you should use to replace a raw device for each container on your table spaces.
4. Create a configuration file for the `db2relocatedb` command. Use the clauses `CONT_PATH` and `LOG_DIR` to specify the old value with the new value. For example, you can create the `moveraw.cfg` file with the following content:

```
DB_NAME=SAMPLE
DB_PATH=/databases/SAMPLE
INSTANCE=db2inst1
NODENUM=0
LOG_DIR=/dev/raw/lograw,/dev/sda5
CONT_PATH=/dev/raw/raw1,/dev/sda1
CONT_PATH=/dev/raw/raw2,/dev/sda2
```

5. Execute the `db2relocatedb` command to change the configuration of the database files:

```
db2relocatedb -f moveraw.cfg
```

6. Activate your database:

```
db2 ACTIVATE DATABASE database-alias
```

7. Test that your database is functioning as expected. Connect to the database and execute queries on tables created on the table spaces that you relocated.
8. If you put the database in quiesce mode, you can restore the access and activate the database using the `UNQUIESCE DATABASE` command:

```
db2 CONNECT TO sample
db2 UNQUIESCE DATABASE
```

If you are restoring from a DB2 Version 9.1 or DB2 UDB Version 8 backup in DB2 Version 9.5, you must do a redirected restore to indicate block devices instead of raw character devices for your containers and log path.

Migrating DB2 servers in a test environment

Migrating DB2 servers in a test environment before you migrate them in your production environment allows you to address any problems during the migration process more effectively and to evaluate the impact of changes introduced in DB2 Version 9.5.

You can also verify that applications, scripts, tools and maintenance procedures work properly before migrating your production environment. In addition, you can assess the disk requirements and the time that it takes to migrate the database, to solidify your migration plan.

Prerequisite

You must have root authority on Linux and UNIX operating systems or Local Administrator authority on Windows. You must also have SYSADM authority.

Procedure

To duplicate your production environment in a test environment, you need to do the following tasks:

1. Install DB2 Version 9.1 or DB2 UDB Version 8.
2. Re-create your instances as test instances.
3. Create your database duplicates in the testing instances. You can duplicate your databases without data to test only database migration or using a data subset to test all your application functionality. Database migration converts only system catalog objects. Therefore, the volume of data in the tables does not impact the disk requirements or the time that it takes to migrate the database.
4. Perform the pre-migration tasks that apply to your DB2 server.
5. Install DB2 Version 9.5.
6. Migrate your instances.
7. Migrate your databases. Keep a record of the time it takes to migrate each database and the size of the system catalog table space, system temporary table space, and log space. The following example shows how to do this on an AIX operating system:

```
time db2 MIGRATE DATABASE nsample | tee migration_time.log
db2 connect to nsample
db2 LIST TABLESPACES SHOW DETAIL | tee tbs_details.log
db2 GET DB CFG FOR nsample | grep '(LOG[FPS])' | tee log_size.log
```

Use this information in your migration plan.

8. If you found any issues migrating your test databases, find a resolution to these issues before migrating your production environment. Add the tasks to resolve these issues to your migration plan.
9. Perform post-migration tasks that apply to your DB2 server.
10. Verify the migration was successful.
11. Test your applications, scripts, tools and maintenance procedures by connecting to the test databases that you migrated to the DB2 Version 9.5 copy if your test databases are populated with data.

Creating database duplicates

Creating production database duplicates in a test environment allows you to test migrating your databases before you migrate them in your production environment.

To create a database duplicate for testing database migration:

1. Log on as the instance owner on the production database server and use the `db2look` command to generate DDL scripts with all the existing objects in your databases. The following command shows how to generate the `sample.ddl` script for the `SAMPLE` database:

```
db2look -d sample -a -e -m -l -x -f -o sample.ddl
```

Edit the generated DDL scripts and change:

- The database name in the `CONNECT` statements
- The path of the user table space containers or data and reduce the sizes to a minimum size since we are going to recreate a database with no data or just a data subset

You can use your own DDL scripts to create test databases in the test instance instead of generating DDL scripts.

2. Log on as the instance owner in the test database server and create your database duplicates. The following example shows how to create a database duplicate of the `SAMPLE` database using the `sample.ddl` script:

```
db2 CREATE DATABASE NSAMPLE
db2 -tvsvf sample.ddl
db2 UPDATE DBM CONFIGURATION USING diaglevel 4
```

All significant migration events are logged in the `db2diag.log` file when the **diaglevel** database manager configuration parameter is set to 3 (default value) or higher. A value of 4 captures additional information that can be helpful in problem determination.

3. Adjust the size of the system catalog table space, temporary table space, and log space in your test databases if required.
4. Export data subsets of your production databases and import these data subsets into your test databases. You only need a data subset if you are going to test your applications in your testing environment.
5. Verify that your database duplicates were created successfully by connecting to the them and issue a small query.

Taking a DB2 server offline before migration

Before you can continue with the migration process, you must take your DB2 server offline by stopping the DB2 license service, stopping all command line processor sessions, disconnecting applications and users, and stopping the database manager.

Prerequisites

- Your system must meet the installation requirements for DB2 Version 9.5 before starting the migration process.
- You must have `SYSADM` authority.

Procedure

To take your server offline:

1. Stop the DB2 license service:
2. Disconnect all applications and users. To get a list of all database connections for the current instance, issue the LIST APPLICATIONS command. If all applications are disconnected, this command returns the following message:

```
db2licd -end  
db2 list applications  
SQL1611W No data was returned by the Database System Monitor.  
SQLSTATE=00000
```

To disconnect all applications and users, use the FORCE APPLICATION command:

- ```
db2 force application all
```
3. Stop all command line processor sessions by entering the following command in each session that was running the command line processor.
  4. When all applications and users are disconnected, stop each database manager instance:

```
db2 terminate
db2stop
```

---

## Chapter 6. Migrating a DB2 server (Windows)

Migrating a DB2 Version 9.1 or DB2 UDB Version 8 server on Windows to DB2 Version 9.5 requires that you install a new DB2 Version 9.5 copy and then migrate your existing instances and databases to this new copy.

If you choose to automatically migrate your existing DB2 Version 9.1 or DB2 UDB Version 8 copy during the DB2 Version 9.5 installation, your instances and DB2 administration server (DAS) are migrated but you still need to migrate your databases after installation. If you choose to install a new DB2 Version 9.5 copy, you must manually migrate your instances, your DAS, and databases.

This migration task describes the steps for direct migration from DB2 Version 9.1 or DB2 UDB Version 8 to DB2 Version 9.5. Review migrating environments with specific characteristics and determine which task applies better to your environment.

### Prerequisites

- Ensure that you have Local Administrator authority.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

### Restrictions

- This procedure applies only to migration from DB2 32-bit servers when you install the DB2 Version 9.5 32-bit database product or from DB2 64-bit servers when you install the DB2 Version 9.5 64-bit database product. The instance bit size is determined by the operating system and the DB2 Version 9.5 database product that you install, see “Support changes for 32-bit and 64-bit DB2 servers” on page 25 for details.
- Additional migration restrictions apply. Review the complete list.

### Procedure

To migrate a DB2 server from DB2 Version 9.1 or DB2 UDB Version 8 to DB2 Version 9.5:

1. Log on to the DB2 server as a user with Local Administrator authority.
2. Install DB2 Version 9.5 by running the setup.exe command to launch the DB2 Setup wizard. See “Installing DB2 servers (Windows)” in *Quick Beginnings for DB2 Servers*. You have two options:
  - Select the **Work with Existing** option on the **Install a Product** panel. Then in the Work with an existing DB2 copy window, choose the DB2 copy name with action migrate. All your instances running on the selected DB2 copy and your DAS are automatically migrated to the DB2 Version 9.5 copy. The selected DB2 copy and add-on products are uninstalled.  
You will get a warning that recommends that you to run the db2ckmig command if you have local databases. If you completed the pre-migration tasks, ignore this warning and continue the migration. Otherwise, verify that your databases are ready for DB2 migration before continuing with the installation.
  - Select the option Install New on the Install a Product panel. This option creates a new copy of DB2 Version 9.5, and you must migrate your instances after installation.

3. If you want your applications to access the DB2 Version 9.5 copy through the default interface or if you migrated your existing DB2 UDB Version 8 copy, set the DB2 Version 9.5 copy as the DB2 default copy. See “Changing the default DB2 and default IBM database client interface after installation (Windows)” in *Quick Beginnings for DB2 Servers*. You must define a default copy if you migrated from DB2 UDB Version 8 because there is no default copy defined in your DB2 server.
4. Optional: When you chose to install a new copy, migrate the DAS if you want to keep your existing DAS configuration and use new functionality available in DB2 Version 9.5. If your DAS is running on DB2 UDB Version 8, you need to migrate it to use the Control Center to administer your DB2 Version 9.5 and Version 9.1 instances.
5. Migrate your databases.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level to its pre-migration value, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating instances

As part of the overall process of migrating your DB2 Version 9.1 or DB2 UDB Version 8 server to DB2 Version 9.5, you must migrate your instances. On Linux and UNIX, you must manually migrate them. On Windows, you must manually migrate them if you did not choose to automatically migrate your existing DB2 Version 9.1 or DB2 UDB Version 8 copy during the DB2 Version 9.5 installation.

### Prerequisites

- You must have root authority on Linux and UNIX operating systems or Local Administrator authority on Windows.
- Before running the `db2imigr` command, it is recommended:
  - Verify that databases are ready for DB2 migration.
  - On Linux and UNIX, ensure that there is 20 MB of free space in the `/tmp` directory. The instance migration trace file is written to `/tmp`.

### Restrictions

- Review the migration restrictions for instance migration.

### Procedure

To manually migrate your DB2 Version 9.1 or DB2 UDB Version 8 instances to DB2 Version 9.5 using the `db2imigr` command:

1. Determine if you can migrate your existing instances to a DB2 Version 9.5 copy that you installed by performing the following actions:
  - Determine the node type. The following examples shows how to use the `GET DBM CFG` command to find out the node type:

| Operating system | Examples                                                                                                                 |
|------------------|--------------------------------------------------------------------------------------------------------------------------|
| Linux and UNIX   | <code>db2 GET DBM CFG   grep 'Node type'</code><br>Node type = Partitioned database server with local and remote clients |
| Windows          | <code>db2 GET DBM CFG   find "Node type"</code><br>Node type = Partitioned database server with local and remote clients |

- Review Table 5 on page 19 to determine the instance type using the nodetype and whether instance migration is supported. In the previous example, the node type is “Partitioned database server with local and remote clients” therefore the instance type is “ese” and you can only migrate to a DB2 Version 9.5 copy of DB2 Enterprise Server Edition. On Linux and UNIX operating systems, you can migrate to a DB2 Version 9.5 copy of DB2 Workgroup Server Edition but your instance is recreated with type wse using default configuration values.

If you cannot migrate your instance to any DB2 Version 9.5 copy that you installed, you need to install a copy of the DB2 Version 9.5 database product that supports migration of your instance type before you can proceed with the next step.

2. Disconnect all users, stop back-end processes and stop your DB2 Version 9.1 or DB2 UDB Version 8 instances by running the following command:
 

```
db2stop force (disconnects all users and stops the instance)
db2 terminate (terminates back-end process)
```
3. Log on to the DB2 server as root authority on Linux and UNIX operating systems or Local Administrator authority on Windows:
4. Migrate your instances by running the db2imigr command from the target DB2 Version 9.5 copy location. The following table shows how to run the db2imigr command to migrate your instances:

| Operating system | Command syntax                                                               |
|------------------|------------------------------------------------------------------------------|
| Linux and UNIX   | <code>\$DB2DIR/instance/db2imigr [ -u fencedID ] InstName<sup>a</sup></code> |
| Windows          | <code>"%DB2PATH%" \bin\db2imigr InstName /u:user,password<sup>b</sup></code> |

**Note:**

- a. Where *DB2DIR* is set to the location you specified during DB2 Version 9.5 installation, *fencedID* is the user name under which the fenced user-defined functions (UDFs) and stored procedures will run, and *InstName* is the login name of the instance owner.
- b. Where *DB2PATH* is set to the location you specified during DB2 Version 9.5 installation, *user,password* are the user name and password under which the DB2 service will run, and *InstName* is the name of the instance.

The db2imigr command implicitly calls the db2ckmig command to verify that your local databases are ready for migration and logs any errors in the migration.log log file. On Linux and UNIX, the log file is created in the instance home directory. On Windows, the log file is created in the current directory where you are running the db2imigr command. The db2imigr does not run as long as the db2ckmig command reports errors. Check the log file if you encounter any errors.

5. Log on to the DB2 server as a user with sufficient authority to start your instance.
6. Restart your instance by running the db2start command:
 

```
db2start
```
7. Verify that your instance is running on to DB2 Version 9.5 by running the db2level command:
 

```
db2level
```

The Informational tokens should include a string like "DB2 v9.5.X.X" where X is a digit number.

---

## Migrating the DB2 Administration Server (DAS)

Migrating your DB2 Administration Server (DAS) is only necessary when you want to keep your existing DAS configuration and use new functionality available in DB2 Version 9.5. If your DAS is running on DB2 UDB Version 8, migrating your DAS is necessary to use the Control Center for administration of DB2 Version 9.5 and Version 9.1 instances, task management, and task scheduling.

Otherwise, you can drop your existing DAS and create a new DAS in DB2 Version 9.5. See “Creating a DB2 administration server (DAS)” in Quick Beginnings for DB2 Servers.

On Windows operating systems, if you chose to automatically migrate your DB2 Version 9.1 or DB2 UDB Version 8 copy and you have a DAS running under this copy, the DAS is also migrated along with your instances.

After installing DB2 Version 9.5, you can manually migrate the DAS by running the `dasmigr` command.

### Prerequisite

- Ensure that you have SYSADM authority, and root access on Linux and UNIX operating systems or Local Administrator authority on Windows operating systems.

### Restrictions

- You can have only one DAS per computer.

### Procedure

To migrate the DAS:

1. Log on to the DB2 server as root on Linux and UNIX operating systems or Local Administrator authority on Windows.
2. Migrate the DAS under DB2 Version 9.1 or DB2 UDB Version 8 by running the `dasmigr` command:

| Operating system | Command syntax                         |
|------------------|----------------------------------------|
| Linux and UNIX   | <code>\$DB2DIR/instance/dasmigr</code> |
| Windows          | <code>%DB2PATH%\bin\dasmigr</code>     |

Where `DB2DIR` and `DB2PATH` indicate the location that you specified during DB2 Version 9.5 installation.

If the DAS is running, the `dasmigr` command stops the DAS before migration and starts the DAS after migration.

3. If you created a tools catalog database on your DB2 Version 9.1 or DB2 UDB Version 8 system and want to use your existing scripts and schedules on the Version 9.5 DB2 Control Center, perform the following steps:
  - Migrate the instance that owns the tools catalog database.
  - Migrate the tools catalog database.
  - Log on to the DB2 server as a user with SYSADM authority and run the `db2tdbmgr toolsdb` migration tool. The tool stops the scheduler before migrating the tools catalog database and restarts it after migration. If you run this tool from a remote client, you must stop the scheduler before migration and restart it after migration.

- Verify that the DAS is configured to access the migrated tools catalog database by running the GET ADMIN CFG command to display the current configuration settings for the tools catalog database:

```
db2 GET ADMIN CFG
```

```
Admin Server Configuration
```

```
...
Tools Catalog Database (TOOLSCAT_DB) = toolscatdb
Tools Catalog Database Instance (TOOLSCAT_INST) = db2inst1
Tools Catalog Database Schema (TOOLSCAT_SCHEMA) = cc
Scheduler User ID =
```

Use the UPDATE ADMIN CFG command if you need to change any configuration settings for the tools catalog database.

You should migrate your tools catalog whether you decide to migrate your DAS or not.

4. If you do not migrate or do not have a tools catalog database, you can create one in a Version 9.5 instance to use the task scheduling capability. See “CREATE TOOLS CATALOG command ” in *Command Reference*.

You can now use the Control Center for remote administration of DB2 Version 9.5 instances, as well as DB2 Version 9.1 and DB2 UDB Version 8 instances.

## Migrating databases

After you migrated your instances to DB2 Version 9.5, you need to migrate each database under each instance.

### Prerequisites

- Ensure that you have SYSADM authority.
- Ensure that all the local databases that you want to migrate are cataloged.
- Ensure that you backed up your databases as indicated in the pre-migration tasks.
- You must have DB2 Version 9.5 installed and migrate the instance to DB2 Version 9.5.

### Restrictions

- Review the migration restrictions for database migration.

### Procedure

To migrate a DB2 database:

1. Log on to the DB2 server as the instance owner or a user with SYSADM authority.
2. Optional: Rename or delete the db2diag.log file so that a new file is created. Also, remove or move to another directory any existing dump files, trap files, and an alert log files in the directory indicated by the **diagpath** parameter. By doing this, the files only contain information about the migration process that helps you to isolate and understand any problem that might occur during database migration.
3. Migrate the database using the MIGRATE DATABASE command:
 

```
db2 MIGRATE DATABASE database-alias USER username USING password
```

where *database-alias* is the name or the alias of the database you want to migrate and the username and password to authenticate a user with SYSADM authority.

4. If the database migration fails and returns the error message SQL1704N with a reason code that describes the cause of the failure, find this SQL error code and determine the list of the possible solutions for each reason code. One of the most common causes of migration failure is that the log file space is not large enough, in which case the following error is returned:  
SQL1704N Database migration failed. Reason code "3".

You must increase log file size and execute the MIGRATE DATABASE command again. Once the database migration is complete reset the value of **logfilesiz**, **logprimary**, and **logsecond** database configuration parameters.

There are additional error codes that are returned by the MIGRATE DATABASE command for specific cases not supported by database migration. These cases are described in the migration restrictions.

5. If the database migration returns the warning message SQL1243W, you need to drop or rename the SYSTOOLS.DB2LOOK\_INFO table. Otherwise, the ALTER TABLE and COPY SCHEMA statements will fail to run. Check if the SYSTOOLS.DB2LOOK\_INFO table exists by running the following command:  

```
db2 "SELECT tabname, tabschema, definer FROM syscat.tables
 WHERE tabschema = 'SYSTOOLS' AND tabname = 'DB2LOOK_INFO'"
```

If you created this table, simply rename it by running the RENAME statement:

```
db2 RENAME SYSTOOLS.DB2LOOK_INFO TO new-table-name
```

If you did not create this table, simply remove it by running the DROP command:

```
db2 DROP TABLE SYSTOOLS.DB2LOOK_INFO
```

6. If you have external unfenced routines on Linux or UNIX that have no dependency on the DB2 engine libraries, the MIGRATE DATABASE command redefines your external routines as FENCED and NOT THREADSAFE and returns the warning message SQL1349W.

This command also generates a script called *alter\_unfenced\_database-name.db2* with all the SQL statements to redefine external unfenced routines, altered during the database migration, as NOT FENCED and THREADSAFE. This script is created in the directory specified by the **diagpath** database manager configuration parameter. If the **diagpath** parameter is not set, the script is created in the INSTHOME/sql/lib/db2dump directory where INSTHOME is the instance home directory. See “Migrating C, C++, and COBOL routines” on page 150 for details on how to safely run your routines in the new multithreaded database manager.

7. Compare your database configuration settings after migration with the configuration settings you had before you migrated your database. Verify the following settings and database information are the same:
  - database configuration parameter settings
  - table spaces information
  - packages information for your applications only

You do not need to check package information for system generated packages. The information about system generated packages can change after migration.

8. Verify your database migration is successful. Connect to the migrated databases and issue a small query:

```
db2 connect to sample
```

```
Database Connection Information
```

```
Database server = DB2/AIX64 9.5.0
SQL authorization ID = TESTDB2
Local database alias = SAMPLE
```

```
db2 "select * from syscat.dbauth"
```

Alternatively, if you have sample files installed, run the testdata.db2 script:

```
cd samplefile-dir-clp
db2 connect to sample
db2 -tvf testdata.db2
```

where *samplefile-dir-clp* is DB2DIR/samples/clp on Linux and UNIX and DB2DIR\samples\clp on Windows, DB2DIR represents the location specified during DB2 Version 9.5 installation, and sample is the database name.

After migrating a DB2 database, performing the recommended post-migration tasks ensures a successful database migration.



---

## Chapter 7. Migrating a DB2 server (Linux and UNIX)

Migrating a DB2 Version 9.1 or DB2 UDB Version 8 server to DB2 Version 9.5 on Linux and UNIX requires that you install a new DB2 Version 9.5 copy and then migrate your existing instances and databases to this new copy. You must manually migrate your instances, your DB2 administration server (DAS), and your databases after you install DB2 Version 9.5.

This migration task describes the steps for direct migration from DB2 Version 9.1 or DB2 UDB Version 8 to DB2 Version 9.5 regardless of the instance bit size. Review migrating environments with specific characteristics and determine which task applies better to your environment.

### Prerequisites

Before migrating the DB2 server:

- Ensure that you have root access.
- Ensure that you meet the installation requirements for DB2 database products. See “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers*. The requirements for Linux and UNIX operating systems have changed.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

### Restrictions

- On Linux and UNIX operating systems except for Linux on x86, your existing 32-bit or 64-bit instances are migrated to DB2 Version 9.5 64-bit instances. The operating system and DB2 Version 9.5 database product that you installed determines the instance bit size, see “Support changes for 32-bit and 64-bit DB2 servers” on page 25 for details.
- Additional migration restrictions apply. Review the complete list.

### Procedure

To migrate a DB2 server from DB2 UDB Version 8 or DB2 Version 9.1 to DB2 Version 9.5:

1. Log on to the DB2 server as root.
2. Install DB2 Version 9.5. See “Installing DB2 servers using the DB2 Setup wizard (Linux and UNIX)” in *Quick Beginnings for DB2 Servers*. Run the `db2setup` command and select Install New on the Install a Product panel to install a new copy of DB2 Version 9.5.
3. Migrate instances from the same installation path that you indicated during DB2 Version 9.5 installation. Any 32-bit instances are migrated to DB2 Version 9.5 64-bit instances except for Linux on x86.
4. Optional: Migrate your DAS if you want to keep your existing DAS configuration and use new functionality available in DB2 Version 9.5. If your DAS is running on DB2 UDB Version 8, you need to migrate it to use the Control Center to administer your DB2 Version 9.5 and Version 9.1 instances.
5. Migrate databases.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating instances

As part of the overall process of migrating your DB2 Version 9.1 or DB2 UDB Version 8 server to DB2 Version 9.5, you must migrate your instances. On Linux and UNIX, you must manually migrate them. On Windows, you must manually migrate them if you did not choose to automatically migrate your existing DB2 Version 9.1 or DB2 UDB Version 8 copy during the DB2 Version 9.5 installation.

### Prerequisites

- You must have root authority on Linux and UNIX operating systems or Local Administrator authority on Windows.
- Before running the `db2imigr` command, it is recommended:
  - Verify that databases are ready for DB2 migration.
  - On Linux and UNIX, ensure that there is 20 MB of free space in the `/tmp` directory. The instance migration trace file is written to `/tmp`.

### Restrictions

- Review the migration restrictions for instance migration.

### Procedure

To manually migrate your DB2 Version 9.1 or DB2 UDB Version 8 instances to DB2 Version 9.5 using the `db2imigr` command:

1. Determine if you can migrate your existing instances to a DB2 Version 9.5 copy that you installed by performing the following actions:
  - Determine the node type. The following examples shows how to use the `GET DBM CFG` command to find out the node type:

| Operating system | Examples                                                                                                                 |
|------------------|--------------------------------------------------------------------------------------------------------------------------|
| Linux and UNIX   | <code>db2 GET DBM CFG   grep 'Node type'</code><br>Node type = Partitioned database server with local and remote clients |
| Windows          | <code>db2 GET DBM CFG   find "Node type"</code><br>Node type = Partitioned database server with local and remote clients |

- Review Table 5 on page 19 to determine the instance type using the `nodetype` and whether instance migration is supported. In the previous example, the node type is “Partitioned database server with local and remote clients” therefore the instance type is “ese” and you can only migrate to a DB2 Version 9.5 copy of DB2 Enterprise Server Edition. On Linux and UNIX operating systems, you can migrate to a DB2 Version 9.5 copy of DB2 Workgroup Server Edition but your instance is recreated with type `wse` using default configuration values.

If you cannot migrate your instance to any DB2 Version 9.5 copy that you installed, you need to install a copy of the DB2 Version 9.5 database product that supports migration of your instance type before you can proceed with the next step.

2. Disconnect all users, stop back-end processes and stop your DB2 Version 9.1 or DB2 UDB Version 8 instances by running the following command:
  - `db2stop force` (disconnects all users and stops the instance)
  - `db2 terminate` (terminates back-end process)
3. Log on to the DB2 server as root authority on Linux and UNIX operating systems or Local Administrator authority on Windows:

- Migrate your instances by running the `db2imigr` command from the target DB2 Version 9.5 copy location. The following table shows how to run the `db2imigr` command to migrate your instances:

| Operating system | Command syntax                                                                             |
|------------------|--------------------------------------------------------------------------------------------|
| Linux and UNIX   | <code>\$DB2DIR/instance/db2imigr [ -u <i>fencedID</i> ] <i>InstName</i><sup>a</sup></code> |
| Windows          | <code>"%DB2PATH%"\bin\db2imigr <i>InstName</i> /u:<i>user,password</i><sup>b</sup></code>  |

**Note:**

- Where *DB2DIR* is set to the location you specified during DB2 Version 9.5 installation, *fencedID* is the user name under which the fenced user-defined functions (UDFs) and stored procedures will run, and *InstName* is the login name of the instance owner.
- Where *DB2PATH* is set to the location you specified during DB2 Version 9.5 installation, *user,password* are the user name and password under which the DB2 service will run, and *InstName* is the name of the instance.

The `db2imigr` command implicitly calls the `db2ckmig` command to verify that your local databases are ready for migration and logs any errors in the `migration.log` log file. On Linux and UNIX, the log file is created in the instance home directory. On Windows, the log file is created in the current directory where you are running the `db2imigr` command. The `db2imigr` does not run as long as the `db2ckmig` command reports errors. Check the log file if you encounter any errors.

- Log on to the DB2 server as a user with sufficient authority to start your instance.
- Restart your instance by running the `db2start` command:  
`db2start`
- Verify that your instance is running on to DB2 Version 9.5 by running the `db2level` command:  
`db2level`

The Informational tokens should include a string like "DB2 v9.5.X.X" where X is a digit number.

---

## Migrating the DB2 Administration Server (DAS)

Migrating your DB2 Administration Server (DAS) is only necessary when you want to keep your existing DAS configuration and use new functionality available in DB2 Version 9.5. If your DAS is running on DB2 UDB Version 8, migrating your DAS is necessary to use the Control Center for administration of DB2 Version 9.5 and Version 9.1 instances, task management, and task scheduling.

Otherwise, you can drop your existing DAS and create a new DAS in DB2 Version 9.5. See "Creating a DB2 administration server (DAS)" in Quick Beginnings for DB2 Servers.

On Windows operating systems, if you chose to automatically migrate your DB2 Version 9.1 or DB2 UDB Version 8 copy and you have a DAS running under this copy, the DAS is also migrated along with your instances.

After installing DB2 Version 9.5, you can manually migrate the DAS by running the `dasmigr` command.

## Prerequisite

- Ensure that you have SYSADM authority, and root access on Linux and UNIX operating systems or Local Administrator authority on Windows operating systems.

## Restrictions

- You can have only one DAS per computer.

## Procedure

To migrate the DAS:

1. Log on to the DB2 server as root on Linux and UNIX operating systems or Local Administrator authority on Windows.
2. Migrate the DAS under DB2 Version 9.1 or DB2 UDB Version 8 by running the `dasmigr` command:

| Operating system | Command syntax                         |
|------------------|----------------------------------------|
| Linux and UNIX   | <code>\$DB2DIR/instance/dasmigr</code> |
| Windows          | <code>%DB2PATH%\bin\dasmigr</code>     |

Where `DB2DIR` and `DB2PATH` indicate the location that you specified during DB2 Version 9.5 installation.

If the DAS is running, the `dasmigr` command stops the DAS before migration and starts the DAS after migration.

3. If you created a tools catalog database on your DB2 Version 9.1 or DB2 UDB Version 8 system and want to use your existing scripts and schedules on the Version 9.5 DB2 Control Center, perform the following steps:
  - Migrate the instance that owns the tools catalog database.
  - Migrate the tools catalog database.
  - Log on to the DB2 server as a user with SYSADM authority and run the `db2tdbmgr toolsdb` migration tool. The tool stops the scheduler before migrating the tools catalog database and restarts it after migration. If you run this tool from a remote client, you must stop the scheduler before migration and restart it after migration.
  - Verify that the DAS is configured to access the migrated tools catalog database by running the `GET ADMIN CFG` command to display the current configuration settings for the tools catalog database:

```
db2 GET ADMIN CFG
```

```
Admin Server Configuration
```

```
...
Tools Catalog Database (TOOLSCAT_DB) = toolsdb
Tools Catalog Database Instance (TOOLSCAT_INST) = db2inst1
Tools Catalog Database Schema (TOOLSCAT_SCHEMA) = cc
Scheduler User ID =
```

Use the `UPDATE ADMIN CFG` command if you need to change any configuration settings for the tools catalog database.

You should migrate your tools catalog whether you decide to migrate your DAS or not.

4. If you do not migrate or do not have a tools catalog database, you can create one in a Version 9.5 instance to use the task scheduling capability. See “`CREATE TOOLS CATALOG` command ” in *Command Reference*.

You can now use the Control Center for remote administration of DB2 Version 9.5 instances, as well as DB2 Version 9.1 and DB2 UDB Version 8 instances.

---

## Migrating databases

After you migrated your instances to DB2 Version 9.5, you need to migrate each database under each instance.

### Prerequisites

- Ensure that you have SYSADM authority.
- Ensure that all the local databases that you want to migrate are cataloged.
- Ensure that you backed up your databases as indicated in the pre-migration tasks.
- You must have DB2 Version 9.5 installed and migrate the instance to DB2 Version 9.5.

