

IBM DB2 Universal Database
Extended Enterprise Edition
for Windows NT**



Quick Beginnings

Version 5.2

IBM DB2 Universal Database
Extended Enterprise Edition
for Windows NT**



Quick Beginnings

Version 5.2

Before using this information and the product it supports, be sure to read the general information under Appendix E, "Notices" on page 239.

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.

Order publications through your IBM representative or the IBM branch office serving your locality or by calling 1-800-879-2755 in U.S. or 1-800-IBM-4YOU in Canada.

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 1993, 1998. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Welcome to DB2 Universal Database	ix
Conventions	ix
Road Map	x

Part 1. Introduction to DB2 Universal Database 1

Chapter 1. About DB2 Universal Database	3
Working with Data Using DB2 Universal Database	5
Accessing DB2 Data from Remote Clients	5
Accessing Multiple DB2 Servers	6
Administering Databases with DB2 Tools	7
Managing Databases Using the Control Center	8
Understanding the Administration Server	10
Monitoring Databases Using DB2 Performance Monitor	10
Viewing SQL Access Plans Using Visual Explain	11
Managing Connections to Databases Using the Client Configuration Assistant	11
Developing Applications Using the DB2 Software Developer's Kit	12
Accessing Host Data from the Desktop Using DB2 Connect Enterprise Edition	13
Accessing DB2 Data from the Web	15
Chapter 2. Introduction to DB2 Extended Enterprise Edition	19
Processing in a Partitioned Database Environment	20
Cost-Based Query Optimization	20
Configuration	21
Machines and Storage	23
Nodegroups and Data Partitioning	23
Multiple Logical Nodes	25
Instances	26
Fast Communication Manager	27
Agents	28

Part 2. Installing DB2 Server 29

Chapter 3. Planning for Installation	31
Memory Requirements	31
Configuration for Accessing DB2 Servers on the LAN	31
Configuration for Accessing LAN and Host Data	33
Configuration for Accessing from the Web	35
Disk Requirements	35
Estimating Fixed Disk Requirements	35
Software Requirements	37
Server Product Requirements	37

Client Product Requirements	38
Possible Client-to-Server Connectivity Scenarios	40
Understanding System Administrative Authority	41
Chapter 4. Installing and Configuring DB2 Extended Enterprise Edition	43
Before You Begin	43
Pre-Installation Requirements	43
Determining User Accounts for DB2 Installation and Configuration	44
Performing the Installation	46
Installing DB2 Extended Enterprise Edition	46
Verifying the Installation	50
Creating a Server Access Profile	51

Part 3. Installing Clients 53

Chapter 5. Installing DB2 Clients	55
Chapter 6. Installing DB2 Clients on Windows 32-Bit Operating Systems	57
Chapter 7. Installing DB2 Clients on OS/2 Operating Systems	61
Chapter 8. Installing DB2 Clients on Windows 3.x Operating Systems	63

Part 4. Configuring DB2 Communications 67

Chapter 9. Configuring Client-to-Server Communications Using the Client Configuration Assistant	69
Verifying the Connection	72

Part 5. Using DB2 Universal Database 75

Chapter 10. Getting Started with DB2 Universal Database	77
Logging on to the System	77
Creating the Sample Database	78
Connecting to a Database	79
Working with the System Administrative Group	80
Granting Users Authorization	80
Software Registration	81
Upgrading from a Try-and-Buy License	82
Chapter 11. Working with Instances	83
Instance Directory	83
Creating Instances	84
Listing Instances	85
Listing Database Partition Servers in an Instance	86
Starting and Stopping an Instance	86