### Restrictions

- Review the migration restrictions for database migration.

### Procedure

To migrate a DB2 database:

1. Log on to the DB2 server as the instance owner or a user with SYSADM authority.
2. Optional: Rename or delete the `db2diag.log` file so that a new file is created. Also, remove or move to another directory any existing dump files, trap files, and an alert log files in the directory indicated by the `diagpath` parameter. By doing this, the files only contain information about the migration process that helps you to isolate and understand any problem that might occur during database migration.
3. Migrate the database using the `MIGRATE DATABASE` command:

```
db2 MIGRATE DATABASE database-alias USER username USING password
```

where *database-alias* is the name or the alias of the database you want to migrate and the username and password to authenticate a user with SYSADM authority.

4. If the database migration fails and returns the error message SQL1704N with a reason code that describes the cause of the failure, find this SQL error code and determine the list of the possible solutions for each reason code. One of the most common causes of migration failure is that the log file space is not large enough, in which case the following error is returned:

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

You must increase log file size and execute the `MIGRATE DATABASE` command again. Once the database migration is complete reset the value of `logfilsiz`, `logprimary`, and `logsecond` database configuration parameters.

There are additional error codes that are returned by the `MIGRATE DATABASE` command for specific cases not supported by database migration. These cases are described in the migration restrictions.

5. If the database migration returns the warning message SQL1243W, you need to drop or rename the `SYSTOOLS.DB2LOOK_INFO` table. Otherwise, the `ALTER TABLE` and `COPY SCHEMA` statements will fail to run. Check if the `SYSTOOLS.DB2LOOK_INFO` table exists by running the following command:

```
db2 "SELECT tabname, tabschema, definer FROM syscat.tables
 WHERE tabschema = 'SYSTOOLS' AND tabname = 'DB2LOOK_INFO'"
```

If you created this table, simply rename it by running the RENAME statement:

```
db2 RENAME SYSTOOLS.DB2LOOK_INFO TO new-table-name
```

If you did not create this table, simply remove it by running the DROP command:

```
db2 DROP TABLE SYSTOOLS.DB2LOOK_INFO
```

6. If you have external unfenced routines on Linux or UNIX that have no dependency on the DB2 engine libraries, the MIGRATE DATABASE command redefines your external routines as FENCED and NOT THREADSAFE and returns the warning message SQL1349W.

This command also generates a script called `alter_unfenced_database-name.db2` with all the SQL statements to redefine external unfenced routines, altered during the database migration, as NOT FENCED and THREADSAFE. This script is created in the directory specified by the **diagpath** database manager configuration parameter. If the **diagpath** parameter is not set, the script is created in the `INSTHOME/sqllib/db2dump` directory where `INSTHOME` is the instance home directory. See “Migrating C, C++, and COBOL routines” on page 150 for details on how to safely run your routines in the new multithreaded database manager.

7. Compare your database configuration settings after migration with the configuration settings you had before you migrated your database. Verify the following settings and database information are the same:

- database configuration parameter settings
- table spaces information
- packages information for your applications only

You do not need to check package information for system generated packages. The information about system generated packages can change after migration.

8. Verify your database migration is successful. Connect to the migrated databases and issue a small query:

```
db2 connect to sample
```

```
Database Connection Information
```

```
Database server = DB2/AIX64 9.5.0
SQL authorization ID = TESTDB2
Local database alias = SAMPLE
```

```
db2 "select * from syscat.dbauth"
```

Alternatively, if you have sample files installed, run the `testdata.db2` script:

```
cd samplefile-dir-clp
db2 connect to sample
db2 -tvf testdata.db2
```

where `samplefile-dir-clp` is `DB2DIR/samples/clp` on Linux and UNIX and `DB2DIR\samples\clp` on Windows, `DB2DIR` represents the location specified during DB2 Version 9.5 installation, and `sample` is the database name.

After migrating a DB2 database, performing the recommended post-migration tasks ensures a successful database migration.

---

## Chapter 8. Migrating environments with specific characteristics

There are many factors that can impact the overall migration process, and the complexity of your environment is one of these factors. If you installed multiple DB2 product components, if you are migrating from a 32-bit Windows operating system to a 64-bit Windows operating system, or if you are migrating from a partitioned database environment, you must perform migration tasks that include steps specific to that environment instead of the basic DB2 server migration task.

Determine which of the following migration tasks applies to your environment, and perform this migration task:

- “Migrating DB2 32-bit servers to 64-bit systems (Windows)”
- “Migrating to a new DB2 server” on page 66
- “Migrating from a DB2 server with multiple DB2 copies” on page 70
- “Migrating partitioned database environments” on page 69
- “Migrating DB2 servers in Microsoft Cluster Server environments” on page 72
- “Migrating DB2 Data Links Manager environments” on page 73
- Migrating from XML Extender to native XML data store
- “Migrating DB2 Connect servers” in *Quick Beginnings for DB2 Connect Servers*
- “Migrating DB2 Spatial Extender” in *Spatial Extender and Geodetic Data Management Feature User’s Guide and Reference*
- “Migrating DB2 Net Search Extender” in *Net Search Extender Administration and User’s Guide*
- “Migrating Query Patroller” in *Query Patroller Administration and User’s Guide*

---

### Migrating DB2 32-bit servers to 64-bit systems (Windows)

On the Windows operating systems, there are two ways to migrate your DB2 UDB Version 8 or DB2 Version 9.1 32-bit server to a DB2 Version 9.5 64-bit server. One way is to migrate your existing DB2 32-bit server to DB2 Version 9.5 32-bit server, and then upgrade to DB2 Version 9.5 64-bit server.

The other way is to migrate to a new computer where DB2 Version 9.5 64-bit database product is installed.

#### Prerequisites

- Ensure that you have Local Administrator authority.
- Ensure that the DB2 server is running 64-bit windows operating system.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

#### Restrictions

- This procedure is covered by this task and only applies to Windows on X64.
- Additional migration restrictions apply. Review the complete list.

#### Procedure

To migrate from a DB2 UDB Version 9.1 or DB2 UDB Version 8 32-bit server to a DB2 Version 9.5 64-bit server:

1. Log on to the DB2 server as a user with Local Administrator authority.
2. If you have multiples copies of DB2 UDB Version 8 32-bit server or multiples copies of DB2 Version 9.1 32-bit server, perform the following actions:
  - Update all your Version 8 instances to run under one DB2 Version 8 32-bit server copy.
  - Update all your Version 9.1 instances to run under one DB2 Version 9.1 32-bit server copy.
  - If you have Version 8 and Version 9.1 instances, migrate your Version 8 instances to the DB2 Version 9.1 32-bit server copy.
  - Uninstall all the remaining DB2 server copies except the DB2 server copy where all instances are running. You should have only one DB2 UDB Version 8 32-bit server or DB2 Version 9.1 32-bit server copy.
3. Install DB2 Version 9.5 32-bit database product and select Work with Existing on the Install a Product panel. See “Installing DB2 servers (Windows) ” in *Quick Beginnings for DB2 Servers*. Then in the Work with an existing DB2 copy window, choose the DB2 copy name with action migrate. The selected DB2 copy is removed, and all your instances running on the selected DB2 copy and your DB2 Administration Server (DAS) are automatically migrated. Do not install additional copies of 32-bit DB2 Version 9.5.

You will get a warning that recommends that you to run the db2ckmig command if you have local databases. Ignore this warning and continue the migration if you completed the pre-migration tasks. Otherwise, verify that your databases are ready for DB2 migration before you continue with the installation.
4. Install DB2 Version 9.5 64-bit database product and select Work with Existing on the Install a Product panel. See “Installing DB2 servers (Windows) ” in *Quick Beginnings for DB2 Servers* . Then in the Work with an existing DB2 copy window, choose the DB2 copy name with action upgrade. This procedure removes DB2 Version 9.5 32-bit database product, and upgrades your existing 32-bit instances to 64-bit instances.
5. If you want your applications to access DB2 Version 9.5 copy through the default interface or if you migrated your existing DB2 UDB Version 8 copy, set the DB2 Version 9.5 copy as the DB2 default copy. See “Changing the default DB2 and default IBM database client interface copy after installation (Windows)” in *Quick Beginnings for DB2 Servers* .
6. Migrate your databases.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating to a new DB2 server

If you want to migrate to a new DB2 Version 9.5 server, you need to recreate your instances and then restore your DB2 Version 9.1 or DB2 UDB Version 8 databases from a database backup. After restoring the database backup, the RESTORE DATABASE command automatically runs the MIGRATE DATABASE command.

### Prerequisites

- Ensure that you have root access on Linux and UNIX operating systems or Local Administrator authority on Windows.

- Ensure that you have SYSADM authority.
- Ensure that you meet the “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers* . The requirements for operating systems have changed.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

### Restrictions

- Review the migration restrictions for DB2 servers.

### Procedure

To migrate to a new DB2 Version 9.5 server:

1. Perform a full offline database backup of your DB2 Version 9.1 or DB2 UDB Version 8 databases if you did not perform such backups as indicated in the pre-migration tasks. If you performed full *offline* database backups recently and you cannot perform another one before migration, you can perform an incremental *offline* database backup instead.
2. Log on to the new DB2 server as root on Linux and UNIX operating systems or user with Local Administrator authority on Windows operating systems.
3. Install DB2 Version 9.5 on the new DB2 server.
4. Re-create your instances by running the db2icrt command from the DB2 Version 9.5 copy location that you installed in the previous step. See “Creating an instance using db2icrt” in *Quick Beginnings for DB2 Servers*. If the new DB2 server has similar resources, then restore the database manager configuration parameter values for each instance using the UPDATE DBM CFG command and the values that you saved in the pre-migration tasks.  
In a partitioned database environment, you should configure the database manager configuration parameter values for all instances on all database partition servers.
5. Optional: Create a new DB2 Administration Server (DAS) on DB2 Version 9.5. You need a DAS if you want to keep your existing DAS configuration and use new functionality available in DB2 Version 9.5. If your DAS is running on DB2 UDB Version 8, you need to migrate it to use the Control Center to administer your DB2 Version 9.5 and Version 9.1 instances.
6. Transfer DB2 Version 9.1 or DB2 UDB Version 8 backup files for all the databases that you want to migrate to the new DB2 server.
7. Log on to the DB2 server as a user with SYSADM authority.
8. Migrate the database using the RESTORE DATABASE command. The following example shows how to restore the sample database on UNIX operating systems:

```
db2 RESTORE DATABASE sample FROM /db2/backups
```

where *sample* is the database name and /db2/backups is the directory for the database backup file.

If you performed an incremental *offline* database backup before migration, you must have access to the most recent full *offline* database backup and the incremental *offline* database backup and use an automatic incremental restore to migrate the database. See “Using incremental restore in a test and production environment” in *Data Recovery and High Availability Guide and Reference*. A manual incremental restore will fail because each RESTORE

DATABASE command tries to migrate the database before the database is completely recovered. The following example shows how to perform an automatic incremental restore:

```
db2 RESTORE DATABASE sample INCREMENTAL AUTOMATIC
 TAKEN AT timestamp WITHOUT PROMPTING
```

In a partitioned database environment, you must execute the RESTORE DATABASE command in all database partitions starting with the catalog partition first.

9. When the database was restored but the database was not migrated, the RESTORE DATABASE command returns the following error and includes the migration error message with the reason code:

```
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
```

The error message SQL1704N indicates the database migration failed. Find this SQL error code in the *Message Reference, Volume 2* to read the list of the possible solutions for each reason code. In the previous example, tokens "3" means reason code 3 which indicates that the migration failed because the database logs are full. If this error occurs, complete the following steps to migrate the database:

- a. Increase the size of the log files.
- b. Migrate the database using the MIGRATE DATABASE command.
- c. If the log file size is still not large enough, the following error is returned:  
SQL1704N Database migration failed. Reason code "3".

You must increase the log file size and attempt to migrate the database again.

- d. Once migration is completed reset the size of the log files.
10. Optional: Configure your new DB2 server to use the new resources available by running the AUTOCONFIGURE command to calculate the buffer pool sizes, and the database manager and database configuration parameters values. The following example shows how to run this command to only display recommended values for the sample database:

```
db2 CONNECT TO sample
db2 AUTOCONFIGURE USING MEM_PERCENT 80
 WORKLOAD_TYPE complex
 NUM_STMTS 1 TPM 73
 ADMIN_PRIORITY performance
 IS_POPULATED YES
 NUM_REMOTE_APPS 15
 ISOLATION CS
APPLY NONE;
```

If you choose not to run this command or not to apply the recommended values, manually configure your DB2 server to use the new resources. Otherwise, your databases might not perform as expected.

11. Verify your database migration is successful. Connect to the migrated databases and issue a small query:

```
db2 CONNECT TO sample

Database Connection Information

Database server = DB2/AIX64 9.5.0
```

```
SQL authorization ID = TESTDB2
Local database alias = SAMPLE
```

```
db2 "SELECT * FROM SYSCAT.DBAUTH"
```

Alternatively, if you have sample files installed, run the testdata.db2 script:

```
cd samplefile-dir-clp
db2 connect to sample
db2 -tvf testdata.db2
```

where *samplefile-dir-clp* is DB2DIR/samples/clp on Linux and UNIX and DB2DIR\samples\clp on Windows, DB2DIR represents the location specified during DB2 Version 9.5 installation, and sample is the database name.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating partitioned database environments

Migrating partitioned database environments requires that you install DB2 Version 9.5 in all database partition servers, migrate the instances and then migrate the databases.

You can migrate database partition servers from the catalog database partition server or any other database partition server. Should the migration process fail, you can retry migration from the catalog database partition server or any other database partition server again.

### Prerequisites

- Ensure that you have root access on Linux and UNIX operating systems or Local Administrator on Windows.
- Ensure that you have SYSADM authority.
- Review the "Installation requirements for DB2 database products" in *Quick Beginnings for DB2 Servers*. The prerequisites for operating systems have changed.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

### Restrictions

- The catalog database partition server must be up and running.
- Additional migration restrictions apply. Review the complete list.

### Procedure

To migrate DB2 servers in a partitioned database environment:

1. Perform a full offline backup for all databases. Verify that your databases are ready for migration, and any other pre-migration tasks that apply.
2. Install DB2 Version 9.5 on each participant database partition server and setup your partitioned database environment. See "Setting up a partitioned database environment" in *Quick Beginnings for DB2 Servers*.
3. Migrate each instance on the database partition server that owns the instance. The first entry in the db2nodes.cfg file of the instance is the database partition server instance owner. You can skip this step if you selected to automatically migrate the instances during DB2 Version 9.5 installation on Windows.

4. Migrate each database by running the `MIGRATE DATABASE` command on the catalog partition. If any database partitions are not available, these database partitions are not migrated. Also, if the `MIGRATE DATABASE` command is terminated, the remaining database partitions are not migrated. However, you can run the `MIGRATE DATABASE` command again to process these particular database partitions afterwards when they are available.

The catalog partition must be available when you issue the `MIGRATE DATABASE` regardless on what database partition you issue this command from.

5. Create a new DB2 Administration Server (DAS) on each database partition server. If you need to keep your existing DAS settings, you can migrate the DAS on each participating database partition server instead of creating a new DAS.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating from a DB2 server with multiple DB2 copies

Migrating from a DB2 server with multiple DB2 copies, requires that you install DB2 Version 9.5 as a new copy and then manually migrate the instances and databases after installation.

You can have a DB2 server with multiple copies of DB2 database products Version 9.1 installed. On Linux and UNIX, you could also have multiples copies of DB2 Enterprise Server Edition (ESE) Version 8 on the same DB2 server if have installed several alternate fix packs as a completely new copy of DB2 ESE Version 8.

You can manually migrate a DB2 Version 9.1 or DB2 UDB Version 8 instance at any fix pack level by executing the `db2imigr` command from the target DB2 Version 9.5 copy of your choice. Once an instance is migrated to a DB2 Version 9.5 copy, you cannot migrate to another DB2 Version 9.5 copy. Also you cannot migrate to DB2 Version 9.1 or DB2 UDB Version 8. However, you can update an instance between different DB2 copies of DB2 Version 9.5 using the `db2iupdt` command.

### Prerequisites

- Ensure that you have root access on Linux and UNIX operating systems or Local Administrator on Windows.
- Ensure that you meet the installation requirements for DB2 database products. The requirements for operating systems have changed.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

### Restrictions

- This procedure does not apply to migration from DB2 32-bit servers to 64-bit systems on Windows. Refer to “Migrating DB2 32-bit servers to 64-bit systems (Windows)” on page 65 for details.
- Review the migration restrictions for DB2 servers.

### Procedure

To migrate a DB2 server with multiple DB2 copies:

1. Log on to the DB2 server as root or a user with Local Administrator authority.

2. Install DB2 Version 9.5 as a new copy of DB2 Version 9.5 by running the DB2 Setup wizard and select Install New on the Install a Product panel:
  - Installing DB2 servers (Windows) in *Quick Beginnings for DB2 Servers*
  - Installing DB2 servers (Linux and UNIX) in *Quick Beginnings for DB2 Servers*
 You can install multiple DB2 Version 9.5 copies, if you want to migrate your DB2 Version 9.1 or DB2 UDB Version 8 instances at different levels to different DB2 Version 9.5 copies.
3. Migrate instances using the db2imigr command from the installation path of the DB2 Version 9.5 copy of your choice. For example, assume that you have the following DB2 copies and instances on an AIX server and a Windows server:

Table 17. Directory examples for DB2 copies.

| Instance name        | OS      | DB2 copy directory                         |
|----------------------|---------|--------------------------------------------|
| db2inst1             | AIX     | /usr/opt/db2_08_FP7/                       |
| db2inst2             | AIX     | /opt/IBM/db2/V9.1                          |
| db2inst3             | AIX     | /home/db2/myV9.1                           |
| No instances created | AIX     | /opt/IBM/db2/V9.5<br>/home/db2/myV9.5      |
| DB2                  | Windows | C:\Program Files\IBM\SQLLIB\ (Version 8.2) |
| DB2_91               | Windows | C:\Program Files\IBM\SQLLIB_91\            |
| No instances created | Windows | C:\Program Files\IBM\SQLLIB_95\            |

You can then run the following commands to successfully migrate your instances to DB2 Version 9.5:

Table 18. Instance migration command examples.

| Migrate Instance | Commands                                                                        |
|------------------|---------------------------------------------------------------------------------|
| db2inst1         | cd /opt/IBM/db2/V9.5/instance<br>./db2imigr -u db2fenc1 db2inst1                |
| db2inst2         | cd /opt/IBM/db2/V9.5/instance<br>./db2imigr -u db2fenc2 db2inst2                |
| db2inst3         | cd /home/db2/myV9.5/instance<br>./db2imigr -u db2fenc3 db2inst3                 |
| DB2              | cd C:\Program Files\IBM\SQLLIB_95\BIN<br>db2imigr DB2 /u:db2admin1,password1    |
| DB2_91           | cd C:\Program Files\IBM\SQLLIB_95\BIN<br>db2imigr DB2_91 /u:db2admin2,password2 |

4. Optional: Migrate the DB2 Administration Server if you want to keep your existing configuration and to administer your DB2 Version 9.5 instances using the Control Center.
5. Log on to the DB2 server as a user with SYSADM authority.
6. Migrate databases.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating DB2 servers in Microsoft Cluster Server environments

Migrating DB2 servers in Microsoft Cluster Server (MSCS) environments to DB2 Version 9.5 requires that you install DB2 Version 9.5 as a new copy in all nodes and then migrate your MSCS instances and databases.

Microsoft Cluster Server (MSCS) provides High Availability functions to windows users. During setup of DB2 server failover support on MSCS, a server instance is transformed into an MSCS instance. You can run the `db2imigr` command to migrate your MSCS instance and to migrate existing DB2 Version 8 MSCS resources to DB2 Version 9.5 DB2 MSCS resources.

### Prerequisites

- Ensure that you have Local Administrator access.
- SYSADM authority is required.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

### Restrictions

- This procedure applies only to migration from DB2 32-bit servers when you install the DB2 Version 9.5 32-bit database product, or from DB2 64-bit servers when you install the DB2 Version 9.5 64-bit database product. The instance bit size is determined by the operating system and the DB2 Version 9.5 database product that you install, see "Support changes for 32-bit and 64-bit DB2 servers" on page 25 for details.
- Additional migration restrictions apply. Review the complete list.

### Procedure

To migrate a DB2 server in an MSCS environment to DB2 Version 9.5:

1. Log on to the DB2 server as a user with Local Administrator authority.
2. Back up your databases.
3. Install DB2 Version 9.5 in all of the nodes in the MSCS cluster. Run the `setup.exe` command to launch the DB2 Setup wizard and select the option Install New on the Install a Product panel. Do not choose the option migrate.
4. Take the resource for the instance offline using the Cluster Administrator. The resource name is the same as the instance name. Ensure that all the remaining resources in the same group as the instance are online.

For more information on using the Cluster Administrator refer to MSCS documentation.

5. Migrate your MSCS instances by running the `db2imigr` command. This command defines a new resource type called "DB2 Server", and updates all DB2 MSCS resources to use the new resource type. Having a new resource type during the migration eliminates conflict with existing DB2 UDB Version 8 MSCS resources.

```
$DB2DIR\bin\db2imigr /u:user,password MSCS-InstName
```

You must run this command from the node that owns all the instance dependent resources.

6. Stop and restart the cluster service in all of the nodes in the MSCS cluster using the Cluster Administrator.
7. Bring online the group of resources containing the migrated instance using the Cluster Administrator.

8. Optional: Migrate your DB2 Administration Server (DAS) if you want to keep your existing DAS configuration and use new functionality available in DB2 Version 9.5. If your DAS is running on DB2 UDB Version 8, you need to migrate it to use the Control Center to administer your DB2 Version 9.5 and Version 9.1 instances. If you choose to create a new DAS, you have to re-configure the DAS settings for your MSCS environment.
9. Migrate your databases.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating DB2 Data Links Manager environments

Migrating a DB2 server where Data Links Manager is installed or where Data Links functionality is enabled from DB2 UDB Version 8 to DB2 Version 9.5 is not supported. However, you can migrate to DB2 Version 9.5 if you remove the Data Links Manager functionality.

### Prerequisites

- Ensure that you have root access on Linux and UNIX operating systems or Local Administrator on Windows.
- Ensure that you have SYSADM authority.
- Ensure that you meet the installation requirements for DB2 database products. The requirements for Linux and UNIX operating systems have changed.
- Review migration recommendations and disk space requirements.
- Perform pre-migration tasks.

### Restrictions

- Review the migration restrictions for DB2 servers.

### Procedure

To migrate a DB2 server in the Data Links environment to DB2 Version 9.5:

1. Remove Data Links Manager from your databases.
2. Drop all references to the DATALINK data type from tables, distinct types, structured types, user-defined functions (UDFs), methods, and dependent objects.
3. If you installed DB2 Net Search Extender (NSE), you need to drop the following UDFs :

```
db2 DROP SPECIFIC FUNCTION DB2EXT.DATALINKCONTENT1;
db2 DROP SPECIFIC FUNCTION DB2EXT.DATALINKCONTENT2;
db2 DROP SPECIFIC FUNCTION DB2EXT.DATALINKCONTENT4;
db2 DROP SPECIFIC FUNCTION DB2EXT.DATALINKCONTENT3;
```

These UDFs are always created by NSE for Data Links support, regardless of Data Links Manager installation. Therefore you need to remove these functions even when Data Links Manager is not installed.

If you plan to migrate by restoring from a database backup, you must drop these UDFs before you back up the database. You cannot restore from a database backup if these UDFs are defined.

4. Uninstall Data Links Manager on the DB2 server that you want to migrate.

5. Update your instances to eliminate the Data Links Manager software and run as a DB2 server only by running the db2iupdt command:

```
db2iupdt instance-name
```

6. Optional: Disable DB2 Data Links functionality by setting the **datalinks** database manager configuration parameter to NO:

```
db2 UPDATE DBM CFG USING datalinks NO
```

When you migrate the instance, the **datalinks** parameter is set to NO.

7. Install DB2 Version 9.5 on your DB2 server. Proceed to step 9, if you are installing DB2 Version 9.5 on Windows and selected to migrate your existing DB2 UDB Version 8 copy.
8. Migrate instances from the same installation path you indicated in step 7.
9. Optional: Migrate the DB2 Administration Server if you want to keep your existing configuration and to administer your DB2 Version 9.5 instances using the Control Center.
10. Migrate databases.

After migrating the DB2 server, perform the recommended post-migration tasks such as resetting the diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that the migration of your DB2 server was successful.

---

## Migrating from XML Extender to native XML data store

You can migrate your database applications from XML Extender to use the native XML data store in DB2 Version 9.5.

DB2 Version 9.5 supports native XML data store in an annotated tree form similar to that of the XML Document Object Model (DOM). This support includes a new XML type, XML indexes and a series of SQL/XML functions.

XML Extender is deprecated in DB2 Version 9.5 and will be discontinued in a future release.

### Prerequisite

A DB2 Version 9.1 or DB2 UDB Version 8 server where XML Extender is installed.

### Procedure

To migrate from the XML Extender to the new native XML storage support:

1. Migrate to DB2 Version 9.5 (Windows) or Migrate to DB2 Version 9.5 (Linux and UNIX).
2. Optional: Convert your databases to Unicode databases. See "Converting non-Unicode databases to Unicode" in *Internationalization Guide*. Although XML type support is provided for non-Unicode databases in DB2 Version 9.5, using a Unicode database eliminates the overhead of character conversion from the database code page to the Unicode code page and preserves the data integrity because there is no character conversion.
3. Add XML type columns to your tables. Use the ALTER TABLE command:

```
db2 ALTER TABLE table_name
ADD column_name XML [NOT NULL]
```

You only need to perform this step if you store intact your XML documents in a column of data type CLOB, VARCHAR, XMLCLOB, XMLVARCHAR, or XMLFILE.

4. Register your XML schemas in the XML Schema repository (XSR). See "Registering and enabling XML schemas for decomposition" in *pureXML Guide* . If you have document type definitions (DTDs), you must convert them to XML schemas and then register them in the XSR. You only need to perform this step only if you want to validate your XML documents.
5. Import XML documents into the table with the new XML data type column.
6. Convert your application to use annotated XML schema decomposition to store content from XML documents in table columns, and the new SQL/XML functions to construct or publish XML using the new XML data type.

Details on all these migration steps and examples of application migration are available in the XML application migration series at [http://www.ibm.com/developerworks/views/db2/libraryview.jsp?search\\_by=viper+migration+series](http://www.ibm.com/developerworks/views/db2/libraryview.jsp?search_by=viper+migration+series).



---

## Chapter 9. Post-migration tasks for DB2 servers

After migrating your DB2 servers, you should perform several post-migration tasks to ensure that your DB2 servers perform as expected and at their optimum level.

Perform the following post-migration tasks that apply to your DB2 server:

1. If you set the *diaglevel* database manager configuration parameter to 3 or higher as recommended in the pre-migration tasks for DB2 servers, reset this parameter to the value set before the migration.
2. Adjust the log space size. If you changed your log space setting as recommended in the pre-migration tasks for DB2 servers, reset the *logfilesiz*, *logprimary*, and *logsecond* database configuration parameters to the values they had before migration. Ensure that the amount of log space that you allocate is adequate for your DB2 server.
3. Activate your database after migration to start up your database and all necessary database services.
4. Manage changes in DB2 server behavior. There are new registry variables, new configuration parameters, and new default values for registry variables and configuration parameters introduced in DB2 Version 9.5 that can impact the behavior of DB2 server. There are also changes in physical design characteristics of databases and changes to security that also have an impact.
5. Set up security to manage database audit in migrated databases. If you enabled the audit facility in your migrated databases, you need to grant security administrator (SECADM) authority to allow users to configure and manage database audit using DDL statements.
6. Convert type-1 indexes to type-2 indexes in migrated databases to take advantage of type-2 index benefits. Also, type-1 indexes are deprecated since DB2 Version 9.1, you should convert them before they are no longer supported.
7. Update the statistics on the system catalog tables. See “Collecting catalog statistics” in *Tuning Database Performance*. During database migration, the statistics collected for system catalog tables are not retained. You must update the statistics for these tables using the RUNSTATS command.
8. Rebind packages in migrated databases to validate packages and to use updated statistics or new index information.
9. Migrate DB2 explain tables, if you need to retain explain table information that you previously gathered.
10. Ensure that you meet system temporary table spaces page sizes requirements to accommodate the largest row size in your result sets from queries or positioned updates, and create a system temporary table space with a larger page size if necessary.
11. If you obtained customized code page conversion tables from the DB2 support service, copy all of the files for those tables from the DB2OLD/conv to DB2DIR/conv, where DB2OLD is the location of your DB2 Version 9.1 or DB2 UDB Version 8 copy and DB2DIR is the location of your DB2 Version 9.5 copy. You do not need to copy standard code page conversion tables.

If you migrated your existing DB2 Version 9.1 or DB2 UDB Version 8 copy on Windows operating systems, you can restore the customized code page

conversion tables that you backed up as part of the pre-migration tasks for DB2 servers to the DB2PATH\conv directory, where DB2PATH is the location of your DB2 Version 9.5 copy.

12. If you created write-to-table event monitors in DB2 Version 9.1 or DB2 UDB Version 8, you need to re-create your write-to-table event monitors so that you can successfully activate these monitors after you migrate to DB2 Version 9.5.
13. Verify that your DB2 server migration was successful. Test your applications and tools to ensure that the DB2 server is working as expected.
14. Back up your databases after the migration is complete.

Perform the following post-migration tasks that apply to your DB2 products or add-on features:

- If you migrate a DB2 server running high availability disaster recovery (HADR) replication, initialize HADR replication. See “Initializing high availability disaster recovery (HADR)” in *Data Recovery and High Availability Guide and Reference*. During migration to DB2 Version 9.5 in a high availability disaster recovery (HADR) replication environment, a database role is changed from primary to standard. Migration of standby databases is not supported because these databases are in roll forward pending state.
- If you are using index extensions or spatial indexes and you migrated from a DB2 UDB Version 8 32-bit instance to a DB2 Version 9.5 64-bit instance, you need to re-create your index extensions or spatial indexes. If you are a Spatial Extender user, review the migrating the Spatial Extender environment task for details on how to re-create your spatial indexes. The DB2 Spatial Extender and Geodetic Data Management Feature User’s Guide and Reference is available by at <http://www.ibm.com/software/data/spatial/db2spatial/library.html>.

Once your DB2 server performance is stable, take advantage of optimizer improvements and collect statistics for new features by updating statistics for your migrated databases. During database migration to DB2 Version 9.5, the statistics collected from your existing database tables retain their values. Statistics for new characteristics on tables and indexes have a value of -1 to indicate there is no information gathered. However, you only need these statistics if you are using new functionality.

After updating statistics for your migrated databases, determine if index or table reorganization is necessary by running the REORGCHK command. Table and index reorganization can help you to improve performance.

At this point, you should resume all of your maintenance activities such as backing up databases and updating statistics. You should also remove any DB2 Version 9.1 or DB2 UDB Version 8 copies that you no longer need.

---

## Adjusting the log space size in migrated databases

You need to set the appropriate size for log files since it is one of the important factors in tuning your DB2 server. Also, if you increased the log files sizes as a pre-migration task, you can restore additional free space to your DB2 server.

### Prerequisite

You must have SYSCTRL or SYSADM authority in order to be able to increase the size of table spaces and log space.

### Restriction

On a partitioned database environment, you only need to adjust the log space size on the catalog database partition server.

### Procedure

1. Connect to the database that you migrated:

```
db2 CONNECT TO sample
```

where *sample* is the database name.

2. Restore your log file size settings to the values you had before migration:

```
db2 UPDATE DB CFG FOR sample using LOGSECOND previous-value
```

where *previous-value* is the setting that you save before migration and *sample* is the database name. In the pre-migration task, only the **logprimary** and the **logsecond** parameters were changed. If you change the setting for the **logfilsiz** parameter, you should restore the previous value.

If you enabled infinite active logging, disable it by running the following commands:

```
db2 UPDATE DB CFG FOR sample using LOGARCHMETH1 previous-value
db2 UPDATE DB CFG FOR sample using LOGSECOND previous-value
```

where *previous-value* is the setting that you save before migration and *sample* is the database name.

3. Optional: Increase your log file size settings. The RID for log records has increased in the amount of 2 bytes, depending on the type of log record this could represent less than 2% increase in the log record size.

In general, your current setting for log space should be sufficient to accommodate this change. However, if you have a concern that your log space setting is undersized, monitor the log space usage to find out the appropriate size. The following example increases log file size by 5% to accommodate the log record size increase:

```
db2 UPDATE DB CFG FOR sample using LOGFILSIZ previous-value*1.05
```

where *previous-value* is the setting that you save before migration and *sample* is the database name.

4. Disconnect from the database that you migrated:

```
db2 CONNECT RESET
```

LOGFILSIZ changes only take effect when the database is reactivated. All applications must first disconnect from the database then deactivate and activate the database again.

---

## Activating a database after migration

Activating your database allows you to ensure that all database services are running properly and to address any problems that might occur during the database activation. You can also eliminate the overhead on DB2 clients that have to wait until the database manager starts up the database to get a connection to this database.

To activate your databases after migration:

1. Start up your database and all necessary database services with the **ACTIVATE DATABASE** command. The following example illustrates the use of this command to activate the *sample* database:

```
db2 ACTIVATE DATABASE sample
```

After this command is executed successfully your database is available for connections.

2. Review the administration notification log or the db2diag.log file to verify that all database services are running properly and all buffer pools are activated. Address any problems that occurred during the database activation.

Remember that a database, activated by the `ACTIVATE DATABASE` command, stops only when you issue the `DEACTIVATE DATABASE` command or the `db2stop` command. If the database is activated when the first connection is established, then the database is stopped when the last connection is closed.

---

## Managing DB2 server behavior changes

The changes in DB2 registry variables, configuration parameters, and database physical design characteristics can have a migration impact. Review these changes to manage the migration impact.

After migrating your DB2 server, compare the values of your registry variables and configuration parameters to their values before migration. If you find any differences, take the time to understand them because they could alter the behavior or performance of your applications. However, consider carefully whether to disable any new features because they provide support for new resources needed by the database manager. You should disable new features only if you experience negative performance or unwanted behavior.

To manage DB2 server behavior changes:

1. Review the information about new, changed, and discontinued registry variables, and based on the migration impact, choose the appropriate settings:
  - New registry variables
  - Changed registry variables
  - Deprecated and discontinued registry variables , see “What’s changed” in *What’s New*
2. Set your DB2 global profile registry variables. The variables that you set at the global profile level, using the `db2set` command with the `-g` option, are not migrated. The global profile variables apply to all instances pertaining to a specific DB2 copy. Therefore, after migrating your instances, use the configuration information that you saved in the pre-migration tasks to restore the values of your global profile registry variables for every DB2 Version 9.5 copy.
3. Review the information about changed and deprecated database manager configuration parameters, and based on the migration impact, choose the appropriate settings:
  - Changed database manager configuration parameters
  - Deprecated database manager configuration parameters
4. Review the information about new, changed, deprecated, and discontinued database configuration parameters, and based on the migration impact, choose the appropriate settings:
  - New database configuration parameters
  - Changed database configuration parameters
  - Deprecated and discontinued database configuration parameters
5. Review the changes in database physical design characteristics and security, and based on the migration impact, modify database objects accordingly:

- Physical design characteristics of databases
- Authorities and privileges

If you change the settings of any database manager configuration parameters that are not dynamic, you might need to restart the instance so the new settings take effect.

---

## Setting up security to manage database auditing in migrated databases

Security administrator (SECADM) authority is now required to configure and manage database auditing using SQL statements, the SYSADM authority is not a requirement anymore. Grant SECADM authority to the users who manage the database auditing in migrated databases.

### Prerequisites

To grant SECADM authority and run the db2audit command, you must have SYSADM authority.

Database and instance-level auditing are separate in DB2 Version 9.5. You can configure database auditing only by using DDL statements. You can continue to use the db2audit command to configure instance auditing.

When you migrate an instance, the audit configuration file is converted to DB2 Version 9.5 format.

When you migrate a database, the instance-level configuration settings for auditing are used to create an audit policy in the database. If the audit facility is enabled at the instance level, the audit policy is associated with the migrated database to enable auditing. Otherwise, the audit policy is not associated. These actions ensure that you observe the same audit behavior on your database after migrating to DB2 Version 9.5.

### Procedure

To setup security to manage database auditing in migrated databases:

1. Grant SECADM authority to the users who manage the audit facility by using the GRANT command. The following sample commands show how to grant SECADM authority to a user:

```
db2 CONNECT TO SAMPLE
db2 GRANT SECADM ON DATABASE TO USER <user-id>
```

2. Verify that the DB2AUDIT\_CFG\_MIGR audit policy was created for your databases during migration by querying the SYSCAT.AUDITPOLICIES system catalog view. The following sample query determines whether this audit policy was created:

```
db2 "SELECT * FROM SYSCAT.AUDITPOLICIES A
 WHERE A.AUDITPOLICYNAME = 'DB2AUDIT_CFG_MIGR'"
```

If the DB2AUDIT\_CFG\_MIGR audit policy was not created during migration, create it using the CREATE AUDIT POLICY statement.

3. Verify that the DB2AUDIT\_CFG\_MIGR audit policy was associated with the migrated databases by querying the SYSCAT.AUDITUSE system catalog view. The following sample query determines whether the audit policy was associated with the SAMPLE database:

```
db2 "SELECT * FROM SYSCAT.AUDITUSE U
 WHERE U.OBJECTNAME = 'SAMPLE'"
```

If the DB2AUDIT\_CFG\_MIGR audit policy cannot be associated to your database during migration, use the AUDIT statement to associate this policy to your database.

4. Optional: If you want to extract all audit records from the original audit log file that you had before migration and place the contents into a new audit log file in the new default location, run the db2audit command with the **extract** parameter. The original audit log file from a previous release remains in the same location that it was in prior to database migration.

The new default location for the audit logs in DB2 Version 9.5 is:

- INSTHOME/sqlib/security/auditdata on Linux and UNIX operating systems, where INSTHOME is the instance home directory.
- INSTHOME\security\auditdata on Windows operating systems

where INSTHOME is the instance home directory that stores user data and instance directories.

Now, you can use the following DDL statements to manage database auditing:

- CREATE AUDIT POLICY
- ALTER AUDIT POLICY
- AUDIT

---

## Converting type-1 indexes to type-2 indexes in migrated databases

You must consider converting any existing type-1 indexes to type-2 indexes after migration in order to improve performance and use automatic maintenance features.

Check the INSPECT CHECK and REORG INDEXES/TABLE command references for details on the required authorization.

All new indexes created in DB2 Version 9.5 are type-2 indexes, except when you create an index on a table that already had type-1 indexes, in which case the new index is also type-1. You can only have type-1 indexes on databases that you migrated to DB2 Version 9.5 that were created on DB2 UDB Version 7 or earlier.

The advantages of type-2 indexes are to improve concurrency because the use of next-key locking is reduced to a minimum, and use columns with length greater than 255 bytes as part of the index key. A table must have only type-2 indexes before online table REORG and online table LOAD commands can be used against the table. This restriction also applies to DB2 Version 9.5, in addition to other restrictions such as not being able to add XML type columns to a table with type-1 indexes.

1. Verify if you have type-1 indexes by using the INSPECT command:

```
db2 INSPECT CHECK DATABASE RESULTS KEEP sample.log
db2inspf $INSTHOME/sqlib/db2dump/sample.log sample.out
```

The formatted output from the db2inspf command in the sample.out file shows the type of index for each table:

```
...
Table phase start (ID Signed: 83, Unsigned: 83; Tablespace ID: 0) :

Data phase start. Object: 83 Tablespace: 0
```

```
The index type is 2 for this table.
DAT Object Summary: Total Pages 1 - Used Pages 0 - Free Space 70 %
Data phase end.
```

```
Index phase start. Object: 83 Tablespace: 0
INX Object Summary: Total Pages 3 - Used Pages 3
Index phase end.
Table phase end.
...
```

2. If you have type-1 indexes, you can easily convert them to type-2 indexes using the REORG INDEXES/TABLE command:

```
db2 REORG INDEXES ALL FOR TABLE employee CONVERT
```

If you plan to reorganize your indexes, it is an excellent opportunity to add the CONVERT option because this option converts only your type-1 indexes and has no effect on your type-2 indexes.

Another advantage of converting to type-2 indexes using the REORG INDEXES/TABLE command is that you will also convert unique indexes created on your database before DB2 UDB Version 5. Alternatively, if you are not converting your type-1 indexes with this command, you must run the db2uidl command to generate the CREATE UNIQUE INDEX statements in a script. At your convenience you can run this script to convert these unique indexes to DB2 Version 9.5 semantics.

---

## Rebinding packages in migrated databases

During database migration, all packages for user applications and routines are marked as invalid. You need to rebind invalidated packages to take advantage of changes in the DB2 server and new statistics. Packages are also placed into an invalid state if they are dependent on a database object that you dropped such as tables, views, aliases, indexes, triggers, referential constraints and table check constraints. If you drop a UDF, your package is placed into an inoperative state.

Packages will be implicitly rebound the first time an application uses them after migrating your database. To eliminate this overhead, you can rebind invalid packages by running the REBIND command or the db2rbind command after the migration process is complete. You must explicitly rebind inoperative packages.

### Prerequisite

Ensure that you have SYSADM authority.

### Restriction

This procedure only applies to Embedded SQL database applications programmed in C, C++, COBOL, FORTRAN, and REXX™.

### Procedure

To rebind packages in migrated databases:

1. Log on as a user with SYSADM authority.
2. Rebind all invalid packages in each database by running the db2rbind command:

```
db2rbind database-name -l logfile all -u userid -p password
```

The all clause rebinds valid and invalid packages. Review the *logfile* file and address any issues rebinding any database packages.

3. Verify that your DB2 server migration was successful. Test your applications and tools to ensure the server is working as expected.

---

## Migrating explain tables

The MIGRATE DATABASE command does not migrate the explain tables. If you need to maintain explain table information that you previously gathered in your existing DB2 copies from previous releases, you need to migrate your explain tables to DB2 Version 9.5.

You can manually migrate your explain tables after you migrate your database, or you can later re-create the explain tables and gather new information.

### Prerequisite

Ensure that you have SYSADM or DBADM authority.

### Procedure

To migrate the explain tables:

1. Run 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, are migrated. The explain tables migration tool renames the existing explain tables, creates a new set of tables using the EXPLAIN.DDL file, and copies the contents of the existing explain tables to the new tables. Finally, it drops the existing explain tables. The db2exmig command preserves any user added columns on the explain tables.

2. Use Visual Explain to see a graphical display of a query access plan or the db2expln command to see the access plan information in the migrated explain tables.

---

## Ensuring system temporary table spaces page sizes meet requirements

The use of larger record identifiers (RID) increases the row size in your result sets from queries or positioned updates. If the row size in your result sets is close to the maximum row length limit for your existing system temporary table spaces, you might need to create a system temporary table space with a larger page size.

### Prerequisite

Ensure that you have SYSCTRL or SYSADM authority to create a system temporary table space if required.

### Procedure

To ensure that the maximum page size of your system temporary table space is large enough for your queries or positioned updates:

1. Determine the maximum row size in your result sets from queries or positioned updates. Monitor your queries or calculate the maximum row size using the DDL statement that you used to create your tables.
2. List your table spaces using the LIST TABLESPACES command, as shown in the following example:

```

db2 LIST TABLESPACES SHOW DETAIL
...
Tablespace ID = 1
Name = TEMPSPACE1
Type = System managed space
Contents = System Temporary data
State = 0x0000
 Detailed explanation:
 Normal
Total pages = 10
Useable pages = 10
Used pages = 10
Free pages = Not applicable
High water mark (pages) = Not applicable
Page size (bytes) = 4096
Extent size (pages) = 32
Prefetch size (pages) = 320
Number of containers = 10
...

```

You can identify the system temporary table spaces in the output by looking for table spaces whose Contents fields have a value of System Temporary data. Take note of the page size for each of your system temporary table spaces and the page size of the table spaces where the tables referenced in the queries or updates were created.

3. Check whether the largest row size in your result sets fits into your system temporary table space page size:

```

maximum_row_size > maximum_row_length - 8 bytes (structure overhead in
 single partition)
maximum_row_size > maximum_row_length - 16 bytes (structure overhead in DPF)

```

where `maximum_row_size` is the maximum row size for your result sets, and `maximum_row_length` is the maximum length allowed based on the largest page size of all of your system temporary table spaces. Review the "SQL and XML limits" in *SQL Reference, Volume 1* to determine the maximum row length per table space page size.

If the maximum row size is less than the calculated value then your queries will run in the same manner that they did in DB2 UDB Version 8, and you do not need to continue with this task.

4. Create a system temporary table space that is at least one page size larger than the table space page size where the tables were created if you do not already have a system temporary table with that page size. For example, on the Windows operating systems, if you created your table in a table space with 4 KB page size, create the additional system temporary table space using an 8 KB page size:

```

db2 CREATE SYSTEM TEMPORARY TABLESPACE tmp_tbsp
 PAGESIZE 8K
 MANAGED BY SYSTEM
 USING ('d:\tmp_tbsp','e:\tmp_tbsp')

```

If your table space page size is 32 KB, you can reduce the information that you are selecting in your queries or split the queries to fit in the system temporary table space page. For example, if you select all columns from a table, you can

instead select only the columns that you really required or a substring of certain columns to avoid exceeding the page size limitation.

---

## Re-creating write-to-table event monitors

If you created write-to-table event monitors in DB2 Version 9.1 or DB2 UDB Version 8, re-create your write-to-table event monitors so that you can successfully activate these monitors after you migrate to DB2 Version 9.5.

The target tables now include new columns for new monitor elements, changed column data types, or longer column lengths for write-to-table event monitors. You need to re-create your existing write-to-table event monitors to re-create your target tables and to be able to use new Version 9.5 monitor elements.

To re-create write-to-table event monitors:

1. Identify which are the target tables for each write-to-table event monitor that you created in DB2 Version 9.1 or DB2 UDB Version 8 by querying the SYSCAT.EVENTTABLES view as shown in the following example:

```
SELECT TABSCHEMA, TABNAME FROM SYSCAT.EVENTTABLES
 WHERE EVMONNAME = 'write-to-table-event-monitor-name'
```

2. Rename or drop the existing target tables that you identified in the previous step by issuing one of the following statements for each target table:

```
RENAME TABLE target-table-name TO new-target-table-name
or
DROP TABLE target-table-name
```

You only need to rename the target tables if you want to keep the existing data that you collected.

3. Drop the write-to-table event monitors by issuing the following statement for each event monitor:

```
DROP EVENT MONITOR write-to-table-event-monitor-name
```

4. Create your write-to-table event monitors.
5. If you created your write-to-table event monitors without the **AUTOSTART** command parameter, activate the write-to-table event monitor to start collecting data by issuing the SET EVENT MONITOR STATE statement as shown in the following example:

```
SET EVENT MONITOR write-to-table-event-monitor-name 1
```

If you have applications that query target tables, you need to modify your applications to manage the changes.

---

## Verifying migration of DB2 servers

When the migration of your DB2 server is complete, it is a good measure to run some tests on the new migrated environment to verify that the DB2 server is working as expected. These tests can consist of batch programs that you usually run against the DB2 server or any programs or scripts that you run for benchmarks.

If you have DB2 command scripts with SQL statements, you can use the db2batch benchmark tool command to execute the statements in these scripts, and gather performance information details and statistics such as CPU time and elapsed time. This tool can work in both a single partition database and in a multiple partition database.

## Prerequisite

Ensure that you have the same authority level that is required to run the SQL statements in your script.

## Procedure

To verify that your DB2 server migration was successful:

1. Log on to the DB2 server as a user with the same authority level that is required to run the SQL statements in the script.
2. Prepare a script with SQL statements that you frequently run. If you installed the sample files, you can also run any of the sample CLP scripts.
3. Run your script using the `db2batch` command. The following example shows you how to run this tool with the `testdata.db2` sample script:

```
cd samplefile-dir-clp
db2batch -d sample -f testdata.db2 -o r 0 p 3
```

where *samplefile-dir-clp* is `DB2DIR/samples/clp` on Linux and UNIX and `DB2DIR\samples\clp` on Windows, `DB2DIR` represents the location for your DB2 Version 9.5 copy, `sample` is the database name, and the option `-o r 0 p 3` indicates to print 0 fetched rows to the output and to report elapsed time, CPU time, and summary of monitoring information for each statement in the `testdata.db2` script.

The following text is an extract of the summary table output generated by the command in the previous example:

Summary Table:

| Type      | Number | Total Time | Min Time | Max Time | Arithmetic Mean | Geometric Mean |
|-----------|--------|------------|----------|----------|-----------------|----------------|
| Statement | 1      | 0.281284   | 0.281284 | 0.281284 | 0.281284        | 0.281284       |
| Statement | 2      | 0.073158   | 0.073158 | 0.073158 | 0.073158        | 0.073158       |
| Statement | 3      | 0.000823   | 0.000823 | 0.000823 | 0.000823        | 0.000823       |
| Statement | 4      | 0.155366   | 0.155366 | 0.155366 | 0.155366        | 0.155366       |

```
* Total Entries: 4
* Total Time: 0.510630 seconds
* Minimum Time: 0.000823 seconds
* Maximum Time: 0.281284 seconds
* Arithmetic Mean Time: 0.127658 seconds
* Geometric Mean Time: 0.040271 seconds
```



---

## Chapter 10. Enabling new DB2 Version 9.5 functionality in migrated databases

After migrating your DB2 server, enable new functionality to enhance the functionality and improve the performance of your migrated databases.

### Prerequisites

You must migrate your DB2 server to DB2 Version 9.5.

### Procedure

Perform the following steps to enable some of the DB2 Version 9.5 functionality in your migrated DB2 environment:

- Enable automatic agent configuration for your databases to ensure that the number of agents and connections is not limited by the values that you set for memory parameters by setting the following database manager configuration parameters to AUTOMATIC:

```
db2 ATTACH TO instance-name
db2 UPDATE DBM CFG USING max_coordagents AUTOMATIC
db2 UPDATE DBM CFG USING num_poolagents AUTOMATIC
db2 UPDATE DBM CFG USING max_connections AUTOMATIC
```

Because these database manager configuration parameters are configurable online, issuing the ATTACH command causes the UPDATE DBM CFG command to apply the changes immediately. If you do not want the change applied immediately, use the UPDATE DBM CFG command with the **DEFERRED** clause.

- Enable the use of non-buffered I/O operations (Concurrent I/O or Direct I/O) in your existing table spaces by using the ALTER TABLESPACE statement with the NO FILE SYSTEM CACHING clause:

```
db2 ALTER TABLESPACE tablespace-name NO FILE SYSTEM CACHING
```

Starting in DB2 Version 9.5 on certain platforms, the default for I/O operations is NO FILE SYSTEM CACHING when you create a table space without specifying whether I/O operations are cached at the file system level. Refer to File system caching configurations for details about platforms where the default option is changing.

For optimal performance of non-buffered I/O, adjust the size of buffer pools. To let the database manager automatically set the buffer pool size, enable the Self Tuning Memory Manager (STMM), and set the buffer pool size to AUTOMATIC:

```
db2 UPDATE DB CFG FOR database-name USING self_tuning_mem ON
db2 ALTER BUFFERPOOL bufferpool-name SIZE AUTOMATIC
```

In addition to enabling buffer pool self tuning, you must enable at least one more memory consumer to activate the memory tuner.

- Enable automatic statistics collection that use real-time statistics by setting the **auto\_stmt\_stats** parameter to ON and increasing the value of the **catalogcache\_sz** parameter by 25% as shown in the following example:

```
db2 UPDATE DB CFG FOR database-name USING auto_stmt_stats ON
 auto_runstats ON auto_tbl_maint ON auto_maint ON
db2 UPDATE DB CFG FOR database-name USING catalogcache_sz maxappls*5
```

In this example, the value of the **catalogcache\_sz** parameter is set to five times the value of the **maxappls** parameter to increase by 25% the pre-migration value which was four times the value of the **maxappls** parameter (default value in previous releases).

- Enable the use of the Workload Manager functionality. After migration, customize the execution environment for your DB2 server to maximize performance by creating user-defined service classes and workloads. In migrated databases, all connections belong to the default workload and are mapped to the default user service class.
- In partitioned database environments, take advantage of single system view backups by issuing the BACKUP DB command with the ON ALL DBPARTITIONNUMS clause:

```
db2 BACKUP DB sample ON ALL DBPARTITIONNUMS TO directory
```

where *directory* is the target directory and must exist in all database partitions. Because multiple backups are running simultaneously, overall system performance is impacted.

After restoring a single system view backup, you can take advantage of the new TO END OF BACKUP clause in the ROLLFORWARD DB command to process the log files until the end of backup time so that all of the database partitions are synchronized and in a consistent state.

If you migrated your DB2 server from DB2 UDB Version 8, enable functionality introduced in DB2 Version 9.1 in your migrated DB2 environment

---

## Chapter 11. Reversing DB2 server migration

Reversing DB2 server migration involves creating a plan using the steps on this procedure. There is no utility to reverse the migration of a DB2 server.

Performing a migration in a test environment will help you identify any issues with the process and avoid having to reverse the migration.

### Prerequisites

- Ensure that you have SYSADM authority, as well as root on Linux and UNIX operating systems or Local Administrator authority on Windows operating systems.
- Perform the following steps before migrating your DB2 server:
  - Review migration recommendations and disk space requirements.
  - Take an offline full backup of all databases that you are going to migrate.
  - Back up all database manager configuration parameter values for each instance and all database configuration parameter values for each database.
  - Perform other pre-migration tasks that apply to your environment.
- Keep your existing DB2 Version 9.1 or DB2 UDB Version 8 copy during migration of your DB2 server. To do this, select the option Install New to create a new copy when installing DB2 Version 9.5. Do not select the option Migrate on Windows operating systems.

### Restrictions

- This procedure applies only to DB2 server migration. It does not include DB2 clients.
- In partitioned database environments you must perform this procedure on all participating database partition servers. If you have several database partitions on a partition server, execute tasks at the database level, such as backup and restore, on each database partition.
- Additional migration restrictions apply. Review the complete list.

### Procedure

To reverse a migration you need to perform the following steps:

1. Log on to the DB2 server as a user with SYSADM authority.
2. Drop all databases in DB2 Version 9.5 by running the DROP DATABASE command.
3. Log on to the DB2 server as root on Linux and UNIX operating systems or a user with Local Administrator authority on Windows operating systems.
4. Drop your DB2 Version 9.5 instances by running the db2idrop command. This command does not remove the database files; you need to drop your databases before dropping your instances.
5. If you migrated your DB2 Version 9.1 or DB2 UDB Version 8 instances to DB2 Version 9.5, re-create your instances in DB2 Version 9.1 or DB2 UDB Version 8 by running the db2icrt. Then restore the database manager configuration parameter values for each instance using the UPDATE DATABASE MANAGER CONFIGURATION command.

6. For each DB2 Version 9.1 or DB2 UDB Version 8 instance, log on to the DB2 server as the instance owner and restore your migrated databases from a previous release offline full backup by running the RESTORE DATABASE command. You cannot migrate your databases from DB2 Version 9.5 to DB2 Version 9.1 or DB2 UDB Version 8.

If you recreated the instances using the same instance owner they had prior to migration and you did not migrate a database to a DB2 Version 9.5 instance, the database is still in DB2 Version 9.1 or DB2 UDB Version 8 and you can access it by just re-cataloging it.

---

## Part 3. Migrating clients

This part of the book contains the following chapters:

- Chapter 12, "Migration for clients," on page 95
- Chapter 13, "Migration essentials for clients," on page 97
- Chapter 14, "Pre-migration tasks for clients," on page 101
- Chapter 15, "Migrating to Data Server Client (Windows)," on page 103
- Chapter 16, "Migrating to Data Server Runtime Client (Windows)," on page 105
- Chapter 17, "Migrating clients (Linux and UNIX)," on page 107
- Chapter 18, "Post-migration tasks for clients," on page 109



---

## Chapter 12. Migration for clients

Upgrading to DB2 Version 9.5 might require migrating your clients.

Migrating a client involves installing a Version 9.5 client and then migrating the client instance. A client instance allows you to connect your application to a database and keeps the information about your client configuration, your cataloged nodes, and your cataloged databases.

The current level of client that you have installed determines the way to proceed with migration to DB2 Version 9.5. You can directly migrate to Version 9.5 clients from Version 8 or Version 9.1 clients. If you have Version 7 or earlier clients, you need to migrate to any Version 8 client first.

Review migration essentials for clients for details about migration support and options available for clients.



## Chapter 13. Migration essentials for clients

Migrating clients to DB2 Version 9.5 requires an understanding of migration concepts, migration options, migration restrictions, migration recommendations, and connectivity between clients and DB2 servers.

After you have a complete understanding of what migrating your clients involves, you can create your own plan to successfully migrate your clients to DB2 Version 9.5.

### Migration options for clients

The migration options vary depending on the type of client that you want to install. The following table describes the migration options for each type of Version 9.5 client:

Table 19. Migration options for Version 9.5 clients

| Migrating from                                                                                                                                                                        | Migrating to                                    | Migration support details                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>Version 8 DB2 Administration Client</li> <li>Version 8 DB2 Application Development Client</li> <li>Version 9.1 DB2 Client</li> </ul> (Windows) | Version 9.5 Data Server Client(Windows)         | You have two options: <ul style="list-style-type: none"> <li>Install the Version 9.5 Data Server Client, and choose the migrate action in the <b>Work with Existing</b> window. The client instance is then automatically migrated for you.</li> <li>Install a new copy of the Version 9.5 Data Server Client, and then manually migrate Version 9.1 or Version 8 client instances.</li> </ul> |
| <ul style="list-style-type: none"> <li>Version 8 DB2 Run-Time Client</li> <li>Version 8 DB2 Run-Time Client Lite</li> <li>Version 9.1 DB2 Runtime Client</li> </ul> (Windows)         | Version 9.5 Data Server Runtime Client(Windows) | <ul style="list-style-type: none"> <li>Install the Version 9.5 Data Server Runtime Client as a new copy, and then manually migrate your Version 9.1 or Version 8 client instance.</li> </ul>                                                                                                                                                                                                   |
| All Version 9.1 or Version 8 clients (Linux or UNIX)                                                                                                                                  | All Version 9.5 clients (Linux or UNIX)         | <ul style="list-style-type: none"> <li>Install a new copy of any Version 9.5 client, and then manually migrate your Version 9.1 or Version 8 client instance.</li> </ul>                                                                                                                                                                                                                       |

When you migrate a client instance, the bit size is determined by the operating systems where you installed the Version 9.5 client. Refer to Table 6 on page 25 for details.

### Migration restrictions for clients

Review “Migration restrictions for DB2 servers” on page 18 for information regarding instance migration and operating system support. These restrictions also apply to clients and can impact their migration.

If you installed a Version 8 client on the same system as a DB2 Version 9.5 server or if you installed a Version 9.5 client on the same system as a DB2

Version 8 server, connections to the databases on the DB2 server from the client cataloged using a local node are not supported. You should migrate both the DB2 server and the client to DB2 Version 9.5. If you do not migrate the Version 8 client or the DB2 Version 8 server, you can only connect to the databases that are cataloged using TCP/IP nodes. Review the “Recataloging nodes and databases using TCP/IP protocol” on page 109 post-migration task for details.

Also, the trusted context capability supports only the TCP/IP protocol. Any connections to migrated databases that you cataloged using a local node are unable to use this capability unless you recatalog the nodes using the TCP/IP protocol.