Adding a Database Partition Server to an Instance	88
Changing the Database Partition Server Configuration	89
Dropping a Database Partition Server from an Instance	91
Setting the Current Instance	92
Running Multiple Instances Concurrently	93
Removing Instances	93
Chapter 12. Using the Administration Server	95
Creating an Administration Server	95
Setting Up the Administration Server to Use the CCA and the Control Center	97
Update Instance Lists	97
Update the Administration Server Configuration	98
Starting the Administration Server	99
Stopping the Administration Server	99
Setting the Logon Account for the Administration Server	99
Removing the Administration Server	100
Chapter 13. Running Your Own Applications	101
Binding Database Utilities	101
Running CLI/ODBC Programs	103
Platform Specific Details for CLI/ODBC Access	104
Detailed Configuration Information	108
Running Java Programs	109
Configuring the Environment	110
Java Applications	111
Java Applets	112
Chapter 14. Entering DB2 Commands and SQL Statements	113
Using the Command Center	113
Using the Command Line Processor	114
Interactive Input Mode	115
Command Line Mode	116
Connecting to a Logical Node	117
Chapter 15. Controlling Your DB2 Environment	119
Controlling the DB2 Profile Registry	120
Setting Your Environment on OS/2	121
Setting Your Environment on Windows 32-Bit Operating Systems	122
Setting Your Environment on Windows 3.x	123
DB2 Registry Values and Environment Variables	124
Configuration Parameters	130
Setting Database Manager Configuration Parameters on a Client Instance	130
Chapter 16. Using the Windows NT Performance Monitor	133
Registering DB2 with the Windows NT Performance Monitor	133
Enabling Remote Access to DB2 Performance Information	134
Deregistering the Username for the DB2 Performance Monitor	135
Displaying DB2 Performance Values	135

Accessing Remote DB2 Performance Information	136
Resetting DB2 Performance Values	136

Part 6. Advanced Installation and Configuration 139

Chapter 17. Migrating from Previous Versions and Releases	141
Migrating from Previous Versions of DB2	141
Upgrading from DB2 Release 5.0 to Release 5.2	142
Pre-Installation Steps	142
Step 1. Back Up Databases	142
Step 2. Verify that Databases Can Be Migrated	143
Installing DB2 Extended Enterprise Edition Version 5	147
Post-Installation Steps	150
Migrating Databases	151
Post Database Migration Actions	152
Updating Catalog Table Data	156
Chapter 18. Using the Client Configuration Assistant	159
Configuring Database Connections	159
Using an Access Profile	160
Searching the Network for Databases	163
Manually Configure a Connection to a DB2 Database	168
Chapter 19. Setting up Communications on the Server Using the Command Line Processor	171
Setting the DB2COMM Registry Value	171
Configuring Named Pipes on the Server	173
Configuring TCP/IP on the Server	173
Step 1. Identify and Record Parameter Values	174
Step 2. Configure the Server	175
Configuring NetBIOS on the Server	177
Step 1. Identify and Record Parameter Values	177
Step 2. Configure the Server	178
Autostarting DB2 with NetBIOS	180
Configuring IPX/SPX on the Server	181
Step 1. Identify and Record Parameter Values	182
Step 2. Configure the Server	182
Configuring APPC on the Server	184
Step 1. Identify and Record Parameter Values	184
Step 2. Configure the Server	184
Chapter 20. Unattended Installation	187
Installing DB2 Products from a Remote Server Hard Disk or CD-ROM	187
Before You Begin	188
Step 1. Make DB2 Files Available for Installation	188
Step 2. Create a Response File for Unattended Installation	189
Step 3. Run the Setup Program from the Workstation	197
Installing DB2 Remotely Using SMS	198

Before You Begin	199
Step 1. Make DB2 Files Available for Installation	199
Step 2. Create a Response File for Unattended Installation	199
Step 3. Import the DB2 Install Files into SMS	200
Step 4. Distribute the DB2 Installation Package Using SMS	200
Step 5. Run the Installation on the SMS Client	201

Chapter 21. Accessing DB2 Universal Database Servers from Host and AS/400 Applications	203
Supported Clients	203
PTFs Required	204
Configuration Steps for DB2 Universal Database Server	204
Configuring DB2 Universal Database Servers for Host or AS/400 Client Access	205
Using the DB2 Universal Database Server from Host or AS/400 Clients	206
Authentication	207
Troubleshooting	207
DRDA Functions Supported	207
Security and Auditability	213
Configuration Considerations	214

Part 7. Appendixes 215

Appendix A. How the DB2 Library Is Structured	217
SmartGuides	217
Online Help	218
DB2 Books	220
Viewing Online Books	224
Searching Online Books	225
Printing the PostScript Books	225
Ordering the Printed DB2 Books	226
Information Center	227

Appendix B. Installation Troubleshooting 229

Appendix C. National Language Support (NLS)	231
Code Page and Language Support	231

Appendix D. Naming Rules	233
General Naming Rules	233
Database, Database Alias, and Catalog Node Names	234
Object Names	234
Usernames, Group Names, and Instance Names	235
Workstation Names (nname)	236
DB2SYSTEM Naming Rules	237
Password Rules	237

Appendix E. Notices	239
Trademarks	240

Trademarks of Other Companies	240
Appendix F. Contacting IBM	243
Index	245

Welcome to DB2 Universal Database

This book describes how to install and use the basic functions of DB2 Universal Database Extended Enterprise Edition.