### Connectivity support between clients and DB2 servers

In DB2 Version 9.5, the following support for connectivity between clients and DB2 servers is available:

Table 20. DB2 Version 9.5 connectivity support

| Client                               | DB2 server                                | Client connectivity support                                         |
|--------------------------------------|-------------------------------------------|---------------------------------------------------------------------|
| 32-bit or 64-bit Version 9.5 clients | 32-bit or 64-bit DB2 Version 9.5 server   | Any Version 9.5 clients can establish 32-bit or 64-bit connections. |
| 32-bit or 64-bit Version 9.5 clients | 32-bit or 64-bit DB2 Version 9.1 server   | Only DB2 Version 9.1 functionality is available.                    |
| 32-bit or 64-bit Version 9.5 clients | 32-bit or 64-bit DB2 UDB Version 8 server | Only DB2 UDB Version 8 functionality is available.                  |
| 32-bit or 64-bit Version 9.1 clients | 32-bit or 64-bit DB2 Version 9.5 server   | Only DB2 Version 9.1 functionality is available.                    |
| 32-bit or 64-bit Version 8 clients   | 32-bit or 64-bit DB2 Version 9.5 server   | Only DB2 UDB Version 8 functionality is available.                  |

Connections to DB2 Version 9.5 servers from a client release prior to Version 8 are not supported.

### New client and new names for existing clients

In DB2 Version 9.5, a new client product is available, and there are new names for existing clients. Refer to “New DB2 client product simplifies deployment (Windows)” in *What’s New* for details about the new name and generic client names. In the migrating client tasks, the term *pre-V9.5 clients* refers to Version 9.1 and Version 8 clients.

### Migration from Version 8 clients

If you are migrating from Version 8 clients, review Migration essentials for DB2 clients in the DB2 Information Center Version 9.1 to learn about additional support changes that can also impact your migration.

---

## Migration best practices for clients

Consider the following best practices when planning your client migration.

### Migrate clients after migrating DB2 servers

In general, you should migrate clients after you migrate your DB2 servers. Version 9.1 and Version 8 clients can connect to Version 9.5 DB2 servers. The only restriction is that new DB2 Version 9.5 features are not available

to pre-V9.5 clients. If you plan to use these features in your applications, you need to migrate your clients to DB2 Version 9.5 or install new Version 9.5 client copies.

If you migrate your clients before you migrate your DB2 servers, you need to be aware that there are known support limitations for connectivity from a Version 9.5 client to a Version 9.1 DB2 server. Refer to “Supported combinations of client and server versions” in *Quick Beginnings for IBM Data Server Clients* to determine if these limitations apply to your application, and take necessary actions.

### **Migrate your clients in a test environment**

Migrating clients in a test environment allows you to determine if the migration can be successful and to address any problems that might occur during the migration process. You can also test your database applications and determine if you need to migrate them to run successfully in DB2 Version 9.5.

### **Install a new client copy instead of migrating existing client**

If you have software that requires a Version 9.1 or Version 8 client, you should install the Version 9.5 client as a new copy and keep your Version 9.1 or Version 8 client copy to satisfy the software requirement. You need to create a Version 9.5 client instance and keep your existing Version 9.1 or Version 8 client instance with its configuration. You can select the option to create a new client instance during the installation, or you can manually create the client instance after installation.

### **Perform pre-migration and post-migration tasks**

Perform the pre-migration and post-migration tasks for clients to ensure a successful migration.



---

## Chapter 14. Pre-migration tasks for clients

Before you migrate your clients, you should complete certain tasks to help ensure that your migration is successful.

Prepare for the migration of your clients by performing the following tasks:

1. Review the migration essentials for clients to determine which factors might impact your client migration.
2. Review the supported and non-supported client configurations.
3. Plan your migration strategy. For example, you might need to migrate your DB2 server first, then your clients.
4. Optional: Migrate your DB2 servers.
5. Back up your client configuration information.
6. Optional: Migrate your clients in a test environment to identify migration issues and to verify that applications, scripts, tools and routines work as expected before migrating your production environment.

---

### Backing up client configuration information

Before you migrate, you should back up the database manager configuration parameter settings of your client instance and the information details about all of your cataloged databases. With this information, you can restore your previous client configuration and cataloged databases after migration, if necessary.

#### Prerequisites

Ensure that you have SYSADM or SYSCTRL authority to run the db2cfexp command.

#### Restrictions

This procedure describes how to back up the configuration information for only one client. If you have different configuration settings on each client, you need to back up the configuration information for each client.

#### Procedure

To back up your client configuration information:

1. Back up your database manager configuration parameter settings by using the GET DATABASE MANAGER CONFIGURATION command to list your settings for the parameters and redirect the command output to a file as shown in the following example:

```
db2 GET DBM CFG > D:\migration\dbm_client.cfg
```

2. Back up the information of cataloged databases by running the db2cfexp command to create a configuration profile:

```
db2cfexp cfg_profile BACKUP
```

The BACKUP option creates the cfg\_profile file as a configuration profile of the client instance that contains all of the instance configuration information, including the registry profile settings and information of a specific nature relevant only to this client instance. You can also use the DB2 Configuration Assistant to export your configuration profile.

---

## Migrating clients in a test environment

Migrating clients in a test environment before you migrate them in your production environment allows you to address problems during the migration process more effectively and to evaluate the impact of changes introduced in DB2 Version 9.5.

### Prerequisite

You must have root authority on Linux and UNIX operating systems or Local Administrator authority on Windows. You must also have SYSADM authority.

### Procedure

To duplicate your production environment in a test environment, you need to do the following tasks:

1. Install the same client and version that you have in your production environment in a test system.
2. Re-create your client instance by running the `db2icrt` command with the `-s` option:

| Operating system | DB2 command                                                          |
|------------------|----------------------------------------------------------------------|
| Windows          | <code>"%DB2PATH%" \bin\db2icrt -s client<br/><i>InstName</i></code>  |
| Linux and UNIX   | <code>\$DB2DIR/instance/db2icrt -s client<br/><i>InstName</i></code> |

where `DB2PATH` and `DB2DIR` are set to the location of the client copy that you installed in the previous step, and `InstName` is the name of the instance.

3. Perform the pre-migration tasks that apply to your client.
4. Install a Version 9.5 client that you can migrate to depending on the client that you are migrating from. Select the option **Install New** to install a new copy. Refer to Table 19 on page 97 to determine the client that you need to install.
5. Migrate your client instance by running the `db2imigr` command:

| Operating system | DB2 command                                             |
|------------------|---------------------------------------------------------|
| Windows          | <code>"%DB2PATH%" \bin\db2imigr <i>InstName</i></code>  |
| Linux and UNIX   | <code>\$DB2DIR/instance/db2imigr <i>InstName</i></code> |

where `DB2PATH` and `DB2DIR` are set to the location of the Version 9.5 client copy that you installed in the previous step, and `InstName` is the name of the instance.

6. If you found any issues migrating your test client instance, resolve these issues and add the tasks to resolve these issues to your migration plan.
7. Perform post-migration tasks that apply to your client.
8. Verify the migration was successful.
9. Test your applications, scripts, tools and maintenance procedures using the Version 9.5 client.

---

## Chapter 15. Migrating to Data Server Client (Windows)

Migrating an existing Version 8 DB2 Administration Client, Version 8 DB2 Application Development Client, or Version 9.1 DB2 Client copy to DB2 Version 9.5 requires that you install a Version 9.5 Data Server Client copy and then migrate your client instance to retain your client configuration and to connect to all your previously cataloged databases.

When you install a Version 9.5 Data Server Client, you can choose to automatically migrate an existing pre-V9.5 client copy. Your existing client instances are migrated to a new Version 9.5 Data Server Client copy and the existing pre-V9.5 client copy is removed. You can also choose to install a new copy of Version 9.5 Data Server Client and then manually migrate your existing client instance after installation.

### Prerequisites

- Ensure that you have SYSADM, SYSCTRL, or SYSMANT authority and Local Administrator authority to run the db2imigr and the db2icrt commands.
- Review supported connectivity between DB2 clients and DB2 servers in migration essentials for DB2 clients.
- Perform pre-migration tasks for DB2 clients.

### Restrictions

- The bit size of the client instance is determined by the operating system where you install a Version 9.5 client. The instance is 32-bit only in 32-bit Windows on x86 or X64. The instance is 64-bit only in 64-bit Windows on X64. Refer to Table 6 on page 25 for details.

### Procedure

To migrate from a Version 8 DB2 Administration Client, Version 8 DB2 Application Development Client, or Version 9.1 DB2 Client copy to a Version 9.5 Data Server Client on Windows:

1. Install Version 9.5 Data Server Client by running the setup.exe command to launch the DB2 Setup wizard. You have two options:
  - Select the option Work with Existing on the Install a Product panel. Then in the Work with an existing DB2 copy window, select the client copy name with action migrate. The selected DB2 copy is removed and your client instance is migrated. You can choose this option, if you have an existing copy of DB2 Version 8 Administration Client, DB2 Version 8 Application Development Client, or DB2 Version 9.1 Client.
  - Select the option Install New in the Install a Product panel. You should choose this option to create a new copy of Version 9.5 Data Server Client and keep your existing client copy. After installation, you must manually migrate the client instance to run on the Version 9.5 Data Server Client copy:
    - Log on to the system as a user with Local Administrator authority.
    - Run the db2imigr command:

```
"%DB2PATH%"\bin\db2imigr InstName
```

where DB2PATH is set to the location that you specified during the Version 9.5 Data Server Client installation and *InstName* is the name of the instance.

2. If you want your applications to use the Version 9.5 Data Server Client copy through the default interface or if you migrated your existing Version 8 client copy, set the Version 9.5 Data Server Client copy as the DB2 default copy. See “Changing the default DB2 and default IBM database client interface copy after installation” in *Quick Beginnings for DB2 Servers*.
3. Optional: You can create a new Version 9.5 client instance instead of migrating the existing client instance. You only need to create a new Version 9.5 client instance when you want to keep multiple client copies running on the same machine, or create a testing environment. To create a new Version 9.5 client instance, run the db2icrt command with the option -s:

```
"%DB2PATH%" \bin\db2icrt -s client InstName
```

To create the same client connectivity environment you had, including the database manager configuration parameter and DB2 profile registry settings, run the db2cfimp command with the configuration profile that you save in the pre-migration tasks.

4. Compare the migrated database manager configuration parameter values with the pre-migration values to ensure the changed values are compatible with your database applications.

After migrating your client, perform the recommended post-migration tasks for DB2 clients, especially verifying migration for clients to ensure that your client migration was successful.

---

## Chapter 16. Migrating to Data Server Runtime Client (Windows)

Migrating an existing Version 8 DB2 Run-Time Client, Version 8 DB2 Run-Time Client Lite, or Version 9.1 DB2 Runtime Client copy to DB2 Version 9.5 requires that you install a Version 9.5 Data Server Runtime Client copy and then migrate your client instance to retain your client configuration and to connect to all your previously cataloged databases

After you install a Version 9.5 Data Server Runtime Client copy, you can manually migrate your existing client instance from a Version 8 DB2 Run-Time, Version 8 DB2 Run-Time Client Lite copy, or a Version 9.1 DB2 Runtime Client copy.

### Prerequisites

- Ensure that you have SYSADM, SYSCTRL, or SYSMANT authority and Local Administrator authority to run the db2imigr and the db2icrt commands.
- Review supported connectivity between clients and DB2 servers in migration essentials for clients.
- Perform pre-migration tasks for clients.

### Restrictions

- The bit size of the client instance is determined by the operating systems where you install Version 9.5 client. The instance is 32-bit only in 32-bit Windows on x86 or X64. The instance is 64-bit only in 64-bit Windows on X64. Refer to Table 6 on page 25 for details.

### Procedure

To migrate from a Version 8 DB2 Run-Time, Version 8 DB2 Run-Time Client Lite, or a Version 9.1 DB2 Runtime Client copy to Version 9.5 Data Server Runtime Client on Windows:

1. Install Version 9.5 Data Server Runtime Client. See “Installing IBM data server clients (Windows)” in *Quick Beginnings for IBM Data Server Clients*. Run the setup.exe command to launch the DB2 Setup wizard.
2. If you want your applications to use the Version 9.5 Data Server Runtime Client copy through the default interface or if you migrated your existing Version 8 client copy, set the Version 9.5 Data Server Runtime Client copy as the DB2 default copy. See “Changing the default DB2 and default IBM database client interface copy after installation” in *Quick Beginnings for DB2 Servers*.
3. Log on to the system as a user with Local Administrator authority.
4. Migrate your existing client instance by running the db2imigr command:

```
"%DB2PATH%" \bin\db2imigr InstName
```

where DB2PATH is set to the location that you specified during the Version 9.5 Data Server Runtime Client installation and *InstName* is the name of the instance.

5. Optional: You can create a new Version 9.5 client instance instead of migrating an existing client instance. You only need to create a new Version 9.5 client instance when you want to keep multiple client copies running on the same machine. To create a new Version 9.5 client instance, run the db2icrt command with the option -s:

```
"%DB2PATH%"bin\db2icrt -s client InstName
```

To create the same client connectivity environment you had, including the database manager configuration parameter and DB2 profile registry settings, run the `db2cfimp` command with the configuration profile that you saved in the pre-migration tasks.

6. Compare the migrated database manager configuration parameter values with the pre-migration values to ensure the changed values are compatible with your database applications.

After migrating your client, perform the recommended post-migration tasks for clients, especially verifying migration for clients to ensure that your client migration was successful.

---

## Chapter 17. Migrating clients (Linux and UNIX)

Migrating Version 8 DB2 clients or Version 9.1 DB2 clients to DB2 Version 9.5 requires that you install a Version 9.5 client copy and then migrate your existing client instances to retain your client configuration and to connect to all your previously cataloged databases.

### Prerequisites

- Ensure that you have root access.
- Ensure that you have SYSADM, SYSCTRL, or SYSMANT authority and root access to run the db2imigr and the db2icrt commands.
- Ensure that you meet the installation requirements for DB2 database products. Some operating systems require a 64-bit kernel.
- Review supported connectivity between clients and DB2 servers in migration essentials for clients.
- Perform pre-migration tasks for clients.

### Restrictions

- You can only migrate from a Version 8 DB2 Administration Client, Version 8 DB2 Application Development Client, or Version 9.1 DB2 Client to a Version 9.5. Data Server Client.
- You can only migrate from a Version 8 DB2 Run-Time Client, Version 8 DB2 Run-Time Client Lite, or Version 9.1 DB2 Runtime Client to a Version 9.5 Data Server Runtime Client.
- On Linux and UNIX except for Linux on x64, your existing 32-bit or 64-bit client instances are migrated to Version 9.5 64-bit client instances. The bit size of the client instance is determined by the operating system where you install the Version 9.5 client. Refer to Table 6 on page 25 for details.

### Procedure

To migrate Version 8 DB2 clients or Version 9.1 DB2 clients to Version 9.5 clients:

1. Install the appropriate Version 9.5 client as a new copy by running the db2setup command and select Install New on the Install a Product panel:
  - If you are migrating from a Version 8 DB2 Administration Client, Version 8 DB2 Application Development Client, or Version 9.1 DB2 Client, install a new Version 9.5 Data Server Client.
  - If you are migrating from a Version 8 DB2 Run-Time Client, Version 8 DB2 Run-Time Client Lite, or Version 9.1 DB2 Runtime Client, install a new Version 9.5 Data Server Runtime Client copy.
2. Log on to the system as root.
3. Migrate your existing Version 9.1 or Version 8 client instances by running the db2imigr command:

```
$DB2DIR/instance/db2imigr InstName
```

where

#### DB2DIR

is set to the location that you specified during the Version 9.5 client installation. The default installation path for UNIX is /opt/IBM/db2/V9.5 and for Linux is /opt/ibm/db2/V9.5.

*InstName*

is the login name of the client instance owner.

4. Optional: You can also create a new Version 9.5 client instance instead of migrating the existing Version 9.1 or Version 8 client instance. You only need to create a new Version 9.5 client instance when you want to keep multiple client copies running on the same machine. To create a new Version 9.5 client instance, run the db2icrt command with the option -s:

```
$DB2DIR/instance/db2icrt -s client InstName
```

where

**DB2DIR**

is set to the location you specified during the Version 9.5 client installation.

*InstName*

Is the login name of the instance owner.

To create the same client connectivity environment you had, including the database manager configuration parameter and DB2 profile registry settings, run the db2cfimp command with the configuration profile that you backed up in the pre-migration tasks.

5. Compare the migrated database manager configuration parameter values with the pre-migration values to ensure the changed values are compatible with your database applications.

After migrating your client, perform the recommended post-migration tasks for clients, especially verifying migration for clients to ensure that your client migration was successful.

---

## Chapter 18. Post-migration tasks for clients

After migrating your clients, you should perform some post-migration tasks to ensure that your clients perform as expected and at their optimum level.

Perform the following post-migration tasks that apply to your clients:

1. Recatalog nodes and databases if you catalogued them using the NetBIOS and SNA protocols in DB2 UDB Version 8. Starting DB2 Version 9.1, the NetBIOS and SNA protocols are discontinued.
2. Manage changes in DB2 server behavior by modifying your settings where required. There are new registry variables, new configuration parameters, and new default values for registry variables and configuration parameters introduced in DB2 Version 9.5 that can impact the behavior of your application.
3. Verify that migration of your clients was successful.

---

### Recataloging nodes and databases using TCP/IP protocol

Recataloging nodes using the TCP/IP protocol is required when the communication protocol that you used to catalog these nodes is no longer supported or when you want to use a feature that supports only TCP/IP protocol. If you change the node name, you need to recatalog the databases as well to use the new node name.

The NetBIOS and SNA protocols are discontinued since DB2 Version 9.1. You need to recatalog, using a valid protocol, any nodes that you cataloged with the NetBIOS and SNA protocols. If you try to connect to any databases cataloged on a node that uses the NetBIOS or SNA protocol, your connection request returns an error because these protocols are invalid.

If you have a Version 8 DB2 client installed on the same system as a DB2 Version 9.5 server or a Version 9.5 DB2 client installed on the same system as a DB2 Version 8 server, connections to the databases on the DB2 server from the DB2 client cataloged using a local node are not supported. If you do not migrate the Version 8 DB2 client or DB2 Version 8 server to DB2 Version 9.5, you need to recatalog local nodes as TCP/IP nodes.

If you want to use the trusted context capability on migrated databases that are cataloged using a local node, you need to recatalog the nodes using the TCP/IP protocol.

#### Prerequisites

- Ensure that you have SYSADM or SYSCTRL authority.
- Ensure that you have network connectivity from the client to the DB2 server.

#### Restriction

The only protocols available in DB2 Version 9.5 are TCP/IP, Named Pipes, and SSL.

#### Procedure

To recatalog nodes and databases specifying the TCP/IP protocol:

1. Determine which are the local nodes that you want to recatalog or the nodes use the NetBIOS or SNA protocol by issuing the LIST NODE DIRECTORY command:

```
db2 LIST NODE DIRECTORY show detail > node_list.log
```

Redirect the output of this command to a file and keep it, because the information is useful to recatalog your nodes.

2. Remove the local nodes that you want to recatalog and all nodes that use NetBIOS or SNA protocol from the node directory by issuing the UNCATALOG NODE command:

```
db2 UNCATALOG NODE node-name
```

3. Determine which databases use the nodes that you uncataloged in the previous step by issuing the LIST DATABASE DIRECTORY command:

```
db2 LIST DATABASE DIRECTORY show detail > database_list.log
```

4. If you are going recatalog your nodes using a different node name, remove all databases using those nodes by issuing the UNCATALOG DATABASE command:

```
db2 UNCATALOG DATABASE database-name
```

5. Recatalog your nodes specifying TCP/IP as the protocol by issuing the CATALOG TCPIP NODE command. If you use the original node name, you do not need to recatalog your databases.

```
db2 CATALOG TCPIP NODE new-node REMOTE host-name
SERVER instance-svcename REMOTE_INSTANCE instance-name
```

You can determine the value of *instance-svcename* by looking at the value of the **svcename** database manager configuration parameter for that instance.

6. If you did not recatalog your nodes using the original node names, recatalog your databases using the new node name by issuing the CATALOG DATABASE command.

```
db2 CATALOG DATABASE db-name [AS alias-db-name]
AT NODE new-node
```

---

## Verifying migration of clients

When the migration of your client is complete, it is a good practice to run some tests in the new migrated environment to verify that your client is working as expected. These tests can consist of running batch programs that connect to databases in a DB2 server or any programs or scripts that you use for benchmarking.

### Prerequisites

- Ensure that you have network connectivity from the client to the DB2 server.
- Ensure that the DB2 servers and instances are up and running.

### Procedure

To verify that your client migration is successful:

1. Test connecting to all cataloged databases. The following example tests a connection to a remote database by issuing the CONNECT command:

```
db2 CONNECT TO sample USER mickey USING mouse
```

```
Database Connection Information
```

Database server = DB2/AIX64 9.5.0  
SQL authorization ID = MICKEY  
Local database alias = SAMPLE

You need to specify a user ID and password when connecting to a remote database.

2. If you experience problems connecting to your cataloged database, use the db2cfimp tool and the configuration profile that you saved by performing the saving DB2 clients configuration pre-migration task to re-create the same client connectivity environment you had prior to migration.
3. Run your client database applications or scripts that connect to your databases to ensure they are working as expected.



---

## Part 4. Migrating applications and routines

This part of the book contains the following chapters:

- Chapter 19, “Migration for database applications and routines,” on page 115
- Chapter 20, “Migration essentials for database applications,” on page 117
- Chapter 21, “Migration essentials for routines,” on page 135
- Chapter 22, “Pre-migration tasks for database applications and routines,” on page 137
- Chapter 23, “Migrating database applications,” on page 139
- Chapter 24, “Migrating routines,” on page 149
- Chapter 25, “Post-migration tasks for database applications and routines,” on page 157
- Chapter 26, “Enabling new DB2 Version 9.5 functionality in database applications and routines,” on page 159



---

## Chapter 19. Migration for database applications and routines

Upgrading to DB2 Version 9.5 involves migrating your database applications and routines if changes in DB2 Version 9.5 impact your database applications and routines.

Migrating your applications and routines involves the following actions:

- Test whether your applications and routines perform as expected in a DB2 Version 9.5 testing environment. You do not need to migrate your applications and routines if they run successfully.
- If your applications or routines have errors running in DB2 Version 9.5, you should:
  - Review migration essentials for database applications to identify any changes in DB2 Version 9.5 that can impact your applications.
  - Review migration essentials for routines to identify any changes in DB2 Version 9.5 that can impact your routines.
  - Plan how to modify your applications and routines to handle these changes. Determine the steps you need to perform by reviewing the Migrating database applications or Migrating routines tasks.
  - Modify your applications and routines according to your plan.
  - Test your applications and routines in your DB2 Version 9.5 testing environment.
- Verify that your applications and routines perform as expected in your DB2 Version 9.5 production environment before deploying them.

If your applications and routines use any functionality that is deprecated in DB2 Version 9.5, you should plan how to remove this functionality from your application code in the near future.

Also, you should consider using new features available in DB2 Version 9.5 to enhance functionality and improve performance.



---

## Chapter 20. Migration essentials for database applications

Changes in application development support, new features, discontinued features, and deprecated features might impact your database applications, scripts and tools.

### Operating system support

A complete list of supported operating systems is available at “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers*. If your current version of operating system is unsupported, you must upgrade it before you install DB2 Version 9.5.

In UNIX operating systems, only 64-bit kernels are supported. Your 32-bit instances are migrated to DB2 Version 9.5 64-bit instances.

If you upgrade to the latest version of your operating system or you install a 64-bit kernel, rebuild all database applications and external routines after you migrate to DB2 Version 9.5 so that they use the new runtime libraries in the operating system.

### Application drivers

The Version 9.5 clients have different names than in previous versions. The following list specifies the application drivers available for installation in each client:

- IBM Data Server Driver for ODBC, CLI, and .NET installs the .NET Data Provider and the drivers for ODBC and CLI.
- IBM Data Server Runtime Client has the same functionality as the previous versions of the Runtime Clients.
- IBM Data Server Client can install all application development drivers, depending on your selections during installation.
- IBM Data Server Driver for ODBC and CLI installs just the drivers for ODBC and CLI.
- IBM Data Server Driver for JDBC and SQLJ installs just this driver. The driver is included in all DB2 database products. Information about the versions of the IBM Data Server Driver for JDBC and SQLJ that are included by DB2 database product is available at “Java software support for DB2 products” in *Quick Beginnings for DB2 Servers*.

The IBM Data Server Driver for JDBC and SQLJ includes the `db2jcc.jar` class file for applications that use JDBC 3.0 methods or earlier and the `db2jcc4.jar` class file for applications that use JDBC 4.0 methods or earlier. The JDBC 4.0 `java.sql.DatabaseMetaData.getDriverName` method returns the IBM Data Server Driver for JDBC and SQLJ name instead of the IBM DB2 JDBC Universal Driver Architecture name. To manage the behavioral differences between the IBM Data Server Driver for JDBC and SQLJ and previous releases of this driver, migrate Java applications that use IBM Data Server Driver for JDBC and SQLJ..

The DB2 JDBC Type 2 driver has been deprecated since DB2 Version 9.1. You should modify your Java applications and external routines to use the IBM Data Server Driver for JDBC and SQLJ with type 2 connections. To manage the behavioral differences between the IBM Data Server Driver for JDBC and SQLJ Version 3.5 and the DB2 JDBC Type 2 driver, migrate your Java applications that use DB2 JDBC Type 2 driver.

## LOB columns and cursor blocking

To enable cursor blocking for LOB columns:

- For embedded SQL applications, use the BIND command with the BLOCKING ALL or BLOCKING UNAMBIGUOUS clause.
- For CLI applications, set the **BlockLobs** CLI configuration keyword to 1.

As part of the changes to improve the retrieval time for cursors involving LOB columns, the DB2 server now consumes more shared memory to store the references to the LOB values in each block of data. Memory usage for blocking cursors with LOB columns varies between 1 KB and 100 KB more than the memory usage for non-blocking cursors. In migrated databases, the **instance\_memory** database manager configuration parameter is set to AUTOMATIC to account for the increase in private and shared memory and other changes to the DB2 memory model. “Migrating embedded SQL applications” on page 140 provides details about how to manage these changes.

Setting the **BlockLobs** CLI configuration keyword to 1 enables a CLI application to receive, in a single fetch request, all of the LOB values immediately after the row data is received if the DB2 server supports LOB blocking. The DB2 CLI client layer buffers the other rows, including LOB values, received for previous FETCH requests. If you use this setting and you bind the LOB values directly to the buffers, your client requires more memory than in previous releases.

## Development software support

Development software support has also changed. To improve performance and avoid technical support issues, rebuild your applications with the latest version of your development software. Review the changes in support for development software for a specific list of what is no longer supported.

## DB2 APIs and DB2 commands

Review the following topics to determine if you have applications and scripts that are impacted by changes to DB2 APIs and DB2 commands in DB2 Version 9.5:

- DB2 API functions
- DB2 command line processor (CLP) and system commands

## SQL statements

Review the changes to SQL statements in DB2 Version 9.5 to determine if you have applications and scripts that are impacted by these changes.

The new DECFLOAT data type introduces the INF, INFINITY, NaN, and sNaN literals. If your applications use these literals as object names, you must delimit these object names using double quotes. The following example shows how to delimit the literal INFINITY:

```
SELECT A."INFINITY" FROM TABLEA A
```

If your applications query target tables for write-to-table event monitors, review the “Target tables changes for write-to-table event monitors” topic in *What’s New* to determine whether your applications are impacted by these changes and how to recreate your target tables for DB2 Version 9.5. You need to modify your applications to manage the changes in column data types and column length.

## System catalog views and system-defined administrative routines and views

After database migration to DB2 Version 9.5, the system catalog views under the SYSCAT schema remain compatible with catalog views that you defined in DB2 Version 9.1. However, there are new columns, increases in column length, or columns with changed data types in some of the system catalog views.

SQL administrative routines include changes such as new parameters and new columns returned. Also, some routines are replaced with system-defined administrative routines and views. In addition, all of the system-defined table functions with names that start with SNAPSHOT\_ have been deprecated since DB2 Version 9.1.

Review the following topics to determine if you have applications and scripts that are impacted by changes to system catalog views and system-defined administrative routines and views:

- System catalog
- “Deprecated system-defined administrative routines and their replacement routines or views” in *Administrative Routines and Views*

### Optimizer and query execution plans

Now, the optimizer evaluates the cost of more alternatives for queries with IN list predicates than in previous releases. The result can be a more optimal query execution plan that should improve query performance. You can use the EXPLAIN facility to find out details about the query execution plan.

The optimizer generates more optimal query execution plans for the following queries:

- Queries that combine the FETCH FIRST N ROWS ONLY and ORDER BY clauses in the same subselect, with the exception of any subselect that is an outermost full select. For any outermost full select with FETCH FIRST N ROWS ONLY, you can use the FETCH FIRST N ROWS ONLY with the OPTIMIZE FOR N ROWS clauses to obtain a similar result.
- Aggregate queries involving MAX or MIN aggregate functions with GROUP BY columns, but only for nested loop joins. There is no support for full outer joins.

The optimizer might choose a different query execution plan than in previous releases to avoid risky plans with potentially poor performance. This improvement should result in increased performance stability as long as the statistics are up to date.

Rebind any statically bound packages after migration to take advantage of this optimizer improvement.

### Database packages

When you migrate a database, all packages for user applications and routines are placed into an invalid state. Packages are also placed into an invalid state if they depend on database objects that you dropped, such as tables, views, aliases, indexes, triggers, referential constraints, and table check constraints. If you drop a UDF, your package is placed into an inoperative state.

Although invalid packages are automatically rebound by the database manager the first time that an application needs to access them, rebind your database packages to control when rebinding occurs and resolve any possible issues. See the Optimizer enhancements section for additional advantages of manually rebinding your database packages.

### 32-bit and 64-bit DB2 server support

On Linux and UNIX operating systems excluding Linux on x86, DB2 Version 9.5 requires 64-bit kernels and supports only 64-bit instances. Therefore, when you migrate to DB2 Version 9.5, your DB2 UDB Version 8 32-bit instances are migrated to 64-bit instances. See 32-bit and 64-bit DB2 server support changes for details.

The following table indicates applications that run after you migrate to DB2 Version 9.5 depending on the embedded shared library path:

Table 21. Embedded shared library paths used in database applications

| Application | Operating system | Migrating from  | Embedded shared library paths                                 |
|-------------|------------------|-----------------|---------------------------------------------------------------|
| 32-bit      | 32-bit or 64-bit | 32-bit instance | \$INSTHOME/sqllib/lib <sup>1</sup><br>\$INSTHOME/sqllib/lib32 |
| 64-bit      | 64-bit           | 64-bit instance | \$INSTHOME/sqllib/lib <sup>2</sup><br>\$INSTHOME/sqllib/lib64 |

**Note:**

1. \$INSTHOME/sqllib/lib is a symbolic link to \$INSTHOME/sqllib/lib32.
2. \$INSTHOME/sqllib/lib is a symbolic link to \$INSTHOME/sqllib/lib64.

where *INSTHOME* is your instance home directory.

During DB2 Version 9.5 installation, statements are added to the db2profile and db2cshrc file to set the environment variables for the library search path. These environment variables specify additional locations where DB2 shared libraries can be loaded at application run time, allowing your application to run after you migrate to DB2 Version 9.5 if you did not specify the correct shared library path. The following table shows the settings that you should have for the library search path environment variables:

Table 22. Environment variable settings for library search paths

| Environment variable and Operating system                                                                                                                   | Application                                     | Shared library path                |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------|------------------------------------|
| <ul style="list-style-type: none"> <li>• LIBPATH (AIX operating system)</li> <li>• LD_LIBRARY_PATH (HP-UX, Linux, and Solaris operating systems)</li> </ul> | 32-bit                                          | INSTHOME/sqllib/lib32 <sup>1</sup> |
| <ul style="list-style-type: none"> <li>• LIBPATH (AIX operating system)</li> <li>• LD_LIBRARY_PATH (HP-UX, Linux, and Solaris operating systems)</li> </ul> | 64-bit                                          | INSTHOME/sqllib/lib64              |
| LIB (Windows operating systems)                                                                                                                             | 32-bit application running on a 64-bit instance | DB2PATH\lib\Win32 <sup>2</sup>     |
| LIB (Windows operating systems)                                                                                                                             | 32-bit or 64-bit                                | DB2PATH\lib                        |

**Note:**

1. *INSTHOME* is your instance home directory. You should indicate this setting at the beginning of the LIBPATH variable.
2. *DB2PATH* is the directory of your DB2 Version 9.5 copy.

These environment variables specify additional locations where DB2 shared libraries can be loaded at application runtime, allowing your application to run after you migrate to DB2 Version 9.5 if you did not specify the correct shared library path.

### **31-bit database applications (Linux on zSeries)**

All migration considerations for 32-bit database applications also apply to 31-bit database applications running on Linux on zSeries.

### **Unicode support**

Databases that you create in DB2 Version 9.5 without specifying a code page use Unicode as the default code page. If you issue the CREATE DATABASE command in your existing applications, specify the code set and territory to indicate a specific code page. Doing so also prevents your applications from failing due to data expansion. Characters in Unicode have a variable length from 1 byte to 4 bytes.

If you plan to access Unicode databases in existing applications, ensure that you allocate enough space for your variables to store the Unicode characters.

### **DB2 server behavior**

In general, the DB2 server behavior is compatible between releases. However, there are changes in behavior to support new features or improve the performance of existing features. Review “DB2 server behavior changes” on page 26 to determine the impact of these behavior changes on your applications.

After migrating your DB2 server, compare your registry variable and configuration parameter values to your values before migration, and change any values according to the needs of your applications.

### **Client connectivity support**

Your applications can use Version 9.1 or Version 8 DB2 clients to access databases in DB2 Version 9.5 servers. However, only DB2 Version 9.1 or DB2 UDB Version 8 functionality is available to your applications. Review Chapter 13, “Migration essentials for clients,” on page 97 to learn details about client connectivity and to identify changes in support that can impact your DB2 clients.

### **DB2 Embedded Application Server (EAS)**

After migrating to DB2 Version 9.5, you need to perform the following tasks to run the DB2WebServices application in Version 9.5:

- . See “Installing the DB2 Embedded Application Server” in *Quick Beginnings for DB2 Servers* for details.
- . See “Deploying the DB2WebServices application” in *Quick Beginnings for DB2 Servers* for details.

The DB2 Web Tools are discontinued in DB2 Version 9.5. If you installed DB2 EAS in previous releases as a requirement to run these tools, you do not need to install DB2 EAS after migrating to DB2 Version 9.5. Alternatively, install the IBM Data Server Client and use the DB2 administration tools.

### **Migration of applications from DB2 UDB Version 8**

If you are migrating from DB2 UDB Version 8, review Migration essentials for applications in DB2 Version 9.1, which describes changes in application

driver support, 32-bit and 64-bit DB2 server support, and discontinued features that might also impact your applications and scripts.

## Changed APIs and data structures

Table 23. Back-level supported APIs and data structures

| API or Data Structure (Version) | Descriptive Name                                                    | New API or Data Structure (Version)                    |
|---------------------------------|---------------------------------------------------------------------|--------------------------------------------------------|
| sqlbftsq (V2)                   | Fetch Table Space Query                                             | sqlbftpq (V5)                                          |
| sqlbstsq (V2)                   | Single Table Space Query                                            | sqlbstpq (V5)                                          |
| sqlbtsq (V2)                    | Table Space Query                                                   | sqlbmtsq (V5)                                          |
| sqlectdd (V2)                   | Catalog Database                                                    | sqlecadb (V5)                                          |
| sqledosd (V8.1)                 | Open Database Directory Scan                                        | db2DbDirOpenScan (V8.2)                                |
| sqledgne (V8.1)                 | Get Next Database Directory Entry                                   | db2DbDirGetNextEntry (V8.2)                            |
| sqledcls (V8.1)                 | Close Database Directory Scan                                       | db2DbDirCloseScan (V8.2)                               |
| sqlpstart (V5)                  | Start Database Manager                                              | db2InstanceStart (V8)                                  |
| sqlpstop (V5)                   | Stop Database Manager                                               | db2InstanceStop (V8)                                   |
| sqlpstr (V2)                    | Start Database Manager (DB2 Parallel Edition Version 1.2)           | db2InstanceStart (V8)                                  |
| sqlstar (V2)                    | Start Database Manager (DB2 Version 2)                              | db2InstanceStart (V8)                                  |
| sqlstop (V2)                    | Stop Database Manager                                               | db2InstanceStop (V8)                                   |
| sqlerstd (V5)                   | Restart Database                                                    | db2DatabaseRestart (V6)                                |
| sqlfddb (V7)                    | Get Database Configuration Defaults                                 | db2CfgGet (V8)                                         |
| sqlfdsys (V7)                   | Get Database Manager Configuration Defaults                         | db2CfgGet (V8)                                         |
| sqlfrdb (V7)                    | Reset Database Configuration                                        | db2CfgSet (V8)                                         |
| sqlfrsys (V7)                   | Reset Database Manager Configuration                                | db2CfgSet (V8)                                         |
| sqlfudb (V7)                    | Update Database Configuration                                       | db2CfgSet (V8)                                         |
| sqlfusys (V7)                   | Update Database Manager Configuration                               | db2CfgSet (V8)                                         |
| sqlfxdb (V7)                    | Get Database Configuration                                          | db2CfgGet (V8)                                         |
| sqlfxsys (V7)                   | Get Database Configuration                                          | db2CfgGet (V8)                                         |
| sqlmon (V6)                     | Get/Update Monitor Switches                                         | db2MonitorSwitches (V7)                                |
| sqlmonss (V5)                   | Get Snapshot                                                        | db2GetSnapshot (V6)                                    |
| sqlmonsz (V6)                   | Estimate Size Required for sqlmonss() Output Buffer                 | db2GetSnapshotSize (V7)                                |
| sqlmrset (V6)                   | Reset Monitor                                                       | db2ResetMonitor (V7)                                   |
| sqluadav (V8)                   | Get Authorizations                                                  | AUTH_LIST_AUTHORITIES_FOR_AUTHID table function (V9.5) |
| sqlubkp (V5)                    | Backup Database                                                     | db2Backup (V8)                                         |
| sqlubkup (V2)                   | Backup Database                                                     | db2Backup (V8)                                         |
| sqluexpr                        | Export                                                              | db2Export (V8)                                         |
| sqlugrpi (V2)                   | Get Row Partitioning Information (DB2 Parallel Edition Version 1.x) | sqlugrpn (V5)                                          |

Table 23. Back-level supported APIs and data structures (continued)

| API or Data Structure (Version) | Descriptive Name                                                         | New API or Data Structure (Version) |
|---------------------------------|--------------------------------------------------------------------------|-------------------------------------|
| sqluhcls (V5)                   | Close Recovery History File Scan                                         | db2HistoryCloseScan (V6)            |
| sqluhget (V5)                   | Retrieve DDL Information From the History File                           | db2HistoryGetEntry (V6)             |
| sqluhgne (V5)                   | Get Next Recovery History File Entry                                     | db2HistoryGetEntry (V6)             |
| sqluhops (V5)                   | Open Recovery History File Scan                                          | db2HistoryOpenScan (V6)             |
| sqluhprn (V5)                   | Prune Recovery History File                                              | db2Prune (V6)                       |
| sqluhupd (V5)                   | Update Recovery History File                                             | db2HistoryUpdate (V6)               |
| sqluimpr                        | Import                                                                   | db2Import (V8)                      |
| sqluload (V7)                   | Load                                                                     | db2Load (V8)                        |
| sqluqry (V5)                    | Load Query                                                               | db2LoadQuery (V6)                   |
| sqlureot (V7)                   | Reorganize Table                                                         | db2Reorg (V8)                       |
| sqlurestore (V7)                | Restore Database                                                         | db2Restore (V8)                     |
| sqlurlog (V7)                   | Asynchronous Read Log                                                    | db2ReadLog (V8)                     |
| sqluroll (V7)                   | Rollforward Database                                                     | db2Rollforward (V8)                 |
| sqlursto (V2)                   | Restore Database                                                         | sqlurst (V5)                        |
| sqlustat (V7)                   | Runstats                                                                 | db2Runstats (V8)                    |
| sqlxhcom (V2)                   | Commit an Indoubt Transaction                                            | sqlxphcm (V5)                       |
| sqlxhqry (V2)                   | List Indoubt Transactions                                                | sqlxphqr (V5)                       |
| sqlxhrol (V2)                   | Roll Back an Indoubt Transaction                                         | sqlxphrl (V5)                       |
| SQL-AUTHORIZATIONS (V8)         | Authorizations Structure                                                 | none                                |
| SQLB-TBSQRY-DATA (V2)           | Table space data structure.                                              | SQLB-TBSPQRY-DATA (V5)              |
| SQLE-START-OPTIONS (V7)         | Start Database Manager data structure                                    | db2StartOptionsStruct (V8)          |
| SQLEDBSTOPOPT (V7)              | Start Database Manager data structure                                    | db2StopOptionsStruct (V8)           |
| SQLEDBSTRTOPT (V2)              | Start Database Manager data structure (DB2 Parallel Edition Version 1.2) | db2StartOptionsStruct (V8)          |
| SQLEDINFO (v8.1)                | Get Next Database Directory Entry data structure                         | db2DbDirInfo (V8.2)                 |
| SQLUEXPT-OUT                    | Export output structure                                                  | db2ExportOut (V8.2)                 |
| SQLUHINFO and SQLUHADM (V5)     | History file data structures                                             | db2HistData (V6)                    |
| SQLUIMPT-IN                     | Import input structure                                                   | db2ImportIn (V8.2)                  |
| SQLUIMPT-OUT                    | Import output structure                                                  | db2ImportOut (V8.2)                 |
| SQLULOAD-IN (V7)                | Load input structure                                                     | db2LoadIn (V8)                      |
| SQLULOAD-OUT (V7)               | Load output structure                                                    | db2LoadOut (V8)                     |
| db2DbDirInfo (V8.2)             | Get Next Database Directory Entry data structure                         | db2DbDirInfoV9 (V9.1)               |
| db2DbDirNextEntryStruct (V8.2)  | Get Next Database Directory Entry data structure                         | db2DbDirNextEntryStructV9 (V9.1)    |

Table 23. Back-level supported APIs and data structures (continued)

| API or Data Structure (Version) | Descriptive Name                                 | New API or Data Structure (Version) |
|---------------------------------|--------------------------------------------------|-------------------------------------|
| db2gDbDirNextEntryStruct (V8.2) | Get Next Database Directory Entry data structure | db2gDbDirNextEntryStrV9 (V9.1)      |

Table 24. Back-level unsupported APIs and data structures

| Name              | Descriptive Name                                        | API or data structure supported in V9 |
|-------------------|---------------------------------------------------------|---------------------------------------|
| sqlufrol/sqlgfrol | Roll Forward Database (DB2 Version 1.1)                 | db2Rollforward                        |
| sqluprfw          | Rollforward Database (DB2 Parallel Edition Version 1.x) | db2Rollforward                        |
| sqlurfwd/sqlgrfwd | Roll Forward Database (DB2 Version 1.2)                 | db2Rollforward                        |
| sqlurllf/sqlgrfwd | Rollforward Database (DB2 Version 2)                    | db2Rollforward                        |
| sqlxphqr          | List an Indoubt Transaction                             | db2XaListIndTrans                     |
| SQLXA-RECOVER     | Transaction API structure                               | db2XaRecoverStruct                    |

## Migration impact of DB2 command changes

The changes in DB2 Version 9.5 to DB2 command line processor (CLP) and system commands can impact your existing applications and scripts after you migrate to DB2 Version 9.5.

The changes to commands include new parameters, modifications to existing parameters, deprecated or discontinued parameters, and modifications to command output. The following table lists the changes that impact applications and scripts:

Table 25. Changes to DB2 CLP commands and system commands

| Command  | Summary of changes                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| db2audit | The syntax of the <b>configure</b> and <b>extract</b> parameters is now different. The output of the <b>describe</b> parameter has changed. The <b>prune</b> parameter is discontinued.                                                                                                                                                                                                                                                                                                                               |
| db2ckmig | This command now fails if the database is in restore pending state. If you have external unfenced routines on Linux and UNIX operating systems that have no dependency on the DB2 engine library in your database, this command returns the warning message SQL1349W and generates a file with a list of all the external unfenced routines that will be altered as FENCED and NOT THREADSAFE when you migrate the database. Refer to “Verifying that your databases are ready for migration” on page 40 for details. |
| db2cos   | On Linux and UNIX operating systems, the database manager now checks first if the INSTHOME/sqllib/adm/db2cos script exists, where INSTHOME is the instance home directory, and runs this script. If this script does not exist, the database manager runs the INSTHOME/sqllib/bin/db2cos default script. You can customize the INSTHOME/sqllib/adm/db2cos script, do not modify the default script. The same behavior applies to the scripts db2cos_datacorruption, db2cos_hang, and db2cos_trap.                     |

Table 25. Changes to DB2 CLP commands and system commands (continued)

| Command               | Summary of changes                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| db2icrt               | <p>When you create an instance on Windows operating systems, the default locations for user data files, such as instance directories and the db2cli.ini file, are under the following directories:</p> <ul style="list-style-type: none"> <li>• Documents and Settings\All Users\Application Data\IBM\DB2\copy name on the Windows XP and Windows 2003 operating systems</li> <li>• ProgramData\IBM\DB2\copy name on the Windows Vista operating system</li> </ul> <p>Ensure that your applications use the new default location for instances created in Version 9.5.</p> <p>On Linux and UNIX operating systems, the <b>-w</b> parameter is discontinued. The bit size for new instances is determined by the operating system where DB2 Version 9.5 is installed. Refer to “Support changes for 32-bit and 64-bit DB2 servers” on page 25 for details.</p> |
| db2iupdt and db2ilist | <p>On Linux and UNIX operating systems, the <b>-w</b> parameter is discontinued. The bit size for new instances is determined by the operating system where DB2 Version 9.5 is installed. Refer to “Support changes for 32-bit and 64-bit DB2 servers” on page 25 for details.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| db2licm               | <p>The <b>-n</b> parameter is discontinued. You do not need to indicate the number of processors that you are entitled to use because licenses are now measured on value units.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| db2look               | <p>The <b>-e</b> parameter now also generates DDL statements for new DB2 Version 9.5 features such as roles and audit policies.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| db2ls                 | <p>If a non-root instance owner runs this command without the <b>-q</b> parameter, the output displays information about all root installations and the non-root installation for this instance owner. For any other user, the output displays information only for the root installations.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| db2mtrk               | <p>The <b>-p</b> parameter (which lists private agent memory heaps) is deprecated and replaced with the <b>-a</b> parameter (which lists all application memory consumption). The command output is slightly different due to changes in the DB2 application memory model.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| db2pd                 | <p>The <b>-catalogcache</b> parameter now provides information about roles. The <b>-applications</b> parameter now provides information about the IP address, the data stream encryption, the system authorization ID, the connection trust type, and the name of the trusted context and role inherited by trusted connections.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| db2uiddl              | <p>This command has been deprecated since DB2 Version 9.1 and remains deprecated in DB2 Version 9.5. It will be removed in a future release. You should convert any unique indexes to DB2 Version 9.5 semantics before this command becomes deprecated. Refer to “Converting type-1 indexes to type-2 indexes in migrated databases” on page 82 for details.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| db2undgp              | <p>This command is discontinued. Refer to “Revoking the EXECUTE privilege on migrated routines from PUBLIC” on page 41 for details about how to run this command prior to migrating to DB2 Version 9.5.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

Table 25. Changes to DB2 CLP commands and system commands (continued)

| Command                           | Summary of changes                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| db2_deinstall and doce_deinstall  | If you are not running these commands from the DB2 copy installation, you must specify the <b>-b</b> parameter to indicate which DB2 copy installation path you want to uninstall. If you do not specify the <b>-b</b> parameter, the commands prompt you for the installation path.                                                                                                                                                                                                                         |
| db2_install                       | The <b>-b</b> parameter is required for a root installation if you specify the <b>-n</b> parameter. It is optional for a non-root installation and the only value that you can specify is the INSTHOME/sqllib directory, where INSTHOME is the instance home directory. If you do not specify the <b>-b</b> parameter, the default installation path is the INSTHOME/sqllib directory.                                                                                                                       |
| installFixPack                    | After applying the fix pack updates to a DB2 copy, this command now runs the db2iupdt or dasupdt commands automatically on the instances and DAS running under this DB2 copy. The <b>-b</b> parameter is optional for a non-root installation and the only value that you can specify is the INSTHOME/sqllib directory, where INSTHOME is the instance home directory.                                                                                                                                       |
| BACKUP DATABASE                   | This command can now perform single system view (SSV) backups on partitioned database environments, which means that you can back up all database partitions simultaneously. The <b>INCLUDE LOGS</b> parameter is the new default for online SSV backups and for backups of single partitioned databases. Use the <b>EXCLUDE LOGS</b> parameter to maintain the same behavior as in previous releases.                                                                                                       |
| CREATE DATABASE                   | The new default code page when you create databases without specifying a code page is Unicode (code set UTF-8).                                                                                                                                                                                                                                                                                                                                                                                              |
| GET AUTHORIZATIONS                | This command is deprecated and has been replaced by the AUTH_LIST_AUTHORITIES_FOR_AUTHID table function. Although this command reports the authorities directly granted or indirectly granted to the current user by any means, it does not report how the authorities were granted. You should use the table function instead of this command, the output of this table function returns all authorities granted to a specific authorization ID by a specific authorization ID type (Group, Role, or User). |
| DESCRIBE                          | The <b>TABLE</b> command parameter now returns information about implicitly hidden columns. The <b>OUTPUT</b> command parameter now returns information about an implicitly hidden column if you specify the column in the <b>SELECT</b> list of the described query.                                                                                                                                                                                                                                        |
| GET DB CFG and UPDATE DB CFG      | The GET DB CFG command output does not show deprecated parameters. The UPDATE DB CFG command using deprecated parameters has no effect.<br><br>In partitioned database environments, by default, the UPDATE DB CFG commands now apply changes to all database partitions. Refer to <b>DB2_UPDDBCFG_SINGLE_DBPARTITION</b> registry variable migration impact for details about how to revert to the same behavior as in previous releases.                                                                   |
| GET DBM CFG and UPDATE DBM CFG    | The GET DBM CFG command output does not show deprecated parameters. The UPDATE DBM CFG command using deprecated parameters has no effect.                                                                                                                                                                                                                                                                                                                                                                    |
| GET SNAPSHOT FOR ALL APPLICATIONS | The command output is slightly different due to changes in the DB2 application memory model.                                                                                                                                                                                                                                                                                                                                                                                                                 |

Table 25. Changes to DB2 CLP commands and system commands (continued)

| Command                               | Summary of changes                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IMPORT, and LOAD                      | Compression dictionaries are now automatically created as part of the table data population operations performed by the INSERT, IMPORT with mode INSERT, LOAD with mode INSERT, and REDISTRIBUTE DATABASE PARTITION GROUP commands.                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| LIST APPLICATIONS                     | The LIST APPLICATIONS command now shows only user applications unless the <b>SHOW DETAIL</b> parameter is indicated to show all applications including the system applications. In previous releases, the LIST APPLICATIONS command showed all applications whether the <b>SHOW DETAIL</b> parameter was indicated or not.                                                                                                                                                                                                                                                                                                                                                |
| REDISTRIBUTE DATABASE PARTITION GROUP | The first time that you run this command after migration, a new directory is created to log message files. This command logs new messages in files in the new directory using a new format and also in the original directory for previous releases using the old format. Message files that you had prior to migration are kept in their original location. The new directories are as follows: <ul style="list-style-type: none"> <li>• HOMEINST/sqllib/redistribute on Linux and UNIX operating systems, where HOMEINST is the instance home directory</li> <li>• DB2PATH\redistribute on Windows operating systems, where DB2PATH is the DB2 copy location</li> </ul> |
| REORG TABLE                           | You can still create a table compression dictionary using the REORG TABLE command with the <b>KEEPDICTIONARY</b> parameter. However, a dictionary is automatically created only if a sufficient volume of data exists in that table.                                                                                                                                                                                                                                                                                                                                                                                                                                      |

On the Windows Vista operating system, to perform administration tasks that require Local Administrator authority, you must run your scripts from a DB2 command prompt with full administrator privileges. Launch the **Command Window - Administrator** shortcut to get a DB2 command prompt with full administrator privileges. If extended security is enabled on the Windows Vista operating system, you also need to log on to the system with a user that is a member of the DB2ADMNS group to launch this shortcut.

## Migration impact of SQL statement changes

The changes to SQL statements in DB2 Version 9.5 can impact your existing applications and scripts after you migrate to DB2 Version 9.5.

The changes to SQL statements include new default behaviors and modifications to statement output. In addition, some statements are discontinued. The following table lists the changes that impact applications and scripts:

Table 26. Changes to SQL statements

| SQL statement                                                                                                  | Summary of changes                                                                                                                                                                    |
|----------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| “ALTER BUFFERPOOL” in <i>SQL Reference, Volume 2</i> and “CREATE BUFFERPOOL” in <i>SQL Reference, Volume 2</i> | The NOT EXTENDED STORAGE and EXTENDED STORAGE clauses are discontinued. The ALTER BUFFERPOOL and CREATE BUFFERPOOL statements return an error if you specify either of these clauses. |

Table 26. Changes to SQL statements (continued)

| SQL statement                                                                                        | Summary of changes                                                                                                                                                                                                                                                                                                                                                             |
|------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| “ALTER TABLE” in <i>SQL Reference, Volume 2</i> and “CREATE TABLE” in <i>SQL Reference, Volume 2</i> | When you issue these statements specifying COMPRESS YES, the table becomes eligible for an automatic build of the compression dictionary when the table has a sufficient volume of data.                                                                                                                                                                                       |
| “ALTER TABLESPACE” in <i>SQL Reference, Volume 2</i>                                                 | The REDUCE clause now supports automatic storage table spaces. Also, reducing the size of a container reduces the high water mark in certain cases for all types of table spaces. Refer to SQL Reference for details about the new REDUCE clause syntax required for automatic storage table spaces and information about how the changes in reducing the size of a container. |
| “INSERT” in <i>SQL Reference, Volume 2</i>                                                           | Compression dictionaries are now automatically created as part of the table data population operations performed by the INSERT, IMPORT with mode INSERT, LOAD with mode INSERT, and REDISTRIBUTE DATABASE PARTITION GROUP commands.                                                                                                                                            |
| “VALIDATED predicate” in <i>SQL Reference, Volume 2</i>                                              | The VALIDATED predicate now also checks for validation of the value specified by <i>xml-expression</i> , not just <i>column-name</i> . This expression must return a value with the XML data type. Other changes to the VALIDATED predicate include the ACCORDING TO XMLSCHEMA and IS NOT VALIDATED clause.                                                                    |

## Migration impact of system catalog changes

In DB2 Version 9.5, system catalog objects are modified to support new features. These changes can impact your existing applications and scripts after you migrate to DB2 Version 9.5.

### System catalog views

In general, modifications to the existing catalog views consist of new columns, changed column data types, or longer column lengths. The following table lists the system catalog view changes that impact applications and scripts in DB2 Version 9.5:

Table 27. Changes to system catalog views

| View name         | Summary of changes with migration impact                                                                                                                                                          |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SYSCAT.ATTRIBUTES | New columns COLLATIONSCHEMA and COLLATIONNAME are added. The DL_FEATURES column returns blank characters because the DATALINK data type is not supported.                                         |
| SYSCAT.CHECKS     | New columns COLLATIONSCHEMA, COLLATIONNAME, COLLATIONSCHEMA_ORDERBY, COLLATIONNAME_ORDERBY, and OWNERTYPE are added. The FUNC_PATH column data type is changed from VARCHAR (254) to CLOB (2 KB). |

Table 27. Changes to system catalog views (continued)

| View name                    | Summary of changes with migration impact                                                                                                                                                                                                               |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SYSCAT.COLUMNS               | New columns ROWCHANGESTAMP, COLLATIONSCHEMA, and COLLATIONNAME are added. The DL_FEATURES column returns null because the DATALINK data type is not supported. New value T is added to the IDENTITY column. New value I is added to the HIDDEN column. |
| SYSCAT.DATATYPES             | New column ARRAY_LENGTH of type INTEGER and columns COLLATIONSCHEMA, COLLATIONNAME, and OWNERTYPE are added. The CLASS column size is increased from VARCHAR (128) to VARCHAR (384).                                                                   |
| SYSCAT.EVENTMONITORS         | New column OWNERTYPE is added. TARGET column size is increased from VARCHAR (256) to VARCHAR (762).                                                                                                                                                    |
| SYSCAT.EVENTS                | TYPE column size is increased from VARCHAR (18) to VARCHAR (128). The FILTER column size is increased from CLOB (32 KB) to CLOB (64 KB).                                                                                                               |
| SYSCAT.EVENTTABLES           | LOGICAL_GROUP column size is increased from VARCHAR (18) to VARCHAR (128).                                                                                                                                                                             |
| SYSCAT.INDEXES               | New columns COLLECTSTATISTICS, OS_PTR_SIZE, and OWNERTYPE are added. IEARGUMENTS column size is increased from CLOB (32 KB) to CLOB (64 KB).                                                                                                           |
| SYSCAT.INDEXEXPLOITRULES     | SEARCHKEY column size is increased from VARCHAR (320) to VARCHAR (640). SEARCHARGUMENT column size is increased from VARCHAR (1800) to VARCHAR (2700).                                                                                                 |
| SYSCAT.INDEXEXTENSIONMETHODS | RANGEFUNCNAME and RANGESPECIFICNAME column sizes are increased from VARCHAR (18) to VARCHAR (128).                                                                                                                                                     |
| SYSCAT.INDEXEXTENSIONPARMS   | New columns COLLATIONSCHEMA and COLLATIONNAME are added.                                                                                                                                                                                               |
| SYSCAT.NICKNAMES             | The OWNER column replaces the DEFINER column. The DEFINER column is included only for compatibility with previous releases. New columns OWNERTYPE and REMOTE_TYPE are added.                                                                           |
| SYSCAT.PACKAGES              | New columns BOUNDBYTYPE, OWNERTYPE, COLLATIONSCHEMA, COLLATIONNAME, COLLATIONSCHEMA_ORDERBY, COLLATIONNAME_ORDERBY, OPTPROFILESCHEMA, and OPTPROFILENAME are added. FUNC_PATH column data type is changed from VARCHAR (254) to CLOB (2 KB).           |
| SYSCAT.PREDICATESPECS        | CONTEXTEXP column size is increased from CLOB (32 KB) to CLOB (2 MB).                                                                                                                                                                                  |

Table 27. Changes to system catalog views (continued)

| View name                                                                 | Summary of changes with migration impact                                                                                                                                                                                                                                                                                                                                                                                     |
|---------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SYSCAT.ROUTINES                                                           | New columns OWNERTYPE, COLLATIONSCHEMA, COLLATIONNAME, COLLATIONSCHEMA_ORDERBY, and COLLATIONNAME_ORDERBY are added. IMPLEMENTATION column size is increased from VARCHAR (256) to VARCHAR (762). JAR_SIGNATURE column size is increased from VARCHAR (1024) to VARCHAR (2048). CLASS column size is increased from VARCHAR (128) to VARCHAR (384). FUNC_PATH column data type is changed from VARCHAR (254) to CLOB (2 KB). |
| SYSCAT.SCHEMATA<br>SYSCAT.SEQUENCES                                       | New columns OWNERTYPE and DEFINERTYPE are added.                                                                                                                                                                                                                                                                                                                                                                             |
| SYSCAT.SECURITYPOLICIES                                                   | New columns ALTER_TIME, GROUPGRANTABLE, ROLEGRANTABLE, and USERGRANTABLE are added.                                                                                                                                                                                                                                                                                                                                          |
| SYSCAT.TABLES                                                             | New columns ALTER_TIME, COLLATIONSCHEMA, COLLATIONNAME, COLLATIONSCHEMA_ORDERBY, COLLATIONNAME_ORDERBY, and OWNERTYPE are added.                                                                                                                                                                                                                                                                                             |
| SYSCAT.TRIGGERS                                                           | New columns OWNERTYPE, COLLATIONSCHEMA, COLLATIONNAME, COLLATIONSCHEMA_ORDERBY, and COLLATIONNAME_ORDERBY are added. FUNC_PATH column data type is changed from VARCHAR (254) to CLOB (2 KB).                                                                                                                                                                                                                                |
| SYSCAT.USEROPTIONS                                                        | New column AUTHIDTYPE is added.                                                                                                                                                                                                                                                                                                                                                                                              |
| SYSCAT.VIEWS                                                              | New column OWNERTYPE is added. FUNC_PATH column data type is changed from VARCHAR (254) to CLOB (2 KB).                                                                                                                                                                                                                                                                                                                      |
| SYSCAT.PACKAGEDEP<br>SYSCAT.ROUTINEDEP<br>SYSCAT.TABDEP<br>SYSCAT.TRIGDEP | New values are added to the BTYPE and TABAUTH columns.                                                                                                                                                                                                                                                                                                                                                                       |