Part 1 gives an overview of DB2 Universal Database Extended Enterprise Edition.

Part 2 provides planning, installing, and basic configuration information for DB2 Extended Enterprise Edition.

Part 3 describes how to install a DB2 Client Application Enabler on OS/2 and Windows operating systems.

Part 4 provides information on configuring communications between DB2 servers and clients.

Part 5 describes how to use DB2 Universal Database.

Part 6 describes how to perform advanced configuration tasks, such as performing administration tasks using the Client Configuration Assistant, using the command line processor for setting up server communications, and using the DB2 workstation as a server for host applications.

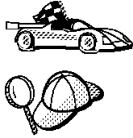
Part 7 contains reference information and describes viewing, printing, and ordering manuals.



Conventions

This book uses these highlighting conventions:

- **Boldface type** indicates commands or graphical user interface (GUI) controls such as names of fields, folders, icons, or menu choices.
- *Italics* indicates variables that you should replace with a value. It is also used to indicate file names, directory paths, book titles and to emphasize words.
- Monospace indicates examples of text you enter exactly as shown.



This is a fast path icon. It tells you if you can skip sections to get to the "how to" information. It is shown to indicate where you should go next.



This icon marks a tip. It provides additional information that can help you complete a task.

Road Map



Use the following table to find the information you need quickly.

If you want to...	Refer to...
Understand the relationship between DB2 and related products,	Chapter 1, "About DB2 Universal Database" on page 3.
Plan your network configuration,	Chapter 3, "Planning for Installation" on page 31.
Confirm that your system meets the DB2 disk and software prerequisites,	"Disk Requirements" on page 35 and "Software Requirements" on page 37.
Install DB2 Extended Enterprise Edition from a CD-ROM,	Chapter 4, "Installing and Configuring DB2 Extended Enterprise Edition" on page 43.
Install the client products,	Chapter 5, "Installing DB2 Clients" on page 55.
Set up communications on the client to access databases,	Chapter 9, "Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 69.
Use the DB2 components,	Chapter 10, "Getting Started with DB2 Universal Database" on page 77.
Create the sample database,	"Creating the Sample Database" on page 78.
Connect to a database,	"Connecting to a Database" on page 79.
Set up a DB2 Call Level Interface or ODBC environment to enable ODBC applications such as Lotus Approach to access DB2 data,	"Running CLI/ODBC Programs" on page 103.
Use the DB2 Command Center or the DB2 command line processor to enter SQL statements or DB2 commands interactively,	Chapter 14, "Entering DB2 Commands and SQL Statements" on page 113.
Print or view online documentation,	"Printing the PostScript Books" on page 225 or "Viewing Online Books" on page 224.
Learn about last-minute changes to the product,	The Installation Notes on the CD-ROM or the Release Notes that are installed with a DB2 product.

To locate information on other topics, see Appendix A, “How the DB2 Library Is Structured” on page 217 for a complete description of the DB2 library.

Part 1. Introduction to DB2 Universal Database

Chapter 1. About DB2 Universal Database

DB2 is a *relational* database management system that is web-enabled with Java support; scalable from single processors to clusters of symmetric multiprocessors; and multimedia capable with image, audio, video, and text support.

Your CD-ROM includes *DB2 Extended Enterprise Edition* and a *DB2 Client Application Enabler*.

DB2 Universal Database Extended Enterprise Edition

A relational database management system that enables local and remote client applications to create, update, control, and manage relational databases using Structured Query Language (SQL), ODBC, JDBC, or CLI. Its main feature is the ability for a database to be partitioned across multiple independent machines of a common platform. To the end-user and application developer, it still appears as a single database on a single machine. This fully scalable database system enables an application to use a database that is simply too large for a single machine to handle efficiently. SQL operations and utilities can execute in parallel both within and between the individual database partitions, thereby speeding up the execution time of a single query or command. This product also includes the DB2 Connect feature that allows access to databases on AS/400, MVS/ESA, OS/390, VM, or VSE systems. DB2 Extended Enterprise Edition was formerly known as DB2 Parallel Edition.