Table 27. Changes to system catalog views (continued)

| View name                | Summary of changes with migration impact |
|--------------------------|------------------------------------------|
| SYSCAT.DBPARTITIONGROUPS | New column OWNERTYPE is added.           |
| SYSCAT.FUNCMAPPINGS      |                                          |
| SYSCAT.INDEXEXTENSIONS   |                                          |
| SYSCAT.REFERENCES        |                                          |
| SYSCAT.ROUTINESFEDERATED |                                          |
| SYSCAT.TABCONST          |                                          |
| SYSCAT.TABLESPACES       |                                          |
| SYSCAT.TYEMAPPINGS       |                                          |
| SYSCAT.XSROBJECTS        |                                          |
| SYSCAT.COLAUTH           | New column GRANTORTYPE is added.         |
| SYSCAT.DBAUTH            |                                          |
| SYSCAT.INDEXAUTH         |                                          |
| SYSCAT.PACKAGEAUTH       |                                          |
| SYSCAT.PASSTHROUGHAUTH   |                                          |
| SYSCAT.ROUTINEAUTH       |                                          |
| SYSCAT.SCHEMAAUTH        |                                          |
| SYSCAT.SEQUENCEAUTH      |                                          |
| SYSCAT.TBAUTH            |                                          |
| SYSCAT.TBSPACEAUTH       |                                          |
| SYSCAT.XSROBJECTAUTH     |                                          |
| SYSCAT.DBPARTITIONGROUPS | New column REDIST_EXECINFO is added.     |

## System-defined built-in routines

Changes to system-defined built-in routines include new routines, new parameters and changes in behavior. The following table lists the new routines and changes to existing routines that impact applications and scripts in DB2 Version 9.5:

Table 28. Changes to system-defined built-in routines

| Routine name                                         | Summary of changes with migration impact                                                                                                                                                                                                                                                                                  |
|------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| COLLATION_KEY_BIT ,<br>RID_BIT, and RID<br>functions | If you have user-defined functions with the same name as these built-in functions and they are not fully qualified in your queries, the resolution path results in the new built-in functions being called. Fully qualify your user- defined function calls with the schema name to avoid using these built-in functions. |

Table 28. Changes to system-defined built-in routines (continued)

| Routine name                                                             | Summary of changes with migration impact                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| INSERT, LEFT, OVERLAY, RIGHT, STRIP, and TRIM                            | To accommodate the variable character size in Unicode, there are new character-sensitive versions of these functions with the SYSIBM schema and one additional parameter to indicate the string unit. If you use the default SQL path and do not specify the string unit, the new version of these functions is called and the behavior is compatible with the behavior in previous releases, but there are some differences. To call the same function available in previous releases, explicitly qualify the function name with the SYSFUN schema. |
| COMPARE_DECFLOAT, DECFLOAT, NORMALIZE_DECFLOAT, QUANTIZE, and TOTALORDER | If you have user-defined functions with the same names as these built-in functions and they are not fully qualified in your queries, the resolution path results in the new built-in functions being called. Fully qualify your user-defined function calls with the schema name to avoid using these built-in functions.                                                                                                                                                                                                                            |

## System-defined administrative routines and views

Changes to system-defined administrative routines and views include new columns returned and new routines and views. In addition, all of the administrative routines with names that start with SNAPSHOT have been deprecated since DB2 Version 9.1. The following table lists the administrative routine and view changes that impact applications and scripts in DB2 Version 9.5:

Table 29. Changes to system-defined administrative routines and views

| Routine or view name | Summary of changes with migration impact                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ADMIN_CMD            | If you call the ADMIN_CMD procedure to run the UPDATE DB CFG or RESET DB CFG command without specifying a partition number, the changes always apply to all database partitions, regardless of the <b>DB2_UPDDBCFG_SINGLE_DBPARTITION</b> registry variable setting. If you call the ADMIN_CMD procedure to run the UPDATE DB CFG command using the <b>maxagents</b> and <b>maxcagents</b> database manager configuration parameters, the return code indicates success, but the command has no effect because these parameters are deprecated. |
| ADMINTABINFO         | New column STATSTYPE is added.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| AUTHORIZATIONIDS     | Role support is added.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| ENV_PROD_INFO        | New columns INSTALLED_PROD_FULLNAME and LICENSE_TYPE are added. One existing column, IS_LICENSED, is changed to LICENSE_INSTALLED with data type of CHAR(1).                                                                                                                                                                                                                                                                                                                                                                                    |
| PRIVILEGES           | Authorization ID type R (role) is added to information returned in column AUTHIDTYPE.                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| SNAPAPPL             | New columns TOTAL_OLAP_FUNCS and OLAP_FUNC_OVERFLOWS are added.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| SNAPAPPL_INFO        | New columns WORKLOAD_ID INTEGER and IS_SYS_APPL are added. The format of information returned in AUTHORITY_LVL column is changed.                                                                                                                                                                                                                                                                                                                                                                                                               |
| SNAPBP               | The PHYSICAL_PAGE_MAPS column is removed. The <b>physical_page_maps</b> monitor element is discontinued.                                                                                                                                                                                                                                                                                                                                                                                                                                        |

Table 29. Changes to system-defined administrative routines and views (continued)

| Routine or view name | Summary of changes with migration impact                                                                                                                                                                                                                                                                                                                                             |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SNAPDB               | The following new columns are added: <ul style="list-style-type: none"> <li>• TOTAL_OLAP_FUNCS</li> <li>• OLAP_FUNC_OVERFLOWS</li> <li>• ACTIVE_OLAP_FUNCS</li> <li>• STATS_CACHE_SIZE</li> <li>• STATS_FABRICATIONS</li> <li>• SYNC_RUNSTATS</li> <li>• ASYNC_RUNSTATS</li> <li>• STATS_FABRICATE_TIME</li> <li>• SYNC_RUNSTATS_TIME</li> <li>• NUM_THRESHOLD_VIOLATIONS</li> </ul> |
| SNAPDBM              | New column POST_THRESHOLD_OLAP_FUNCS is added. The <b>agents_waiting_top</b> , <b>agents_waiting_on_token</b> , and <b>max_agent_overflows</b> monitor elements are deprecated. Therefore, the corresponding columns return a NULL value for each of these monitor elements in this routine and view.                                                                                |
| SNAPDYN_SQL          | New columns STATS_FABRICATION_TIME and SYNC_RUNSTATS_TIME are added.                                                                                                                                                                                                                                                                                                                 |
| SNAP_GET_TAB_REORG   | Some identifiers have been changed in the information returned for the REORG_TYPE column.                                                                                                                                                                                                                                                                                            |

Review the “Deprecated SQL administrative routines and their replacement routines or views” in *Administrative Routines and Views* to determine additional changes that might impact your applications and scripts.

## System catalog changes between Version 8 and Version 9.1

If you are migrating from DB2 UDB Version 8, system catalog changes between DB2 UDB Version 8 and DB2 Version 9.1 can also impact your applications and scripts. Review the migration essentials for applications topic in the DB2 Information Center Version 9.1 and the Version 9.1 incompatibilities with previous releases topic or details about the changes to system catalog views and system-defined routines.



---

## Chapter 21. Migration essentials for routines

Migration essentials describe changes in application development support, changes to support new features, unsupported features, and deprecated features that might impact your routines.

The changes described in Chapter 20, “Migration essentials for database applications,” on page 117 could also impact your routines.

### Development software support

The information about development software support in Chapter 20, “Migration essentials for database applications,” on page 117 applies to external stored procedures and user-defined functions (UDFs).

### Unfenced external routines

During database migration to DB2 Version 9.5 on Linux and UNIX operating systems, all external unfenced routines that have no dependency on the DB2 engine libraries (libdb2e.a or libdb2apie.a) are altered to FENCED and NOT THREADSAFE so you can safely run these routines under the new multithreaded database manager. Running external routines defined as NOT FENCED and THREADSAFE in the new multithreaded database manager that are not thread safe can yield incorrect results, database corruption, or abnormal termination of the database manager. Refer to “Migrating C, C++, and COBOL routines” on page 150 for details about how to manage this change.

### 32-bit external routines

The implementation for LOB locators depends on DB2 database product installed. Also, you can use LOB locators only in unfenced routines. If you migrate from a DB2 UDB Version 8 32-bit instance to a DB2 Version 9.5 64-bit instance, you must rebuild 32-bit external routines that use LOB locators as 64-bit unfenced routine libraries.

Support for default function entry points in external routine libraries is deprecated in DB2 Version 9.1. If you migrated from a DB2 UDB Version 8 32-bit instance on AIX or Windows operating systems, you should specify an explicit entry point for your routine library.

### 31-bit external routines (Linux on zSeries)

All migration considerations for 32-bit external routines also apply to 31-bit external routines running on a DB2 database on Linux on zSeries.

### SQL stored procedures

SQL stored procedures that you created in DB2 UDB Version 8.1 will run in DB2 Version 9.5 if you migrate from a DB2 UDB Version 8 32-bit instance to a DB2 Version 9.5 32-bit instance, provided that they do not reference any unsupported features. This also applies if you migrate from a DB2 UDB Version 8 64-bit instance to a DB2 Version 9.5 64-bit instance. However, if you migrate from a DB2 UDB Version 8.1 32-bit instance to a DB2 Version 9.5 64-bit instance, your SQL procedures do not run because the 64-bit DB2 engine cannot load the 32-bit libraries associated to these procedures. You must drop and re-create these SQL procedures.

If you created SQL stored procedures in DB2 UDB Version 8.2 or DB2 Version 9.1 and migrate your databases to DB2 Version 9.5, your SQL

stored procedures are migrated to the DB2 Version 9.5 executable code and will function successfully provided that they do not reference any unsupported features.

### Java external routines

Starting DB2 Version 9.5, the default JDBC driver to run JDBC routines is the IBM Data Server Driver for JDBC and SQLJ. See “Migrating Java routines” on page 152 for details on how to manage this change.

DB2 Version 9.5 installs a 32-bit JVM by default on Linux on x86 and Windows (when DB2 Version 9.5 32-bit product is installed) operating systems. For all other supported operating systems, DB2 Version 9.5 installs a 64-bit JVM.

If you migrate an instance to DB2 Version 9.5, the **jdk\_path** database manager configuration parameter is set to the following value:

*Table 30. jdk\_path database manager configuration parameter settings*

| DB2 Version 9.5 instance  | Operating System | jdk_path value              |
|---------------------------|------------------|-----------------------------|
| 32-bit instance           | Linux            | INSTHOME/sql/lib/java/jdk32 |
| 64-bit instance           | Linux and UNIX   | INSTHOME/sql/lib/java/jdk64 |
| 32-bit or 64-bit instance | Windows          | DB2PATH\java\jdk            |

In DB2 Version 9.5 64-bit instances, Java external routines require that the **jdk\_path** parameter is set to a 64-bit JVM installation path to run successfully. A DB2 Version 9.5 64-bit instance cannot load a 32-bit JVM.

### Migration of routines from DB2 UDB Version 8

If you are migrating from DB2 UDB Version 8, review the Migration essentials for routines in DB2 Version 9.1 that describe changes in application development support, changes to support new features, unsupported features, and deprecated features that might also impact your routines.

---

## Chapter 22. Pre-migration tasks for database applications and routines

Before you migrate your database applications and routines, you should perform certain tasks to help you ensure a successful migration.

Prepare for the migration of your database applications and routines by performing the following tasks:

1. Review migration essentials for database applications to determine which changes might impact your database applications.
2. Review migration essentials for routines to determine which changes might impact your routines.
3. Plan your migration strategy.
4. Upgrade your operating system to a supported level if necessary.
5. Upgrade your development software to a supported level if necessary.
6. Optional: Migrate your client or install a Version 9.5 application driver if your application requires one. Although DB2 Version 9.5 server provides connectivity support for earlier clients, migrating to a Version 9.5 client eliminates any limitations and incompatibilities between releases.
7. Test your database applications in a DB2 Version 9.5 testing environment. If testing is successful, you do not need to migrate your applications. However, review the migrating database applications task and consider performing any steps that can help you improve performance.
8. Test your routines in a DB2 Version 9.5 testing environment. If testing is successful, you do not need to migrate your routines. However, review the migrating routines task and consider performing any steps that can help you improve performance.



---

## Chapter 23. Migrating database applications

Migrating database applications that you created for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these applications by modifying your code and rebuilding your applications.

You only need to modify your application code to manage changes in DB2 Version 9.5 that impact your applications, to remove the use of deprecated or discontinued functionality in DB2 Version 9.5, or to use new features.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that you meet the installation requirements for DB2 database products.
- Ensure that the development software is at a version level that is supported by DB2 database products.
- Perform the pre-migration tasks for database applications.

### Restriction

This procedure only applies to database applications programmed in C, C++, COBOL, FORTRAN, Java, Perl, PHP, REXX, and .NET languages.

### Procedure

To migrate your database applications to DB2 Version 9.5:

1. If you identified changed DB2 commands, changed SQL statements, and changed system catalog views and built-in functions that impact your applications, edit your application code or scripts to modify:
  - DB2 CLP and system command syntax
  - SQL statements syntax
  - SQL statements using catalog views and SQL Administrative views and routines
  - SQL statements using target tables for write-to-table event monitors
  - User defined routine names that are not fully qualified with a schema name
  - DB2 API calls
  - Application programming interface calls such as JDBC, ODBC and CLI
  - If your applications or scripts read from the command output, modify them to read the changed output format.
2. If you identified changes specific to the development environment that impact your applications, modify them to support these changes. Migrate your:
  - Embedded SQL applications
  - CLI applications
  - Java applications that use the IBM Data Server Driver for JDBC and SQLJ or that use the DB2 JDBC Type 2 driver
  - ADO and .NET applications
  - Scripts that use DB2 CLP commands and SQL statements
  - 32-bit database applications to run on 64-bit instances

3. Rebuild all changed database applications programmed in C/C++, COBOL, FORTRAN, and REXX, using the appropriate DB2 build file and specifying the appropriate DB2 shared library path as shown in Table 21 on page 120.
4. Test your database applications to verify your changes and to ensure that they run as expected using DB2 Version 9.5.

After migrating your database applications, perform the recommended post-migration tasks for database applications to ensure that your migration was successful.

---

## Migrating embedded SQL applications

Migrating embedded SQL applications that you developed for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these applications.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that the C, C++, COBOL, FORTRAN, or REXX development software is at a version level that is supported by DB2 database products.
- Perform previous steps in the migrating database applications task.

### Restriction

This procedure only applies to database applications programmed in C, C++, COBOL, FORTRAN, and REXX.

### Procedure

To migrate your embedded SQL applications to DB2 Version 9.5:

1. If you modified the library path environment variables, ensure that those variables include the correct DB2 shared library path for your applications as shown in Table 22 on page 120. The environment variables listed in this table specify additional paths to enable your applications to find the appropriate DB2 shared library at runtime (in most cases).  
**On the Linux operating system:** if you link an application using the RPATH link option without also specifying the RUNPATH link option, the LD\_LIBRARY\_PATH environment variable will be ignored at application run time, which can cause your application to fail.
2. Test your embedded SQL applications in a DB2 Version 9.5 testing environment. If testing is successful, you do not need to perform any additional steps.
3. If you bound your embedded applications using the BIND command with the BLOCKING ALL or BLOCKING UNAMBIGUOUS clause to enable the blocking of cursors for LOB columns, ensure that the **instance\_memory** or **database\_memory** database configuration parameters are set to AUTOMATIC or increase their numeric value to account for the extra memory usage. If you cannot increase these database configuration parameters, you have the following options:
  - Rebind them using the BIND command specifying **BLOCKING NO** or precompile them using the PRECOMPILE command specifying the **SQLRULES STD** command parameter. The **BLOCKING NO** clause disables

blocking of all cursors in the application. The **SQLRULES STD** command parameter might have other effects than disabling blocking cursors.

- Modify the application source code and declare the cursor with the FOR UPDATE clause to disable blocking.
4. To explicitly specify the correct DB2 shared library path for your applications, do one of the following:
    - If the application source code is available, rebuild the application. Specify the required DB2 shared library path as shown in Table 21 on page 120. This is the best option.
    - Create a wrapper script to run your application. In the wrapper script, explicitly set the library path environment variable to the required DB2 shared library path as shown in Table 22 on page 120.
    - If you do not have the original source code available, run the `db2chglbpath` command to update the embedded runtime library path within the binary code of your application. This command is provided as-is and should therefore be considered a last resort.

After migrating your embedded SQL applications, perform the remaining steps in the migrating database applications task.

---

## Migrating CLI applications

Migrating CLI applications that you developed for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these applications, such as operating system support changes, development software support changes, the bit-width of the application, and the bit-width of the DB2 instance on which you deploy the applications.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that the C and C++ development software is a version that is supported by DB2 database products.
- Perform previous steps in the Chapter 23, “Migrating database applications,” on page 139 task.

### Restriction

This procedure only applies to database applications programmed in C or C++ using the CLI interface.

### Procedure

To migrate your CLI applications to DB2 Version 9.5:

1. If you modified the library path environment variables, ensure that those variables include the correct DB2 shared library path for your applications, as shown in Table 22 on page 120. You can use the environment variables listed in this table to specify additional paths that enable your applications to find the appropriate DB2 shared library at run time (in most cases).

**On Linux operating systems only:** If you link an application using the `RPATH` link option without also specifying the `RUNPATH` link option, the `LD_LIBRARY_PATH` environment variable is ignored at application run time, which can cause your application to fail.

2. If you have set the CLISchema configuration keyword in your db2cli.ini file, set the SysSchema configuration keyword instead. The CLISchema configuration keyword is discontinued.

SysSchema = alternative schema

3. Test your CLI applications in a DB2 Version 9.5 testing environment. If testing is successful, you do not need to perform the remaining steps.
4. If you set the **BlockLobs** CLI configuration keyword to 1 and your application gets the error message SQL0973N, perform one of the following actions:
  - Set the **database\_memory** configuration parameter to AUTOMATIC. This is the best option.
  - Reset the **BlockLobs** CLI configuration keyword to 0.
  - Bind LOB values directly to buffers instead of using LOB locators.

Your client requires more memory to receive LOB data because this cursor blocking setting using the **BlockLobs** keyword sends all the LOB values immediately to your client after the row data is sent.

5. Explicitly specify the correct DB2 shared library path for your applications by performing one of the following actions:
  - If the application source code is available, rebuild the applications. Specify the required DB2 shared library path as shown in Table 21 on page 120. This is the best option.
  - Create a wrapper script to run your applications. In the wrapper script, explicitly set the library path environment variable to the required DB2 shared library path as shown in Table 22 on page 120.
  - If you do not have the original source code available, run the db2chglbpath command to update the embedded runtime library path within the binary code of your applications. This command is provided as-is and should therefore be considered a last resort.

After migrating your CLI applications, perform the remaining steps in the Chapter 23, “Migrating database applications,” on page 139 task.

---

## Migrating Java applications that use IBM Data Server Driver for JDBC and SQLJ

Migrating Java applications that you built for DB2 Version 9.1 or DB2 UDB Version 8 that use previous releases of the IBM Data Server Driver for JDBC and SQLJ Version 4.0 or Version 3.50 involves managing the changes in DB2 Version 9.5 and changes between different releases of this driver that can impact these applications.

### Prerequisites

- Review the migration essentials for applications to identify key changes that might impact your Java database applications.
- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that the Java application development software and IBM Data Server Driver for JDBC and SQLJ are at a version level that is supported by DB2 database products.
- Perform the previous steps in the migrating database applications task.

### Restrictions

- The minimum supported Java SDK for the IBM Data Server Driver for JDBC and SQLJ Version 3.50 is Java SDK 1.4.2.
- The minimum supported Java SDK for the IBM Data Server Driver for JDBC and SQLJ Version 4.0 is Java SDK 6.
- This procedure applies only to Java applications using the IBM Data Server Driver for JDBC and SQLJ.

## Procedure

To migrate your Java database applications using the IBM Data Server Driver for JDBC and SQLJ to DB2 Version 9.5:

1. Install the IBM Data Server Driver for JDBC and SQLJ Version 4.0 or Version 3.50:
  - If you use methods in JDBC 4.0 or earlier specifications in your applications, install IBM Data Server Driver for JDBC and SQLJ Version 4.0.
  - If you use methods in JDBC 3.0 or earlier specifications in your applications, install IBM Data Server Driver for JDBC and SQLJ Version 3.50.
2. If you are migrating applications that use the IBM DB2 Driver for JDBC and SQLJ prior to Version 3.50, update your applications to manage the following differences between this driver and the IBM Data Server Driver for JDBC and SQLJ Version 4.0 or Version 3.50:
  - The IBM Data Server Driver for JDBC and SQLJ Version 4.0 returns a different result set than previous releases of this driver for the `ResultSetMetaData.getColumnLabel` and `ResultSetMetaData.getColumnName` methods to conform to the JDBC 4.0 standard. If you need these methods to return the same result set returned with the IBM DB2 Driver for JDBC and SQLJ prior to Version 4.0, you can set the `useJDBC4ColumnNameAndLabelSemantics` property to `DB2BaseDataSource.NO` in the `Connection` or `DataSource` object.
  - The IBM Data Server Driver for JDBC and SQLJ allows you to invoke the `commit ()` or `rollback ()` methods if the connection is in auto-commit mode and your application does not receive an exception anymore.
  - If the JNDI store is not available due to JNDI bind or lookup failures, then the IBM Data Server Driver for JDBC and SQLJ attempts a connection to the standard server and port properties of a `datasource` even when the `datasource` is configured to use JNDI for client reroute primaries and alternates. The driver now accumulates warnings to indicate these failures with the original message from the exception appended. In previous releases, the driver did not use this information and threw exceptions.
3. If you are migrating applications that use IBM DB2 Driver for JDBC and SQLJ prior to Version 3.1, update your applications to manage the following differences between this driver and the IBM Data Server Driver for JDBC and SQLJ Version 4.0 or Version 3.50:
  - If your applications connect to a DB2 server that supports progressive streaming, also known as dynamic data format, retrieving LOBs using progressive streaming is enabled by default starting with IBM DB2 Driver for JDBC and SQLJ Version 3.2 to provide improved performance to your Java database applications. You need to manage any changes in semantic that might impact your applications. Refer to LOBs in JDBC applications with the IBM Data Server Driver for JDBC and SQLJ in *Developing Java Applications* for details.
  - If your application connects to a DB2 server that supports progressive streaming, and you want to continue using LOB locators instead of LOB

retrieval using progressive streaming, set the `progressiveStreaming` property to: `DB2BaseDataSource.NO` in the `Connection` or `DataSource` object.

- As of Version 3.0, you need to set the `sendDataAsIs` property to indicate if you want the driver to do the data type conversion or not. To maintain the conversion of input parameter values to the target column data types, which was the default behavior prior to IBM DB2 Driver for JDBC and SQLJ Version 3.0, set the `sendDataAsIs` property to `false`. If you set the `sendDataAsIs` property to `true`, the driver converts to the data type indicated by the `setXXX` method regardless of the information in the `Connection` or `DataSource` object.
  - If you use the JDBC 1.0 method to update or delete data on a database server that supports multiple-row `FETCH` and you intent to update or delete a single row, modify your applications to use the method described in *Specifying updatability, scrollability, and holdability for ResultSets in JDBC applications* in *Developing Java Applications* to avoid updating or deleting multiple rows.
4. If you changed your Java application source code, rebuild your Java application. Refer to one of the following tasks for details on how to rebuild them:
- Building JDBC applications in *Developing Java Applications*
  - Building SQLJ applications in *Developing Java Applications*

Upon completion of this task, your Java application should perform successfully using DB2 Version 9.5.

After migrating your Java applications, perform the remaining steps in the migrating database applications task.

---

## Migrating Java applications that use DB2 JDBC Type 2 driver

The DB2 JDBC Type 2 driver is deprecated. Although your Java applications that use the DB2 JDBC Type 2 driver will function successfully with DB2 Version 9.5, migrating those applications to the IBM Data Server Driver for JDBC and SQLJ as soon as possible will help you to avoid possible support problems in future releases.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that the Java application development software is at a version level that is supported by DB2 database products.
- Perform the previous steps in the migrating database applications task.

### Restriction

- The minimum supported Java SDK is Java SDK 1.4.2.

### Procedure

To migrate your Java database applications to DB2 Version 9.5:

1. Install the IBM Data Server Driver for JDBC and SQLJ.
2. Update your Java applications to use the IBM Data Server Driver for JDBC and SQLJ.

3. Review the information in the following topics in *Developing Java Applications* to identify behavioral differences between drivers that might impact your Java application:
  - “Driver support for JDBC APIs”
  - “JDBC differences between the IBM Data Server Driver for JDBC and SQLJ and other DB2 JDBC drivers”
  - “SQLJ differences between the IBM Data Server Driver for JDBC and SQLJ and other DB2 JDBC drivers”
4. Update your Java applications to resolve any issues created by the behavioral differences that you identified in the previous step. These changes might include modifying existing method calls and removing the use of unsupported features in DB2 Version 9.5.
5. If you changed your Java application source code in any of the previous steps, rebuild your Java applications. Refer to one of the following tasks in *Developing Java Applications* for details on how to rebuild them:
  - “Building JDBC applications”
  - “Building SQLJ applications”

After migrating your Java applications, perform the remaining steps in the migrating database applications task.

---

## Migrating ADO.NET applications

Migrating ADO.NET applications that you built for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these applications.

You do not need to migrate ADO.NET applications that use the OLE DB .NET Data Provider or the ODBC .NET Data Provider to run with DB2 Version 9.5. However, migrating these applications to the Developing ADO.NET and OLE DB Applications can be beneficial for the following reasons:

- The Developing ADO.NET and OLE DB Applications has a far more extensive set of APIs than the OLE DB and ODBC .NET data providers.
- Access to the DB2 database development productivity tools integrated with Visual Studio.
- Use of the Developing ADO.NET and OLE DB Applications can bring significant performance improvements.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that a supported version of the Microsoft .NET Framework software is installed on the DB2 database client computer. See “Supported .NET development software” in *Developing ADO.NET and OLE DB Applications*.
- Perform the previous steps in the migrating database applications task.

### Procedure

To migrate your ADO.NET applications to DB2 Version 9.5:

1. Review the support for the Data Server Provider for .NET and how to program your applications to use the Developing ADO.NET and OLE DB Applications and determine what changes to make on your ADO.NET applications.
2. Rebuild your ADO.NET applications to use the Developing ADO.NET and OLE DB Applications. DB2 Version 9.5 features two Developing ADO.NET and OLE DB Applications versions:
  - one for the .NET Framework Version 1.1
  - one for the .NET Framework Version 2.0, which is the most highly optimized data provider for DB2 Version 9.5, and it has a greatly extended set of features, in case that you plan to further develop your applications.

After migrating your ADO.NET applications, perform the remaining steps in the migrating database applications task.

---

## Migrating scripts

Migrating your scripts that use DB2 command line processor (CLP) commands, DB2 system commands or SQL statements involves managing the changes in DB2 Version 9.5 related to SQL statements, DB2 CLP and system commands, SQL Administrative views and routines, built-in functions, and catalog views.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases.
- Ensure that a DB2 Version 9.5 client is installed.
- Perform the previous steps in the migrating database applications task.

### Restriction

This procedure only applies to scripts that use DB2 CLP commands, DB2 system commands or SQL statements.

### Procedure

To migrate your scripts with DB2 CLP commands to DB2 Version 9.5:

1. Run your scripts to detect any incompatibilities with DB2 Version 9.5. If your scripts run successfully, you do not need to perform any additional steps. However, consider performing the remaining steps to remove deprecated features in DB2 Version 9.5 before they become discontinued or to use new command functionality.
2. Remove the DB2 CLP and system commands that display or update registry variables and configuration parameters that are deprecated or discontinued:
  - Deprecated and discontinued registry variables
  - Deprecated and discontinued database manager configuration parameters.
  - Deprecated and discontinued database configuration parameters
3. If your scripts perform snapshot or event monitoring, you need to modify your scripts to remove references to discontinued monitor elements or use a new name when they have been replaced by a new monitor element.
4. Determine the migration impact from system catalog changes. Using the changed views and routines requires that you:
  - Change the view names on your queries.
  - Change column names in your queries for columns that have been renamed in the view or routine.

- Remove column names from your queries for columns that are not available in the view or result sets from routines.
  - Replace \* in your queries for a specific list of column names that you want to receive as a result set because the changed view result set has additional columns.
  - Change routines names and parameter names, and indicate new additional parameters.
  - Modify your script to process additional columns in a result set when calling a changed routine or querying a changed view that returns additional columns.
5. Test your scripts to ensure that they run as expected using DB2 Version 9.5.

After migrating your scripts, perform the remaining steps in the migrating database applications task.

---

## Migrating 32-bit database applications to run on 64-bit instances

Migrating 32-bit database applications from a DB2 Version 8 32-bit instance to a DB2 Version 9.5 64-bit instance requires that your 32-bit database applications are linked to the appropriate shared library path to run them successfully.

You do not need to modify your 32-bit database applications if you linked them to the \$INSTHOME/sqllib/lib32 shared library path on Linux and UNIX or the DB2PATH\lib\Win32 shared library path on Windows, where INSTHOME is the instance home directory and DB2PATH is the location of the DB2 copy.

### Prerequisites

- Ensure that you have access to a DB2 UDB Version 8 32-bit instance that you migrated to a DB2 Version 9.5 64-bit instance that includes 32-bit shared libraries.
- Ensure that the development software is at a version level that is supported by DB2 database products.
- Perform the previous steps in the migrating database applications task.

### Restrictions

- This procedure applies only to 32-bit database applications programmed in C/C++, COBOL, FORTRAN, and REXX.

### Procedure

To migrate 32-bit database applications to run on a DB2 Version 9.5 64-bit instance:

1. Ensure that the library path environment variables include the correct DB2 shared library path for 32-bit libraries as shown in Table 22 on page 120, so that the correct library can be loaded at runtime.
2. Test your 32-bit applications in a DB2 Version 9.5 testing environment. If testing is successful, you do not need to perform any additional steps. However, consider performing step 4 on page 148 or 5 on page 148 if they apply to your applications, to improve their support by using the correct client and shared library path.
3. Perform any other steps in the following migration tasks that apply to your applications:
  - Embedded SQL applications
  - CLI applications

- Java applications that use the IBM Data Server Driver for JDBC and SQLJ or that use the DB2 JDBC Type 2 driver
  - ADO and .NET applications
  - Scripts that use DB2 CLP commands and SQL statements
4. Specify the correct library path by linking or rebuilding your 32-bit applications using the DB2 shared library paths for 32-bit libraries shown in Table 21 on page 120.
  5. Optional: If you no longer have the source code to rebuild your applications or the use of environmental variables is not possible, you can run the `db2chglbpath` command to change the DB2 shared library path to `$INSTHOME/sqllib/lib32` on your application binary file as long as it has an embedded runtime path. The embedded runtime path can be changed to a new path with the same length or less.
  6. Test your 32-bit applications to ensure that they run as expected using DB2 Version 9.5.

After migrating your 32-bit database applications, perform the remaining steps in the migrating database applications task.

---

## Chapter 24. Migrating routines

Migrating routines that you created for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these routines and making sure that they function as expected. Managing these changes might require that you modify your routine code, rebuild your external routines, recreate your external routines in the database, and recreate SQL routines.

Test your routines in a DB2 Version 9.5 testing environment. If they run successfully, you are not required to change them. You only need to modify your routines to manage any changes between releases, to remove the use of discontinued or deprecated functionality in DB2 Version 9.5, or to use new features.

### Prerequisites

- Review migration essentials for routines to identify any changes that apply to your routines.
- Ensure that you have access to migrated DB2 Version 9.5 databases. These can be test databases.
- Ensure that you meet the installation requirements for DB2 database products. See “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers* .
- Ensure that the development software is at a version level that is supported by DB2 database products.
- Perform the pre-migration tasks for routines.
- Ensure that you have SYSADM or DBADM authority to use the following SQL statements:
  - ALTER FUNCTION
  - ALTER PROCEDURE

Other authorizations allowed are listed in the *SQL Reference, Volume 2*.

### Restriction

This procedure only applies to SQL routines and external routines programmed in C/C++, COBOL (procedures only), Java, and .NET languages.

### Procedure

To migrate your routines to DB2 Version 9.5 databases:

1. If you identified changes in DB2 Version 9.5 that impact your routines, edit your routine code and modify:
  - SQL statement syntax
  - SQL statements using SQL Administrative views and routines, built-in routines, and catalog views
  - User defined routine names that are not fully qualified with a schema names
  - Application programming interface calls such as JDBC and CLI
2. If you identified changes specific to the development environment that impact your routines, modify them to support these changes. Migrate your:
  - C, C++, and COBOL routines
  - Java routines.

- .NET CLR routines.
  - SQL stored procedures, if you created your SQL procedures in DB2 Version 8.1 and you migrated from a 32-bit DB2 Version 8 instance to a DB2 Version 9.5 64-bit instance.
  - 32-bit external routines to run on 64-bit instances.
3. Rebuild all changed external routine libraries or if you performed operating system or development software upgrades.
  4. Test your routines to verify your changes and to ensure that the routines run as expected using DB2 Version 9.5.

After migrating your routines, perform the recommended post-migration tasks for routines.

---

## Migrating C, C++, and COBOL routines

Migrating C, C++, or COBOL routines that you created for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these routines and making sure that they function as expected.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that the C, C++, or COBOL routine development software are at a version level that is supported by DB2 database products by reviewing the following requirements:
  - “Support for external routine development in C” in *Administrative Routines and Views*
  - “Support for external routine development in C++” in *Administrative Routines and Views*
  - “Support for external procedure development in COBOL” in *Administrative Routines and Views*
- Ensure that you have SYSADM or DBADM authority to use the following statements:
  - ALTER FUNCTION
  - ALTER PROCEDURE

Other authorizations allowed are listed in the *SQL Reference, Volume 2*.

- Perform the previous steps in the migrating routines task.

### Restriction

This procedure only applies to external routines programmed in C/C++, and COBOL (procedures only).

### Procedure

To migrate a C, C++, or COBOL routine to DB2 Version 9.5, do the following:

1. If you migrated to a DB2 Version 9.5 64-bit instance, change your routine libraries or routine definitions according to the following table:

Table 31. Migrating C, C++, and COBOL routines to a Version 9.5 64-bit instance

| Routine definition                                                     | Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>unfenced</i> 32-bit routine library that use the DB2 engine library | <p>Rebuild the routine source code into a 64-bit library using the DB2 Version 9.5 bldrtn script and redeploy the library to the DB2 server. If LOB locators are referenced in the routine, you must rebuild your routines. You can determine most of the routines that reference lob locators by executing the following query:</p> <pre>SELECT DISTINCT a.routineschema, a.routinename, a.specificname FROM syscat.routines a, syscat.routineparms b WHERE a.specificname = b.specificname AND b.locator = 'Y' AND a.fenced = 'N'</pre> <p>An advantage of this approach is that using a 64-bit library results in better routine runtime performance than using a 32-bit library.</p> |
| <i>fenced</i> 32-bit routine library                                   | <ul style="list-style-type: none"> <li>• Rebuild the routine source code into a 64-bit library using the DB2 Version 9.5 bldrtn scripts and redeploy the library to the DB2 server.</li> <li>• If you cannot rebuild your routines, define the routine as not threadsafe using the ALTER PROCEDURE or ALTER FUNCTION statement with the NOT THREADSAFE clause.</li> </ul>                                                                                                                                                                                                                                                                                                                |
| migrated from a Version 8 32-bit instance (AIX and Windows)            | <p>You should specify a library entry point for any routine that relies on a default entry point using the ALTER PROCEDURE or ALTER FUNCTION statement. For example, to explicitly specify the entry point for an existing procedure use the following statement:</p> <pre>ALTER SPECIFIC PROCEDURE <i>schema-name.specific-name</i> EXTERNAL NAME '<i>library-name!function-name</i>'</pre> <p>where <i>library-name</i> is the library to be loaded and <i>function-name</i> is the explicit entry point for the function associated with the routine.</p>                                                                                                                             |

If none of the previously mentioned situations apply, you do not need to change your routine libraries or routine definitions.

2. If you are using the cursor blocking and found any differences in the behavior of your Java routines, review the “Migrating embedded SQL applications” on page 140 task to learn how to manage those differences.
3. For routines that you did not rebuild but that you modified, rebind the routine packages to the target DB2 database.
4. Determine if the external routines that were altered during database migration or the external routines that use the DB2 engine libraries can safely run as NOT FENCED and THREADSAFE. If you have external unfenced routines in your database, the MIGRATE DATABASE command performs the following actions:
  - Returns the warning message SQL1349W.
  - Redefines all your external unfenced routines that have no dependency on the DB2 engine library as FENCED and NOT THREADSAFE.
  - Creates a CLP script called alter\_unfenced\_dbname.db2 in the directory specified by the DIAGPATH database manager configuration parameter to redefine the affected routines as NOT FENCED and THREADSAFE.

If you can safely run the external routines altered by database migration as NOT FENCED and THREADSAFE, you can redefine them as NOT FENCED and THREADSAFE using the original CLP script or a modified version with just specific routines that you want to redefine. You do not need to redefine your routines if you can run them as FENCED and NOT THREADSAFE.

After migrating your C, C++, or COBOL routines, perform the remaining steps in the migrating routines task.

---

## Migrating Java routines

Migrating Java routines that you created for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these routines and making sure that they function as expected.

### Prerequisites

The following prerequisites must be met to perform this task:

- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be a test system.
- Ensure that the Java routine development software is at a version level that is supported by DB2 database products. See “Supported Java routine development software” in *Developing User-defined Routines (SQL and External)*.
- Ensure that you are using supported DB2 drivers for JDBC and SQLJ APIs. See “Supported drivers for JDBC and SQLJ” in *Developing Java Applications*.
- Ensure that you have SYSADM or DBADM authority to use the following statements:

- ALTER FUNCTION
- ALTER PROCEDURE

Other authorizations allowed are listed in the *SQL Reference, Volume 2*.

- Perform the previous steps in the migrating routines task.

### Procedure

To migrate your Java routines:

1. Ensure the **jdk\_path** database manager configuration parameter specifies the correct JVM to run your routines. Determine the current value by issuing the following command:

```
db2 GET DBM CFG
```

By default the **jdk\_path** database manager configuration parameter value is set during instance migration to the values shown in Table 30 on page 136. If you want to use a JVM other than the one installed in your DB2 Version 9.5 copy, you must set this configuration parameter to that JVM path with the same bit width as the DB2 instance by updating the **jdk\_path** parameter:

```
db2 UPDATE DBM CFG USING jdk_path <JVM-path>
```

2. Set the **DB2\_USE\_DB2JCCT2\_JROUTINE** registry variable to indicate the default JDBC driver to run Java routines. By default this registry variable is not set, which means the default JDBC driver is the IBM Data Server Driver for JDBC and SQLJ. This setting gives you access to features particular to this driver and use of XML parameters. Use the **db2set** command with the **-g** option to set the default JDBC driver for all instances running under the same DB2 Version 9.5 copy:

| Default driver                           | Command to set default driver           |
|------------------------------------------|-----------------------------------------|
| IBM DB2 JDBC Type 2 driver               | db2set -g DB2_USE_DB2JCCT2_JROUTINE=NO  |
| IBM Data Server Driver for JDBC and SQLJ | db2set -g DB2_USE_DB2JCCT2_JROUTINE=YES |

Use the *-i instance name* option instead of the *-g* option, to apply the registry variable setting to a specific instance.

3. Test your Java routines in your DB2 Version 9.5 database. If testing is successful and your Java routine perform as expected, you do not need to perform any additional steps.
4. If you are using the IBM Data Server Driver for JDBC and SQLJ and found any differences in the behavior of your Java routines, review the migrating Java applications task to learn how to manage those differences.
5. Explicitly define your Java routines as fenced using the ALTER FUNCTION or ALTER PROCEDURE statement with the FENCED clause. All Java routines run as fenced, regardless of how you defined them, but defining your Java routine definitions as fenced improves routine manageability and maintenance.
6. Optional: If your Java routine class is included within a JAR file that has been installed into a DB2 instance using a specific JAR file ID, ensure that the Java class is resolved more quickly by the DB2 database manager by specifying the JAR file ID as part of the EXTERNAL NAME clause in the routine definition. Use the ALTER PROCEDURE or ALTER FUNCTION statement to update the EXTERNAL NAME clause if required.
7. If you created projects in the Development Center to develop your Java routines, migrate any existing projects to the Data Studio using the migration wizard.

After migrating your Java routines, perform the remaining steps in the migrating routines task.

---

## Migrating .NET CLR routines

Migrating your .NET CLR routines that you created for DB2 Version 9.1 or DB2 UDB Version 8 involves managing the changes in DB2 Version 9.5 that impact these routines and making sure that they function as expected.

### Prerequisites

- Review the migration essentials for routines to identify key changes that might apply to your .NET CLR routines.
- Ensure that you have access to a DB2 Version 9.5 server, including instances and databases. The DB2 server can be part of a testing environment.
- Ensure that a supported version of the Microsoft .NET Framework software is installed on the DB2 server.
- Perform the previous steps in the migrating routines task.

### Procedure

To migrate your .NET CLR routines to DB2 Version 9.5:

1. Connect to the DB2 Version 9.5 database in which you defined the .NET CLR routines.
2. Rebuild your .NET CLR routine source code using the compile and link options specified in bldrtn.bat, the DB2 sample script for building .NET CLR routines.
3. Deploy the routine assembly to the DB2 server in the same location specified by the EXTERNAL clause in the routine definition. The routines should function successfully, with no differences in between previous releases and DB2 Version 9.5.

After migrating your .NET CLR routines, perform the remaining steps in the migrating routines task.

---

## Migrating SQL procedures

SQL procedures that you created in DB2 Version 9.1 or DB2 UDB Version 8.2 are automatically migrated when you migrate your databases. SQL procedures that you created before those releases might require manual migration.

If you migrated from a DB2 UDB Version 8 instance to a DB2 Version 9.5 instance with the same bit size, your routines will run successfully in DB2 Version 9.5. However, if you created your SQL procedures in DB2 UDB Version 8.1 and migrated from a DB2 UDB Version 8 32-bit instance to a DB2 Version 9.5 64-bit instance, you must drop and re-create those SQL procedures as part of the manual migration process.

### Prerequisites

- Ensure that you have access to your migrated database on DB2 Version 9.5.
- Ensure that you have the necessary authorizations and privileges to use the CREATE PROCEDURE and DROP PROCEDURE statements. You can find the complete list of authorizations and privileges required in the *SQL Reference, Volume 2*.
- Perform the previous steps in the migrating routines task.

### Restriction

This procedure applies only to SQL procedures that were created in DB2 UDB Version 8.1 prior to FixPak 7 (also known as Version 8.2).

### Procedure

To manually migrate your SQL procedures to DB2 Version 9.5:

1. Connect to the migrated database.
2. Run the following query to identify the SQL procedures that you need to re-create:

```
SELECT procschema, specificname
FROM syscat.procedures
WHERE language = 'SQL' AND fenced = 'N' AND
 substr(IMPLEMENTATION, 10,6) = 'pgsjmp'
```

Take note of the schema and specific name values returned by this query, because you will need this information to perform subsequent steps.

3. Run the db2look tool to generate a DDL script for all your database objects:

```
db2look -d sample -e -o db2look.sql -a
```

where sample is the database name, the -e option generates DDL statements for database objects, the -o db2look.sql option indicates the output file that will contain the DDL statements, and the -a option indicates all objects created by all users.

Edit the db2look.sql file to keep only the DDL statements necessary to create the SQL procedures that you identified in step 2.

4. For each SQL stored procedures that you identified in step 2, use the DROP PROCEDURE statement indicating the schema name and specific name to uniquely identify each procedure:

```
DROP SPECIFIC PROCEDURE <schema-name>.<specific-name>
```

Alternatively, if you have a DDL script that drops and re-creates your SQL procedures, edit it to drop and re-create only the SQL procedures identified in step 2 on page 154, and run it. Then proceed to step 6.

5. Re-create the SQL procedures identified in step 2 on page 154 using the CREATE PROCEDURE statement. Alternatively, you can run your own DDL script or the db2look.sql file that you created in step 3 on page 154.
6. Test your SQL procedures to ensure that they run as expected under DB2 Version 9.5. You can use the Data Studio or the Command Line Processor (CLP) interface to test them. The following example illustrates how to invoke an SQL procedure using the CLP :

```
CONNECT TO sample
```

```
Database Connection Information
```

```
Database server = DB2/AIX64 9.5.0
SQL authorization ID = TESTDB2
Local database alias = SAMPLE
```

```
CALL <schema-name>.<procedure-name> ([<parameter-list>])
```

7. If you created projects in the Development Center to develop your SQL procedures, migrate any existing projects to the Data Studio using the migration wizard. You can use projects that you created in the Developer Workbench without having to migrate them.

After migrating your SQL procedures, perform the remaining steps in the migrating routines task.

---

## Migrating 32-bit external routines to run on 64-bit instances

Migrating 32-bit external routines that you created for DB2 Version 9.1 or DB2 UDB Version 8 to a DB2 Version 9.5 64-bit instance involves managing the changes in DB2 Version 9.5 that impact these routines and making sure that they function as expected.

### Prerequisites

- Ensure that you have access to a DB2 Version 9.5 64-bit instance that includes 32-bit shared libraries.
- Ensure that the development software is at a version level that is supported by DB2 database products.
- Ensure that you have SYSADM or DBADM authority to use the following SQL statements:
  - ALTER FUNCTION
  - ALTER PROCEDURE

Other authorizations allowed are listed in the *SQL Reference, Volume 2*.

- Perform the previous steps in the migrating routines task.

### Restrictions

- This procedure applies only to 32-bit external routines programmed in C and COBOL.
- This procedure indicates only the changes that are required to run 32-bit external routines on a 64-bit instance that includes 32-bit shared libraries.

### Procedure

To migrate 32-bit external routines to run on a DB2 Version 9.5 64-bit instance:

1. Ensure that the library path environment variables include the correct DB2 shared library path for 32-bit libraries as shown in Table 22 on page 120, so that the correct library can be loaded at runtime.
2. Test your routines in a DB2 Version 9.5 testing environment. If testing is successful, you do not need to perform any additional steps. However, consider performing the remaining steps in this task if they apply to your routine for better support by using the correct library path and development software.
3. Specify the correct library path by linking or rebuilding your 32-bit external routines using the DB2 shared library paths for 32-bit libraries shown in Table 21 on page 120. If you migrated from a DB2 UDB Version 8 32-bit instance to a DB2 Version 9.5 64-bit instance, you must rebuild 32-bit external routines that use LOB locators as 64-bit routine libraries.
4. Optional: If you no longer have the source code to rebuild your routine library or use environmental variables, you can use the `db2chglbpath` command to change the DB2 shared library path to `$INSTHOME/sqllib/lib32` on your routine binary file as long as it has an embedded runtime path. The embedded runtime path can be changed to a new path with the same length or less.
5. Perform any other steps in the “Migrating C, C++, and COBOL routines” on page 150 task that apply to your routines.
6. Determine if the external routines that were altered during database migration or the external routines that use the DB2 engine libraries can safely run as `NOT FENCED` and `THREADSAFE`. If you have external unfenced routines in your database, the `MIGRATE DATABASE` command performs the following actions:
  - Returns the warning message `SQL1349W`.
  - Redefines all your external unfenced routines that have no dependency on the DB2 engine library as `FENCED` and `NOT THREADSAFE`.
  - Creates a CLP script called `alter_unfenced_dbname.db2` in the directory specified by the `DIAGPATH` database manager configuration parameter to redefine the affected routines as `NOT FENCED` and `THREADSAFE`.

If you can safely run the external routines altered by database migration as `NOT FENCED` and `THREADSAFE`, you can redefine them as `NOT FENCED` and `THREADSAFE` using the original CLP script or a modified version with just specific routines that you want to redefine. You do not need to redefine your routines if you can run them as `FENCED` and `NOT THREADSAFE`.

After migrating your 32-bit external routines, perform the remaining steps in the migrating routines task.

---

## Chapter 25. Post-migration tasks for database applications and routines

After migrating your database applications and routines, you should perform several post-migration tasks to ensure that your database applications and routines perform as expected and at their optimum levels.

Perform the following post-migration tasks that apply to your database applications and routines:

1. Tune your database applications. Review important guidelines related to:

- Character conversion
- Optimization class
- Specifying the isolation level
- Locks and concurrency
- Parallel processing for applications

Refer to the *Tuning Database Performance* for complete details on how to tune applications.

2. Tune your routines. Review important guidelines related to:

- Stored procedures
- SQL procedures

In addition, review guidelines on improving the performance of database applications that also apply to routines, such as the guidelines on optimization classes, locks, concurrency, and query tuning.

3. Remove dependencies on features that are deprecated in DB2 Version 9.5 in your database applications and routines before those features become discontinued.

4. Start enabling new DB2 Version 9.5 features in database applications, where appropriate, to improve performance or add new functionality. Check the Sample files to understand how the new features work.



---

## Chapter 26. Enabling new DB2 Version 9.5 functionality in database applications and routines

After migrating to DB2 Version 9.5, enhance the functionality and improve the performance of your database applications by enabling new functionality.

You must migrate your DB2 server to DB2 Version 9.5.

For applications that access migrated databases, perform the following steps to enable the following DB2 Version 9.5 functionality:

1. Use of the new optimistic locking functionality in your applications by performing the following steps:
  - a. Add a row change timestamp column with the `IMPLICITLY HIDDEN` clause so that adding this column does not impact existing queries in your applications.

```
db2 ALTER TABLE staff ADD COLUMN RCT TIMESTAMP NOT NULL
 GENERATED ALWAYS
 FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP
 IMPLICITLY HIDDEN
```

- b. Use the `RID_BIT` or `RID` system built-in function in positioned updates or deletes. See “Enabling optimistic locking in applications” in *Data Servers, Databases, and Database Objects Guide* for details.

See “Optimistic locking overview” in *Data Servers, Databases, and Database Objects Guide* for additional details.

2. Enable your applications and routines to benefit from optimizer enhancements. The optimizer now chooses more optimal query execution plans for specific types of queries. Review the Optimizer enhancements section to determine if you can modify the queries in your applications to become queries targeted by these enhancements.
3. Enable the use of the `ARRAY` collection data type by declaring and using variables and parameters of `ARRAY` type in stored procedures and applications. Arrays are transient values that are stored in tables. See “Array support enhances application portability” in *What’s New*.
4. Add the use of global variables in your applications to share data between SQL statements without the need for application logic to support this data transfer. You can control access to global variables through the `GRANT` and `REVOKE` statements. See “Global variables improve data sharing between SQL statements” in *What’s New*.



---

## Part 5. Appendixes



---

## Appendix A. Important references

The following list of references can help you with the migration of your DB2 environment.

### **DB2 operating system requirements Web page**

You can find the operating system and hardware requirements for DB2 Version 9.5 installation in “Installation requirements for DB2 database products” in *Quick Beginnings for DB2 Servers*. The most up to date version of this topic is available in the DB2 Information Center at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/topic/com.ibm.db2.luw.qb.server.doc/doc/r0025127.html>.

### **DB2 Information Center**

You can find the information in this book in the online DB2 Information Center at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/index.jsp>. Refer to the “Migrating” topic under the “Database fundamentals” section. The title for the most high level topic is “Migrating to DB2 Version 9.5”. The online DB2 Information Center also contains information on migration-related topics such as DB2 database product installation. You can also find other information referenced in this book.

### **DB2 DB2 Version 9.5 manuals in PDF format**

DB2 DB2 Version 9.5 manuals in PDF format are available for complimentary download at <http://www.ibm.com/support/docview.wss?rs=71&uid=swg27009474>.

### **DB2 migration portal**

The DB2 migration portal at <http://www.ibm.com/support/docview.wss?rs=73&uid=swg21200005> provides you with a single place for accessing up-to-date information about the migration process and additional resources as they become available.

### **DB2 database product education**

The Information Management Training Web site at <http://www.ibm.com/software/data/education/> offers a wide variety of training options and the list of skills resources and communities to help you find the educational resources that are right for you. Review the list of complimentary DB2 database product self-study courses that can help you build skills at your own pace at <http://www.ibm.com/software/data/education/selfstudy.html>.

### **developerWorks Information Management Web site**

The developerWorks Information Management Web site at <http://www.ibm.com/developerworks/db2> offers technical resources for DB2 Information Management software. It features product information, downloads, learning resources, support, forums, and newsletters. On this Web site you can find many articles and tutorials that can help you to learn about new features of DB2 database products and how to use them in your applications.

This Web site also offers portals of learning resources such as New to DB2, Migrate to DB2, and DBA Central. Follow the **Migrate to DB2** link to

access resources that can help you migrate from Microsoft SQL Server, Oracle, Sybase, and other database platforms to DB2 Database products.

### **DB2 forums**

The DB2 forums are places to exchange ideas and share solutions with your peers in the IBM DB2 product community. In addition, DB2 forums include forums that are mirrors to DB2 newsgroups, such as the `ibm.software.db2.udb` and `ibm.software.db2.udb.beta` newsgroups. The DB2 forums are hosted by developerWorks at [http://www.ibm.com/developerworks/forums/db2\\_forums.jsp](http://www.ibm.com/developerworks/forums/db2_forums.jsp).

---

## Appendix B. Overview of the DB2 technical information

DB2 technical information is available through the following tools and methods:

- DB2 Information Center
  - Topics (Task, concept and reference topics)
  - Help for DB2 tools
  - Sample programs
  - Tutorials
- DB2 books
  - PDF files (downloadable)
  - PDF files (from the DB2 PDF DVD)
  - printed books
- Command line help
  - Command help
  - Message help

**Note:** The DB2 Information Center topics are updated more frequently than either the PDF or the hard-copy books. To get the most current information, install the documentation updates as they become available, or refer to the DB2 Information Center at [ibm.com](http://ibm.com)<sup>®</sup>.

You can access additional DB2 technical information such as technotes, white papers, and IBM Redbooks<sup>®</sup> publications online at [ibm.com](http://www.ibm.com). Access the DB2 Information Management software library site at <http://www.ibm.com/software/data/sw-library/>.

### Documentation feedback

We value your feedback on the DB2 documentation. If you have suggestions for how to improve the DB2 documentation, send an email to [db2docs@ca.ibm.com](mailto:db2docs@ca.ibm.com). The DB2 documentation team reads all of your feedback, but cannot respond to you directly. Provide specific examples wherever possible so that we can better understand your concerns. If you are providing feedback on a specific topic or help file, include the topic title and URL.

Do not use this email address to contact DB2 Customer Support. If you have a DB2 technical issue that the documentation does not resolve, contact your local IBM service center for assistance.

---

## DB2 technical library in hardcopy or PDF format

The following tables describe the DB2 library available from the IBM Publications Center at [www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order). English DB2 Version 9.5 manuals in PDF format and translated versions can be downloaded from [www.ibm.com/support/docview.wss?rs=71&uid=swg2700947](http://www.ibm.com/support/docview.wss?rs=71&uid=swg2700947).

Although the tables identify books available in print, the books might not be available in your country or region.

The form number increases each time a manual is updated. Ensure that you are reading the most recent version of the manuals, as listed below.

**Note:** The DB2 Information Center is updated more frequently than either the PDF or the hard-copy books.

*Table 32. DB2 technical information*

| <b>Name</b>                                                                          | <b>Form Number</b> | <b>Available in print</b> |
|--------------------------------------------------------------------------------------|--------------------|---------------------------|
| <i>Administrative API Reference</i>                                                  | SC23-5842-01       | Yes                       |
| <i>Administrative Routines and Views</i>                                             | SC23-5843-01       | No                        |
| <i>Call Level Interface Guide and Reference, Volume 1</i>                            | SC23-5844-01       | Yes                       |
| <i>Call Level Interface Guide and Reference, Volume 2</i>                            | SC23-5845-01       | Yes                       |
| <i>Command Reference</i>                                                             | SC23-5846-01       | Yes                       |
| <i>Data Movement Utilities Guide and Reference</i>                                   | SC23-5847-01       | Yes                       |
| <i>Data Recovery and High Availability Guide and Reference</i>                       | SC23-5848-01       | Yes                       |
| <i>Data Servers, Databases, and Database Objects Guide</i>                           | SC23-5849-01       | Yes                       |
| <i>Database Security Guide</i>                                                       | SC23-5850-01       | Yes                       |
| <i>Developing ADO.NET and OLE DB Applications</i>                                    | SC23-5851-01       | Yes                       |
| <i>Developing Embedded SQL Applications</i>                                          | SC23-5852-01       | Yes                       |
| <i>Developing Java Applications</i>                                                  | SC23-5853-01       | Yes                       |
| <i>Developing Perl and PHP Applications</i>                                          | SC23-5854-01       | No                        |
| <i>Developing User-defined Routines (SQL and External)</i>                           | SC23-5855-01       | Yes                       |
| <i>Getting Started with Database Application Development</i>                         | GC23-5856-01       | Yes                       |
| <i>Getting Started with DB2 installation and administration on Linux and Windows</i> | GC23-5857-01       | Yes                       |
| <i>Internationalization Guide</i>                                                    | SC23-5858-01       | Yes                       |
| <i>Message Reference, Volume 1</i>                                                   | GI11-7855-00       | No                        |
| <i>Message Reference, Volume 2</i>                                                   | GI11-7856-00       | No                        |
| <i>Migration Guide</i>                                                               | GC23-5859-01       | Yes                       |
| <i>Net Search Extender Administration and User's Guide</i>                           | SC23-8509-01       | Yes                       |
| <i>Partitioning and Clustering Guide</i>                                             | SC23-5860-01       | Yes                       |
| <i>Query Patroller Administration and User's Guide</i>                               | SC23-8507-00       | Yes                       |
| <i>Quick Beginnings for IBM Data Server Clients</i>                                  | GC23-5863-01       | No                        |

*Table 32. DB2 technical information (continued)*

| <b>Name</b>                                                                             | <b>Form Number</b> | <b>Available in print</b> |
|-----------------------------------------------------------------------------------------|--------------------|---------------------------|
| <i>Quick Beginnings for DB2 Servers</i>                                                 | GC23-5864-01       | Yes                       |
| <i>Spatial Extender and Geodetic Data Management Feature User's Guide and Reference</i> | SC23-8508-01       | Yes                       |
| <i>SQL Reference, Volume 1</i>                                                          | SC23-5861-01       | Yes                       |
| <i>SQL Reference, Volume 2</i>                                                          | SC23-5862-01       | Yes                       |
| <i>System Monitor Guide and Reference</i>                                               | SC23-5865-01       | Yes                       |
| <i>Troubleshooting Guide</i>                                                            | GI11-7857-01       | No                        |
| <i>Tuning Database Performance</i>                                                      | SC23-5867-01       | Yes                       |
| <i>Visual Explain Tutorial</i>                                                          | SC23-5868-00       | No                        |
| <i>What's New</i>                                                                       | SC23-5869-01       | Yes                       |
| <i>Workload Manager Guide and Reference</i>                                             | SC23-5870-01       | Yes                       |
| <i>pureXML Guide</i>                                                                    | SC23-5871-01       | Yes                       |
| <i>XQuery Reference</i>                                                                 | SC23-5872-01       | No                        |

*Table 33. DB2 Connect-specific technical information*

| <b>Name</b>                                              | <b>Form Number</b> | <b>Available in print</b> |
|----------------------------------------------------------|--------------------|---------------------------|
| <i>Quick Beginnings for DB2 Connect Personal Edition</i> | GC23-5839-01       | Yes                       |
| <i>Quick Beginnings for DB2 Connect Servers</i>          | GC23-5840-01       | Yes                       |
| <i>DB2 Connect User's Guide</i>                          | SC23-5841-01       | Yes                       |

*Table 34. Information Integration technical information*

| <b>Name</b>                                                                                   | <b>Form Number</b> | <b>Available in print</b> |
|-----------------------------------------------------------------------------------------------|--------------------|---------------------------|
| <i>Information Integration: Administration Guide for Federated Systems</i>                    | SC19-1020-01       | Yes                       |
| <i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i> | SC19-1018-02       | Yes                       |
| <i>Information Integration: Configuration Guide for Federated Data Sources</i>                | SC19-1034-01       | No                        |
| <i>Information Integration: SQL Replication Guide and Reference</i>                           | SC19-1030-01       | Yes                       |
| <i>Information Integration: Introduction to Replication and Event Publishing</i>              | SC19-1028-01       | Yes                       |

---

## Ordering printed DB2 books

If you require printed DB2 books, you can buy them online in many but not all countries or regions. You can always order printed DB2 books from your local IBM representative. Keep in mind that some softcopy books on the *DB2 PDF Documentation DVD* are unavailable in print. For example, neither volume of the *DB2 Message Reference* is available as a printed book.