DB2 Client Application Enabler Component

Provides the ability for a client workstation to access a DB2 server. It is a component that is provided on your CD-ROM.

There are several other DB2 products, which you can order separately:

DB2 Universal Database Enterprise Edition

A relational database management system that enables local and remote client applications to create, update, control, and manage relational databases using Structured Query Language (SQL), ODBC, JDBC, or CLI. This product also includes support for host and AS/400 connectivity. This product provides its users with access to DB2 databases that reside on host systems such as MVS/ESA, OS/390, AS/400, VM, and VSE. This product is available for AIX, HP-UX, OS/2, SCO UnixWare 7, Solaris and Windows NT.

DB2 Universal Database Workgroup Edition

A relational database management system that enables local and remote client applications to create, update, control, and manage relational databases using Structured Query Language (SQL), ODBC, JDBC, or CLI.

This product is available for OS/2, Windows NT, and SCO UnixWare 7 only.

DB2 Connect Enterprise Edition

Provides access from clients on a network to DB2 databases residing on host systems such as AS/400, MVS/ESA, OS/390, VM, and VSE. DB2 Connect Enterprise Edition is

available for the AIX, HP-UX, OS/2, Solaris, SCO UnixWare 7, and Windows NT. This product was formerly known as DDCS Multi-User Gateway.

DB2 Application Developer's Kit

Contains a collection of DB2 Universal Database products, clients, DB2 Connect products, DB2 Software Developer's Kits, and application development tools for all supported operating systems.

The Application Developer's Kit gives you all the tools you need to create multimedia database applications that can run on a variety of platforms and can connect to any DB2 server, including DataJoiner.

DB2 Universal Database Personal Edition

Allows you to create and use local databases. DB2 Personal Edition can also act as client to access remote DB2 servers. This product is available for the OS/2 and Windows 32-bit operating systems only.

DB2 Connect Personal Edition

Provides access from a single workstation to DB2 databases residing on host systems such as MVS/ESA, OS/390, OS/400, VM, and VSE, as well as access to DB2 databases. DB2 Connect Personal Edition is available for the OS/2, Windows 3.x, and Windows 32-bit operating systems. This product was formerly known as DDCS Single-User.

DB2 Client Application Enablers

Contains all the latest DB2 Client Application Enablers. This is not a separately orderable product, but it is available with DB2 products.

DataJoiner

Provides access from DB2 clients on the network to databases residing on various heterogeneous data sources such as DB2, Informix, Microsoft SQL Server, Oracle, and Sybase.

DB2 File Manager

Manages files on a file server that are pointed to, and managed by, DATALINK values in a DB2 database. Use the DB2 File Manager to control access to files that are external to a DB2 database.

This product is available for the AIX operating system.

In addition to the DB2 products listed above, we provide DB2 Client Application Enablers CD-ROMs in all the DB2 products packages except the Personal Edition versions. These CD-ROMs contain all the latest DB2 Client Application Enablers. With DB2 Client Application Enablers, clients from a variety of platforms can connect to DB2 databases.



DB2 Workgroup Edition, DB2 Enterprise Edition, DB2 Personal Edition, and DB2 Extended Enterprise Edition are commonly referred to as DB2 server or DB2 Universal Database throughout this book.

Working with Data Using DB2 Universal Database

As well as providing a relational database to store your data, DB2 lets you administer requests to query, update, insert, or delete data using local or remote client applications.

DB2 Universal Database includes tools that enable you to tune performance, access remote DB2 servers, manage servers from a single point of control, develop powerful applications, and process SQL queries. For a description of these tools, see “Administering Databases with DB2 Tools” on page 7.

Accessing DB2 Data from Remote Clients

DB2 Client Application Enablers provide a run-time environment that enables client applications to access one or more remote databases. Local applications and all Java applications (either local or remote) access a database through a DB2 Client Application Enabler component on the server. All remote applications that are not Java applets must have a DB2 Client Application Enabler installed on the client machine before they can access the remote database.

The *DB2 Client Application Enablers CD-ROMs* provide support for clients on the following operating systems:

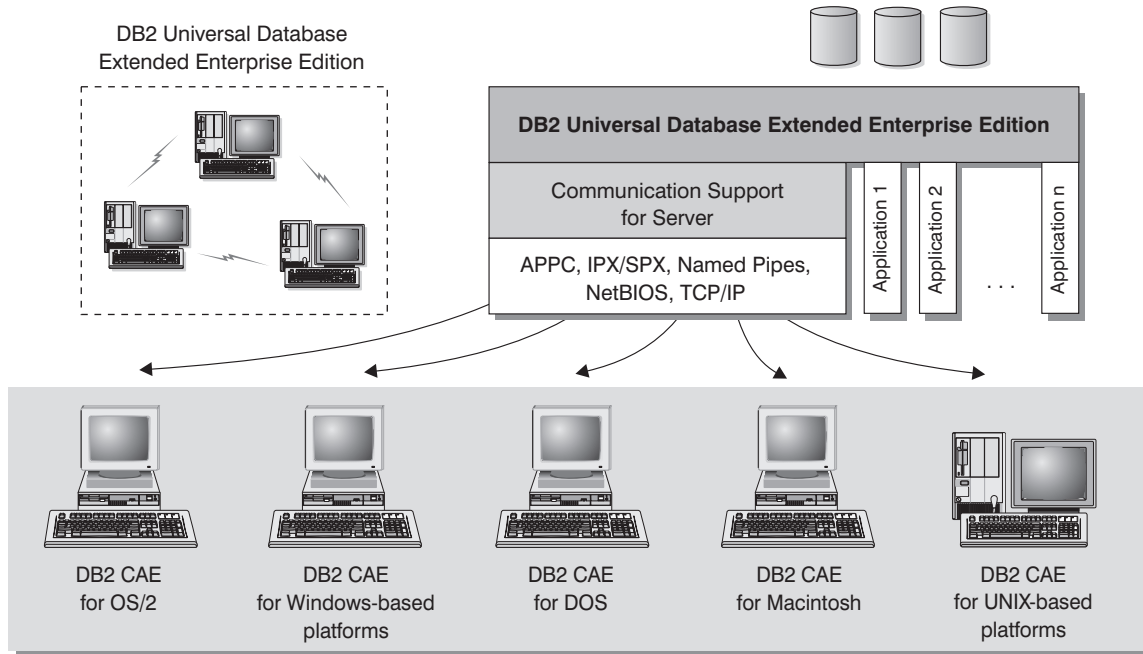
- Macintosh
- OS/2
- UNIX (AIX, HP-UX, SCO OpenServer, SCO UnixWare 7, SGI IRIX, SINIX, Solaris)¹
- Windows 3.x, and Windows 32-bit operating systems

The Client Application Enabler Version 1.2 for DOS is only available from the Web. See Chapter 5, “Installing DB2 Clients” on page 55 for information on how to download the DOS client.

Figure 1 on page 6 shows a DB2 server with local applications and applications that run on remote clients. This is the system that contains your databases. Any applications that run on this system are known as local applications. The client systems need to have the appropriate DB2 Client Application Enabler installed to enable applications to access the data on the remote server system.

¹ Client support on SCO OpenServer and SINIX is at a Version 2 level only.

DB2 Universal Database Extended Enterprise Edition - Remote Client Support



NOTE: CAE is Client Application Enabler. Not all protocols are supported for all the clients.

Figure 1. DB2 with Local Applications and Remote Clients

In addition to accepting requests from the clients shown in Figure 1, DB2 for OS/390, DB2 for AS/400, and DB2 for VSE & VM clients can access data on DB2 Universal Database Workgroup Edition, Enterprise Edition, and Extended Enterprise Edition servers. See Chapter 21, "Accessing DB2 Universal Database Servers from Host and AS/400 Applications" on page 203, or refer to the *Administration Guide* for more information.

Accessing Multiple DB2 Servers

Once a network is operational and protocols are functional on all workstations, LAN-to-LAN connections between DB2 servers and clients require no additional software.

For example, you can have a DB2 server that is connected to a LAN, and a DB2 Extended Enterprise Edition server connected to a LAN. As long as there is a connection between the two LANs, clients on either network can access either server. See Figure 2 on page 7.