Printed versions of many of the DB2 books available on the *DB2 PDF Documentation DVD* can be ordered for a fee from IBM. Depending on where you are placing your order from, you may be able to order books online, from the IBM Publications Center. If online ordering is not available in your country or region, you can always order printed DB2 books from your local IBM representative. Note that not all books on the *DB2 PDF Documentation DVD* are available in print.

**Note:** The most up-to-date and complete DB2 documentation is maintained in the DB2 Information Center at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5>.

To order printed DB2 books:

- To find out whether you can order printed DB2 books online in your country or region, check the IBM Publications Center at <http://www.ibm.com/shop/publications/order>. You must select a country, region, or language to access publication ordering information and then follow the ordering instructions for your location.
- To order printed DB2 books from your local IBM representative:
  1. Locate the contact information for your local representative from one of the following Web sites:
    - The IBM directory of world wide contacts at [www.ibm.com/planetwide](http://www.ibm.com/planetwide)
    - The IBM Publications Web site at <http://www.ibm.com/shop/publications/order>. You will need to select your country, region, or language to the access appropriate publications home page for your location. From this page, follow the "About this site" link.
  2. When you call, specify that you want to order a DB2 publication.
  3. Provide your representative with the titles and form numbers of the books that you want to order. For titles and form numbers, see "DB2 technical library in hardcopy or PDF format" on page 165.

---

## Displaying SQL state help from the command line processor

DB2 returns an `SQLSTATE` value for conditions that could be the result of an SQL statement. `SQLSTATE` help explains the meanings of SQL states and SQL state class codes.

To invoke SQL state help, open the command line processor and enter:

```
? sqlstate or ? class code
```

where *sqlstate* represents a valid five-digit SQL state and *class code* represents the first two digits of the SQL state.

For example, ? 08003 displays help for the 08003 SQL state, and ? 08 displays help for the 08 class code.

---

## Accessing different versions of the DB2 Information Center

For DB2 Version 9.5 topics, the DB2 Information Center URL is <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>

For DB2 Version 9 topics, the DB2 Information Center URL is <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>

For DB2 Version 8 topics, go to the Version 8 Information Center URL at: <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>

---

## Displaying topics in your preferred language in the DB2 Information Center

The DB2 Information Center attempts to display topics in the language specified in your browser preferences. If a topic has not been translated into your preferred language, the DB2 Information Center displays the topic in English.

- To display topics in your preferred language in the Internet Explorer browser:
    1. In Internet Explorer, click the **Tools** → **Internet Options** → **Languages...** button. The Language Preferences window opens.
    2. Ensure your preferred language is specified as the first entry in the list of languages.
      - To add a new language to the list, click the **Add...** button.

**Note:** Adding a language does not guarantee that the computer has the fonts required to display the topics in the preferred language.

    - To move a language to the top of the list, select the language and click the **Move Up** button until the language is first in the list of languages.  - 3. Clear the browser cache and then refresh the page to display the DB2 Information Center in your preferred language.
- To display topics in your preferred language in a Firefox or Mozilla browser:
  1. Select the button in the **Languages** section of the **Tools** → **Options** → **Advanced** dialog. The Languages panel is displayed in the Preferences window.
  2. Ensure your preferred language is specified as the first entry in the list of languages.
    - To add a new language to the list, click the **Add...** button to select a language from the Add Languages window.
    - To move a language to the top of the list, select the language and click the **Move Up** button until the language is first in the list of languages.
  3. Clear the browser cache and then refresh the page to display the DB2 Information Center in your preferred language.

On some browser and operating system combinations, you might have to also change the regional settings of your operating system to the locale and language of your choice.

---

## Updating the DB2 Information Center installed on your computer or intranet server

If you have installed the DB2 Information Center locally, you can obtain and install documentation updates from IBM.

Updating your locally-installed DB2 Information Center requires that you:

1. Stop the DB2 Information Center on your computer, and restart the Information Center in stand-alone mode. Running the Information Center in stand-alone mode prevents other users on your network from accessing the Information Center, and allows you to apply updates. Non-Administrative and Non-Root DB2 Information Centers always run in stand-alone mode. .
2. Use the Update feature to see what updates are available. If there are updates that you would like to install, you can use the Update feature to obtain and install them

**Note:** If your environment requires installing the DB2 Information Center updates on a machine that is not connected to the internet, you have to mirror the update site to a local file system using a machine that is connected to the internet and has the DB2 Information Center installed. If many users on your network will be installing the documentation updates, you can reduce the time required for individuals to perform the updates by also mirroring the update site locally and creating a proxy for the update site.

If update packages are available, use the Update feature to get the packages. However, the Update feature is only available in stand-alone mode.

3. Stop the stand-alone Information Center, and restart the DB2 Information Center on your computer.

**Note:** On Windows Vista, the commands listed below must be run as an administrator. To launch a command prompt or graphical tool with full administrator privileges, right-click on the shortcut and then select **Run as administrator**.

To update the DB2 Information Center installed on your computer or intranet server:

1. Stop the DB2 Information Center.
  - On Windows, click **Start** → **Control Panel** → **Administrative Tools** → **Services**. Then right-click on **DB2 Information Center** service and select **Stop**.
  - On Linux, enter the following command:  

```
/etc/init.d/db2icdv95 stop
```
2. Start the Information Center in stand-alone mode.
  - On Windows:
    - a. Open a command window.
    - b. Navigate to the path where the Information Center is installed. By default, the DB2 Information Center is installed in the <Program Files>\IBM\DB2 Information Center\Version 9.5 directory, where <Program Files> represents the location of the Program Files directory.
    - c. Navigate from the installation directory to the doc\bin directory.
    - d. Run the help\_start.bat file:  

```
help_start.bat
```
  - On Linux:

- a. Navigate to the path where the Information Center is installed. By default, the DB2 Information Center is installed in the /opt/ibm/db2ic/V9.5 directory.
- b. Navigate from the installation directory to the doc/bin directory.
- c. Run the help\_start script:  

```
help_start
```

The systems default Web browser launches to display the stand-alone Information Center.

3. Click the **Update** button (🔧). On the right hand panel of the Information Center, click **Find Updates**. A list of updates for existing documentation displays.
4. To initiate the installation process, check the selections you want to install, then click **Install Updates**.
5. After the installation process has completed, click **Finish**.
6. Stop the stand-alone Information Center:

- On Windows, navigate to the installation directory's doc\bin directory, and run the help\_end.bat file:

```
help_end.bat
```

**Note:** The help\_end batch file contains the commands required to safely terminate the processes that were started with the help\_start batch file. Do not use Ctrl-C or any other method to terminate help\_start.bat.

- On Linux, navigate to the installation directory's doc/bin directory, and run the help\_end script:

```
help_end
```

**Note:** The help\_end script contains the commands required to safely terminate the processes that were started with the help\_start script. Do not use any other method to terminate the help\_start script.

7. Restart the DB2 Information Center.
  - On Windows, click **Start** → **Control Panel** → **Administrative Tools** → **Services**. Then right-click on **DB2 Information Center** service and select **Start**.
  - On Linux, enter the following command:  

```
/etc/init.d/db2icdv95 start
```

The updated DB2 Information Center displays the new and updated topics.

---

## DB2 tutorials

The DB2 tutorials help you learn about various aspects of DB2 products. Lessons provide step-by-step instructions.

### Before you begin

You can view the XHTML version of the tutorial from the Information Center at <http://publib.boulder.ibm.com/infocenter/db2help/>.

Some lessons use sample data or code. See the tutorial for a description of any prerequisites for its specific tasks.

## DB2 tutorials

To view the tutorial, click on the title.

**“pureXML™” in *pureXML Guide***

Set up a DB2 database to store XML data and to perform basic operations with the native XML data store.

**“Visual Explain” in *Visual Explain Tutorial***

Analyze, optimize, and tune SQL statements for better performance using Visual Explain.

---

## DB2 troubleshooting information

A wide variety of troubleshooting and problem determination information is available to assist you in using DB2 products.

### DB2 documentation

Troubleshooting information can be found in the DB2 Troubleshooting Guide or the Support and Troubleshooting section of the DB2 Information Center. There you will find information on how to isolate and identify problems using DB2 diagnostic tools and utilities, solutions to some of the most common problems, and other advice on how to solve problems you might encounter with your DB2 products.

### DB2 Technical Support Web site

Refer to the DB2 Technical Support Web site if you are experiencing problems and want help finding possible causes and solutions. The Technical Support site has links to the latest DB2 publications, TechNotes, Authorized Program Analysis Reports (APARs or bug fixes), fix packs, and other resources. You can search through this knowledge base to find possible solutions to your problems.

Access the DB2 Technical Support Web site at <http://www.ibm.com/software/data/db2/udb/support.html>

---

## Terms and Conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

**Personal use:** You may reproduce these Publications for your personal, non commercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these Publications, or any portion thereof, without the express consent of IBM.

**Commercial use:** You may reproduce, distribute and display these Publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these Publications, or reproduce, distribute or display these Publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the Publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the Publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.



---

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

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

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

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

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

**The following paragraph does not apply to the United Kingdom or any other country/region where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This document may provide links or references to non-IBM Web sites and resources. IBM makes no representations, warranties, or other commitments whatsoever about any non-IBM Web sites or third-party resources that may be referenced, accessible from, or linked from this document. A link to a non-IBM Web site does not mean that IBM endorses the content or use of such Web site or

its owner. In addition, IBM is not a party to or responsible for any transactions you may enter into with third parties, even if you learn of such parties (or use a link to such parties) from an IBM site. Accordingly, you acknowledge and agree that IBM is not responsible for the availability of such external sites or resources, and is not responsible or liable for any content, services, products, or other materials on or available from those sites or resources. Any software provided by third parties is subject to the terms and conditions of the license that accompanies that software.

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information that has been exchanged, should contact:

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

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

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

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

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

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

COPYRIGHT LICENSE:

This information may contain sample application programs, in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

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

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

## Trademarks

The following terms are trademarks or registered trademarks of the International Business Machines Corporation in the United States, other countries, or both.

|                               |             |
|-------------------------------|-------------|
| pureXML                       | DB2 Connect |
| Redbooks                      | Informix    |
| developerWorks                | IBM         |
| DB2                           | REXX        |
| zSeries                       | AIX         |
| IBM Virtual Innovation Center | ibm.com     |
| POWER                         |             |

The following terms are trademarks or registered trademarks of other companies

- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- UNIX is a registered trademark of The Open Group in the United States and other countries.
- Microsoft, and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

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



---

# Index

## Special characters

- .NET
  - common language runtime
  - migrating routines 153

## Numerics

- 32-bit applications
  - migrating 147
- 32-bit external routines
  - migrating 155
- 32-bit Windows migration 65
- 64-bit instances
  - migrating 32-bit applications 147
  - migrating 32-bit external routines 155

## A

- ACTIVATE DATABASE command
  - post-migration tasks for DB2 servers 79
- adjusting log space
  - larger RID 78
- administrative routines
  - migration impact 128
- administrative views
  - migration impact 128
- ADO .NET applications
  - migrating 145
- APIs
  - back level 122
- applications
  - migrating 139
  - migrating XML Extender 74
  - migration impact
    - catalog view changes 128
    - DB2 command changes 124
    - SQL statements 127
    - system built-in routine changes 128
    - system-defined administrative routine and view changes 128
  - migration overview 115
  - migration support 117
  - planning migration 10
  - post-migration tasks
    - enabling new functionality 159
    - overview 157
    - removing deprecated features 157
    - tuning 157
  - pre-migration tasks
    - migrating clients 137
    - overview 137
    - testing 137
    - upgrading operating systems and development software 137
  - pre-migration tasks reviewing migration essentials 137
- ARRAY data type
  - enabling new functionality
    - post-migration tasks for applications and routines 159
- autonomic computing
  - enabling features after migration 21

- AWE buffer pools
  - discontinued functionality
  - migration impact 35

## B

- backing up
  - all database partitions in migrated databases 89
  - client configuration 101
  - databases
    - pre-migration tasks for DB2 servers 42
    - DB2 server configuration 43
- BACKUP DATABASE command
  - pre-migration tasks for DB2 servers 42
- books
  - printed
    - ordering 168
- built-in routines
  - migration impact 128

## C

- catalog views
  - migration impact 128
- changing raw devices to block devices (Linux)
  - pre-migration tasks for DB2 servers 46
- CLI applications
  - migrating 141
- clients
  - migrating 95, 97
    - Linux and UNIX 107
    - Windows 103, 105
  - migration best practices 98
  - migration support 97
  - planning migration 9
  - post-migration tasks
    - managing server changes 109
    - overview 109
    - recataloging nodes 109
    - verifying migration 110
  - pre-migration tasks 101
    - backing up configuration 101
    - migrating DB2 servers 101
    - migrating in test environments 102
    - reviewing migration essentials 101
- CLP (command line processor)
  - scripts
    - migrating 146
    - migration impact 124
- commands
  - dasmigr
    - migrating DAS 54, 61
  - db2ckmig command
    - pre-migration tasks for DB2 servers 40
  - db2exmig
    - post-migration tasks for DB2 servers 84
  - db2imigr
    - actions 17
    - migrating instances 52, 60
  - db2migrmigration support 18

- commands (*continued*)
  - db2tdbmgr
    - migrating DAS 54, 61
  - db2uiddl
    - converting unique indexes 82
  - MIGRATE DATABASE
    - migrated database entities 17
    - migrating databases 55, 63
    - migration support 18
- complex environments
  - migrating DB2 servers 65
- configuration backup
  - clients 101
  - pre-migration tasks for DB2 servers 43
- configuration parameters
  - migrating 80
  - migration impact 26
  - saving settings
    - pre-migration tasks for DB2 servers 43
- CREATE TABLESPACE statement
  - adjusting system temporary table spaces page sizes 84

## D

- dasmigr command
  - migrating DAS 54, 61
- Data Links Manager environment
  - migrating 73
- Data Server Client
  - migrating (Windows) 103
- Data Server Runtime Client
  - migrating (Windows) 105
- database applications
  - enabling new functionality 159
  - migration overview 115, 117
- database auditing
  - post-migration 81
- database duplicates
  - testing DB2 server migration 49
- databases
  - enabling new functionality after migration 89
  - migrating
    - procedure 55, 63
  - physical design characteristic changes 26
  - pre-migration tasks 40
  - setting up auditing
    - post-migration tasks for DB2 servers 81
- DB2 administration server (DAS)
  - migrating 54, 61
- DB2 Information Center
  - languages 169
  - updating 170
  - versions 169
  - viewing in different languages 169
- DB2 JDBC Type 2 Driver
  - migrating Java applications 144
- DB2 servers
  - behavior changes 26
  - migrating
    - 32-bit to 64-bit 65
    - alternate fix pack installations 70
    - best practices 21
    - databases 55, 63
    - DB2 administration server (DAS) 54, 61
    - instances 52, 60
    - Linux 59
    - multiple DB2 copies 70

- DB2 servers (*continued*)
  - migrating (*continued*)
    - new server 66
    - overview 15, 17
    - partitioned database environments 69
    - planning 7
    - reversing 91
    - support 17
    - UNIX 59
    - Windows 51
  - migration impact
    - behavior changes 26
    - deprecated functionality 35
    - discontinued functionality 35
    - registry variables 26
  - post-migration tasks
    - activating databases 79
    - activating services 79
    - adjusting log space 78
    - adjusting system temporary table space page sizes 84
    - converting type-1 indexes to type-2 indexes 82
    - managing server behavior changes 80
    - migrating explain tables 84
    - overview 77
    - rebinding packages 83
    - setting up database auditing 81
    - verifying migration 86
  - pre-migration tasks
    - backing up configuration 43
    - backing up databases 42
    - changing raw devices to block devices (Linux) 46
    - increasing log space 44
    - increasing table space sizes 44
    - migrating in test environments 48
    - overview 39
    - revoking EXECUTE privilege on PUBLIC 41
    - taking servers offline 49
    - verifying databases 40
- DB2 Spatial Extender
  - migrating 21, 65
- DB2\_USE\_DB2JCCT2\_JROUTINE variable
  - migrating Java routines 152
- db2batch command
  - verifying migration 86
- db2ckmig command
  - pre-migration tasks for DB2 servers 40
- db2exmig command
  - post-migration tasks for DB2 servers 84
- db2imigr command
  - actions 17
  - migrating instances 52, 60
  - migration support 18
- db2rbind command
  - post-migration tasks for DB2 servers 83
- db2support command
  - pre-migration tasks for DB2 servers 43
- db2tdbmgr command
  - migrating DAS 54, 61
- db2uiddl command
  - converting unique indexes 82
- db2undgp command
  - pre-migration tasks for DB2 servers 41
- deprecated functionality
  - migration impact 35
- Direct I/O (DIO)
  - changing raw devices to block devices (Linux) 46

- discontinued functionality
  - migration impact 35
- disks
  - space requirements for migrating DB2 servers 24
- documentation
  - overview 165
  - PDF 165
  - printed 165
  - terms and conditions of use 172

## E

- embedded SQL applications
  - migrating 140
- explain tables
  - migrating 84

## F

- FORTRAN applications
  - migrating 140

## G

- global variables
  - enabling new functionality
  - post-migration tasks for applications and routines 159

## H

- help
  - configuring language 169
  - SQL statements 168

## I

- IBM Data Server Driver for JDBC and SQLJ
  - migrating Java applications 142
- increasing log space
  - migrating DB2 servers 44
- instance type
  - migration support 18
- instances
  - 32-bit and 64-bit migration support 25
  - migrating 52, 60

## J

- Java
  - applications
    - migrating (DB2 JDBC Type 2 Driver) 144
    - migrating (IBM Data Server Driver for JDBC and SQLJ) 142
  - routines
    - migrating 152
- jdk\_path configuration parameter
  - routines
    - migrating 152

## L

- larger RID
  - adjusting log space 78
  - adjusting system temporary table space page sizes 84

- Linux
  - changing raw devices to block devices 46
  - migrating
    - clients 107
    - DB2 servers 59
- log file space requirements
  - migrating DB2 servers 24

## M

- managing database auditing
  - post-migration tasks for DB2 servers 81
- managing server changes
  - post-migration tasks for clients 109
- Microsoft Cluster Server (MSCS)
  - migrating 72
- Microsoft SQL Server
  - migrating 36
- MIGRATE DATABASE command
  - migrated database entities 17
  - migrating databases 55, 63
  - migration support 18
- migrated databases
  - enabling new functionality 89
- migrating
  - .NET CLR routines 153
  - 32-bit and 64-bit support 25
  - 32-bit applications 147
  - applications 139
    - ADO .NET 145
    - C, C++, COBOL, Fortran, or REXX 140
    - catalog view changes 128
    - DB2 CLI 141
    - DB2 command changes 124
    - DB2 Version 9.5 3
    - embedded SQL 140
    - Java using DB2 JDBC Type 2 driver 144
    - Java using IBM Data Server Driver for JDBC and SQLJ 142
    - overview 115
    - planning 10
    - post-migration tasks 157
    - pre-migration tasks 137
    - SQL statement changes 127
    - support 117
    - system built-in routine changes 128
    - system-defined administrative routine and view changes 128
    - XML Extender 74
  - clients 95
    - DB2 Version 9.5 3
    - Linux and UNIX 107
    - planning 9
    - post-migration tasks 109
    - pre-migration tasks 101
    - test environment 102
  - Data Links 73
  - databases 55, 63
  - DB2 Administration Server (DAS) 54, 61
  - DB2 environment 3
  - DB2 server performance 21
  - DB2 servers 15, 17, 21
    - 32-bit to 64-bit Windows 65
    - adjusting log space 78
    - alternate fix pack installations 70
    - complex environments 65
    - configuration parameters changes 26

- migrating (*continued*)
  - DB2 servers (*continued*)
    - configuration parameters, registry variables and physical characteristics 80
    - creating database duplicates for test environments 49
    - database physical characteristics changes 26
    - DB2 Version 9.5 3
    - discontinued features 18
    - Linux and UNIX 59
    - log space and table space requirements 24
    - multiple DB2 copies 70
    - new server 66
    - partitioned database environments 69
    - planning 7
    - post-migration tasks 77
    - pre-migration tasks 39
    - registry variables changes 26
    - restrictions 18
    - reversing migration 91
    - taking servers offline 49
    - test environments 48
    - Windows 51
  - DB2 Spatial Extender 21
  - DB2 Version 9.5 3
  - enabling autonomic computing features 21
  - explain tables 84
  - HADR 18
  - instances 52, 60
    - 32-bit and 64-bit migration support 25
  - Microsoft Cluster Server (MSCS) 72
  - Microsoft SQL server 36
  - non-DB2 relational databases 36
  - NSE 73
  - Oracle 36
  - routines 149
    - 32-bit external routines 155
    - C, C++, and COBOL 150
    - DB2 Version 9.5 3
    - Java 152
    - overview 115
    - planning 10
    - post-migration tasks 157
    - pre-migration tasks 137
    - revoking the EXECUTE privilege on PUBLIC 41
    - SQL procedures 154
    - support 135
  - scripts 146
    - support 117
  - SQL replication environments 21
  - support
    - clients 97
    - DB2 servers 17
  - Sybase 36
  - tools catalog database 54, 61
  - Windows
    - Data Server Client 103
    - Data Server Runtime Client 105
  - XML data type 74
  - XML Extender 74
- migration
  - C, C++, and COBOL applications 140
  - C, C++, and COBOL routines 150
  - important references 163
  - overview 3
  - planning 5
    - applications 10
    - clients 9

- migration (*continued*)
  - planning (*continued*)
    - DB2 environments 5
    - DB2 servers 7
    - routines 10
  - support
    - 32-bit and 64-bit instances 18
    - applications 117
    - routines 135
    - scripts 117
- migration best practices
  - clients 98
  - DB2 servers 21
- Migration guide
  - description v
  - migrating applications and routines 113
  - migrating clients 93
  - migrating DB2 environments 1
  - migrating DB2 servers 13
- migration portal
  - additional resources 5
- migration support
  - 32-bit and 64-bit 25
  - instance type 18
- multiple DB2 copies
  - migrating DB2 servers 70

## N

- Net Search Extender (NSE)
  - migrating 65, 73
- NetBIOS
  - discontinued functionality
    - post-migration tasks for clients 109
- Network Information Services (NIS)
  - deprecated functionality 35
- new server
  - migrating DB2 servers 66
- non-buffered I/O operations
  - migrated databases 89
- notices 175

## O

- O\_DIRECT
  - changing raw devices to block devices (Linux) 46
- optimistic locking
  - enabling new functionality
    - post-migration tasks for applications and routines 159
- optimizer enhancements
  - enabling new functionality
    - post-migration tasks for applications and routines 159
- Oracle
  - migrating 36
- ordering DB2 books 168

## P

- partitioned database environments
  - migrating 69
- post-migration tasks
  - applications
    - enabling new functionality 159
    - removing deprecated features 157
    - tuning 157

- post-migration tasks (*continued*)
  - clients
    - managing server changes 109
    - overview 109
    - recataloging nodes 109
    - verifying migration 110
  - DB2 servers
    - activating databases 79
    - activating services 79
    - adjusting log spaces 78
    - adjusting system temporary table space page sizes 84
    - converting type-1 indexes to type-2 indexes 82
    - enabling new functionality 89
    - managing behavior changes 80
    - migrating explain tables 84
    - overview 77
    - re-creating write-to-table event monitors 86
    - rebinding packages 83
    - setting up database auditing 81
    - verifying migration 86
  - routines
    - enabling new functionality 159
    - removing deprecated features 157
    - tuning 157
- pre-migration tasks
  - applications
    - overview 137
  - clients
    - backing up configuration 101
    - migrating in test environments 102
    - overview 101
  - DB2 servers
    - backing up configuration 43
    - backing up databases 42
    - changing raw devices to block devices (Linux) 46
    - increasing log space 44
    - migrating in test environments 48
    - overview 39
    - revoking EXECUTE privilege on PUBLIC 41
    - taking servers offline 49
    - verifying databases are ready to migrate 40
  - routines
    - overview 137
- problem determination
  - information available 172
  - tutorials 172

## Q

- Query Patroller
  - migration 65

## R

- raw I/O
  - changing raw devices to block devices (Linux) 46
- raw logs
  - deprecated functionality
    - migration impact 35
- re-creating write-to-table event monitors
  - post-migration tasks for DB servers 86
- real-time statistics
  - migrated databases 89
- REBIND command
  - post-migration tasks for DB2 servers 83

- rebinding
  - packages
    - post-migration tasks for DB2 servers 83
- recataloging nodes
  - NetBIOS and SNA protocol
    - post-migration tasks for clients 109
- references
  - migration 163
- registry variables
  - migrating 80
  - migration impact 26
  - saving settings
    - pre-migration tasks for DB2 servers 43
- removing deprecated features
  - post-migration tasks 157
- REORG INDEXES command
  - converting type-1 indexes to type-2 indexes 82
- RESTORE DATABASE command
  - migrating DB2 servers 66
- reversing migration
  - DB2 servers 91
- reviewing migration essentials
  - pre-migration tasks for applications and routines 137
- revoking the EXECUTE privilege on PUBLIC
  - pre-migration tasks for DB2 servers 41
- REXX applications
  - migrating 140
- routines
  - migrating 149
    - C, C++, and COBOL 150
    - Java 152
    - revoking the EXECUTE privilege on PUBLIC 41
  - migration overview 115
  - migration support 135
  - planning migration 10
  - post-migration tasks
    - enabling new functionality 159
    - overview 157
    - removing deprecated features 157
    - tuning 157
  - pre-migration tasks
    - overview 137
    - testing 137
    - upgrading development software 137
    - upgrading operating systems 137
  - pre-migration tasks reviewing migration essentials 137

## S

- scenarios
  - migrating 65
- scripts
  - migrating 146
  - migration impact 124, 127
  - migration support 117
- security administrator authority (SECADM)
  - setting up database auditing
    - post-migration tasks for DB2 servers 81
- SNA (Systems Network Architecture)
  - discontinued functionality
    - post-migration tasks for clients 109
- SQL Administrative views and routines
  - migrating 146
- SQL procedures
  - migrating 154
- SQL replication environments
  - migrating 21

- SQL statements
  - displaying help 168
  - migrating 146
  - migration impact 127
- stored procedures
  - migrating 149
  - migration support 135
- Sybase
  - migrating 36
- system built-in routines
  - migration impact 128
- system catalog views
  - migration impact 128
- system commands
  - scripts
    - migrating 146
    - migration impact 124
- system temporary table spaces
  - page sizes
    - post-migration tasks for DB2 servers 84
- system-defined administrative routines
  - migration impact 128
- system-defined administrative views
  - migration impact 128

**T**

- table space requirements
  - migrating DB2 servers 24
- taking servers offline
  - pre-migration tasks 49
- TCP/IP
  - recataloging nodes
    - post-migration tasks for clients 109
- terms and conditions
  - use of publications 172
- test environments
  - migrating clients 102
  - migrating DB2 servers 48
    - creating database duplicates 49
- tools catalog database
  - migrating 54, 61
- troubleshooting
  - online information 172
  - tutorials 172
- tuning
  - applications
    - post-migration tasks 157
  - routines
    - post-migration tasks 157
- tutorials
  - problem determination 172
  - troubleshooting 172
  - Visual Explain 171

**U**

- UNIX
  - migrating
    - clients 107
    - DB2 servers 59
- updates
  - DB2 Information Center 170
- upgrading development software
  - pre-migration tasks for applications and routines 137

- upgrading operating system
  - pre-migration tasks for applications and routines 137
- user-defined functions
  - migrating 149
  - migration support 135

## V

- verifying migration
  - post-migration tasks for clients 110
  - post-migration tasks for DB2 servers 86
- Visual Explain
  - tutorial 171

## W

- Web sites
  - DB2 Migrate Now! 36
  - developerWorks - Information Management 36
  - IBM Virtual Innovation Center 36
  - migration portal 5
- Windows operating systems
  - migrating
    - Data Server Client 103
    - Data Server Runtime Client 105
    - DB2 servers 51
- workload management
  - migrated databases 89
- write-to-table event monitors
  - re-creating after migration 86

## X

- XML collections (XML Extender)
  - migrating applications 74
- XML data type
  - migrating applications 74





Printed in USA

GC23-5859-01



Spine information:

DB2 Version 9.5 for Linux, UNIX, and Windows

Migration Guide