Communicating Between DB2 Universal Database Servers

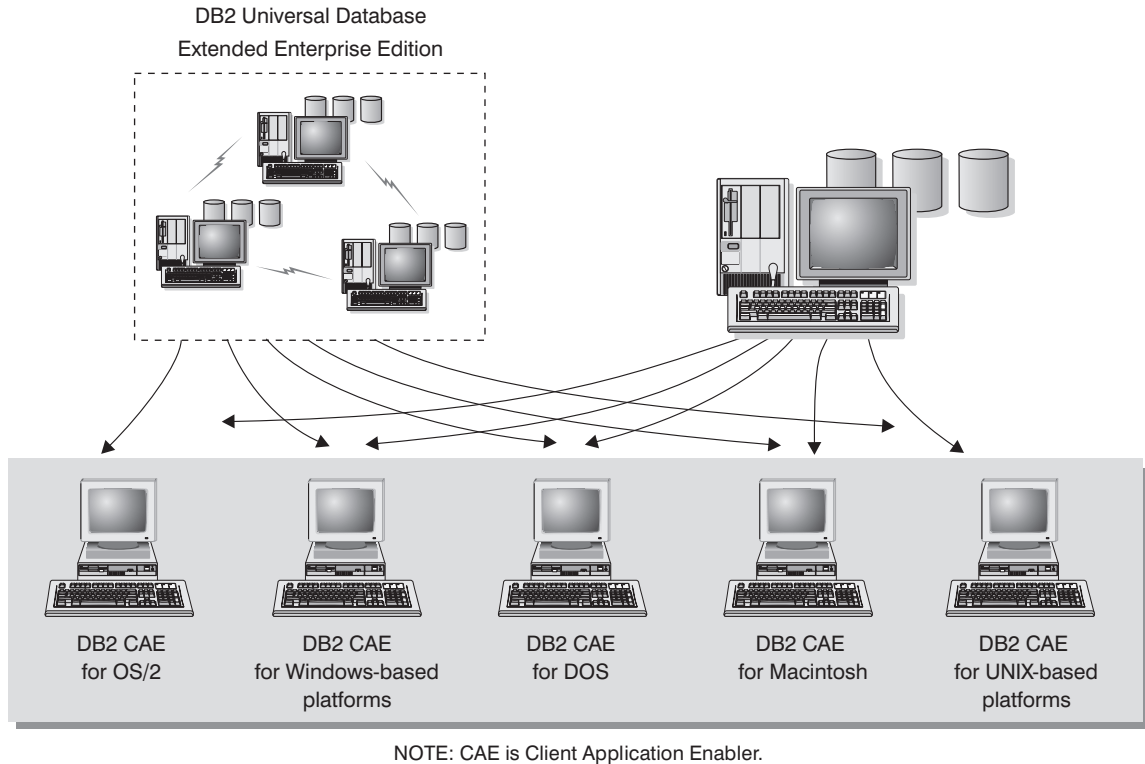


Figure 2. Accessing Data on Multiple Servers

Within a single transaction, databases on both servers are accessed and updated, and the integrity of the data on both servers is maintained. This is commonly known as two-phase commit, or distributed-unit-of-work access. Refer to the *Administration Guide* for more information.

Administering Databases with DB2 Tools

Use the *Control Center* to perform administration tasks such as configuring DB2 instances and databases, backing up and recovering data, scheduling jobs, and managing media, all from a graphical interface.

The Control Center uses TCP/IP to retrieve status, connection, and snapshot information from each database partition server (node). You assign a TCP/IP port number for the Performance Monitor component of the Control Center during installation.

Managing Databases Using the Control Center

The Control Center displays database objects (such as database partitions, table partitions, and packages) and their relationships to each other. Using the Control Center, you can manage a local database server or multiple remote DB2 servers and the database objects within them, all from a single point of control. See Figure 3 for an example of the main Control Center window. You can choose to install the Control Center on OS/2 or Windows 32-bit operating systems.

The Control Center distinguishes between single-partition and multipartition database systems via *Discovery*. Discovery uses the *DB2SYSTEM*, *DB2ADMINSERVER*, and *DB2COMM* registry values. For more information on these registry values, see Chapter 15, “Controlling Your DB2 Environment” on page 119.

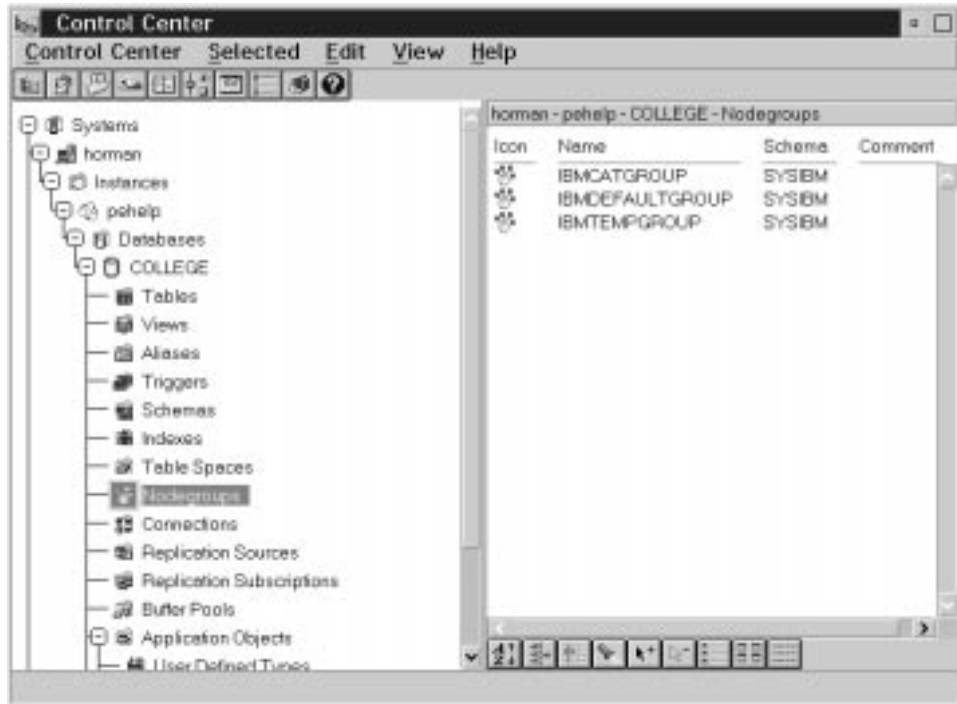


Figure 3. Control Center Main Window

From the Control Center, you can perform operations on database objects. These operations include:

- Create and drop a database
- Create, alter, and drop a table space or table
- Create, alter, and drop an index
- Create nodegroups
- Backup and recover a database partition or table space partition.
- Define the replication sources and subscriptions to replicate data between systems

For more information on objects in a partitioned database system, see Chapter 2, “Introduction to DB2 Extended Enterprise Edition” on page 19.

The Control Center provides additional features to assist you in managing your DB2 servers:



Use the **Command Center** to enter DB2 commands and SQL statements in an interactive window and see the execution result in a result window. You can scroll through the results and save the output to a file.



Use the **Script Center** to create scripts, which you can store and invoke at a later time. These scripts can contain DB2 commands, SQL statements, as well as operating system commands. Scripts can be scheduled to run unattended. These jobs can be run once or set up to run on a repeating schedule; a repeating schedule is particularly useful for tasks like backup.



Use the **Journal** to view all available information about jobs that are pending execution, executing, or that have completed execution. You can also view the recovery history log, the alerts log, and the messages log; and review the results of jobs that are run unattended.



Use the **Alert Center** to monitor your system for early warnings of potential problems or to automate actions to correct problems discovered.



Use the **Tools Setting** to change the settings for the Control Center, Alert Center, Replication, and Node Status.

You can run these functions from the Control Center toolbar or from icons in the Administration Tools folder.

You can install the following options for the Control Center:

DB2 Performance Monitor	Use the DB2 Performance Monitor to monitor the performance of your DB2 system. You can monitor activity using snapshots of data over a period of time or using data for a particular event. See “Monitoring Databases Using DB2 Performance Monitor” on page 10 for more information.
Visual Explain	Use Visual Explain to graphically analyze and tune SQL statements and to analyze query access plans. See “Viewing SQL Access Plans Using Visual Explain” on page 11 for more information.

You can find additional information in the *Administration Getting Started* guide or in the Control Center's online help.

Understanding the Administration Server

The Administration Server satisfies requests from the DB2 Administration Tools and the Client Configuration Assistant. It enables the DB2 Administration Tools to start, stop, catalog, and set values for the database manager configuration parameters. It is also used by the Client Configuration Assistant to catalog databases and node directories for the client.

The Administration Server must reside on every DB2 server that you want to administer and detect. The setup program creates the Administration Server on the instance-owning machine and automatically starts it at boot time. The default name of the Administration Server is DB2DAS00. For more information, see Chapter 12, "Using the Administration Server" on page 95.

Monitoring Databases Using DB2 Performance Monitor

With the DB2 Performance Monitor, you can:

- Identify and analyze performance problems in database applications or the database manager.
- Use the early warning system to detect potential problems.
- Automate actions to correct problems that are discovered.
- Define your own statistics, in addition to the default set that is provided.

You can choose to monitor snapshots or events. The Snapshot Monitor allows you to capture point-in-time information at specified intervals. The Event Monitor allows you to record information about an occurrence of an event such as a deadlock.

You can find additional information in the *Administration Getting Started* guide or in the online help for the DB2 Performance Monitor.



You are also able to use the Windows NT Performance Monitor to monitor both database and system performance. See Chapter 16, "Using the Windows NT Performance Monitor" on page 133 for details on how to register DB2 resources and to use the Windows NT Performance Monitor.

Snapshot Monitoring Considerations

The snapshot monitoring function in the Control Center is split between an administrative client and the database server. A separate process is started on behalf of the user to gather snapshot information on the server and return the data to the client. This separate process runs locally on the server and therefore it must be locally authenticated; because of this, the user must have a valid username and password on

each server that is monitored. If the server is configured for SERVER authentication, this is transparent to the user. If the server is configured for CLIENT authentication, the user will be prompted for a username and password that will be authenticated locally on the server.

Note: DCE authentication for snapshot monitoring is not supported at this time.

You can find additional information in the *Administration Getting Started* guide or in the online help for DB2 Snapshot Monitor.

Viewing SQL Access Plans Using Visual Explain

Visual Explain helps database administrators and application developers to:

- View the access plan chosen by the database manager's optimizer for a given SQL statement.
- Tune SQL statements for better performance.
- Design application programs and databases.
- View all the details of an access plan, including the statistics in the system catalogs.
- Decide whether or not to add an index to a table.
- Identify the source of problems by analyzing the access plan or performance of SQL statements.
- Use the portable snapshot function to view snapshots from any remote DB2 server.
- Display access plans for queries on all supported DB2 configurations.

You can find additional information in the *Administration Getting Started* guide or in the online help for Visual Explain.

Managing Connections to Databases Using the Client Configuration Assistant

The Client Configuration Assistant helps you manage your database connections to remote DB2 servers. The Client Configuration Assistant is available on OS/2 and Windows 32-bit operating systems.

With the Client Configuration Assistant, you can:

- Catalog databases so that they can be used by applications. Three methods are available:
 - Search the network for available databases and selecting one. Client access is automatically set up for that database.

- Use a database access profile provided by a database administrator to automatically define your connections. Client access is automatically set up for that database.
- Manually configure a connection to a database by entering a few required connection parameters.
- Remove cataloged databases, or change the properties of a cataloged database.
- Test connections to local or remote databases identified on your system.
- Bind applications to a database by selecting utilities or bind files from a list.
- Tune the client configuration parameters on your system. Parameters are logically grouped and suggested settings are provided on the interface as parameters are selected.
- Set up connections to DB2 for OS/390, DB2 for AS/400, DB2 for VSE & VM.

Developing Applications Using the DB2 Software Developer's Kit

The DB2 Software Developer's Kit is a collection of tools that are designed to meet the needs of database application developers. It includes libraries, header files, documented APIs, and sample programs to build character-based, multimedia, or object-oriented applications.

You can install the DB2 Software Developer's Kit on a DB2 server or on a remote workstation. If you install it on the server, it can access both remote and local databases.

There is a platform-specific version of the DB2 Software Developer's Kit available for each of the supported operating systems and it is available in the Application Development Kit. Applications that are developed with the DB2 Software Developer's Kit will run on any platform where the equivalent DB2 Client Application Enabler component is installed. Through a DB2 Client Application Enabler, these applications can access all DB2 servers and, by using DB2 Connect, they can also access DB2 for AS/400, DB2 for OS/390, and DB2 for VSE & VM database servers.

The DB2 Software Developer's Kit allows you to develop applications that use the following interfaces:

- Embedded SQL
- Call Level Interface (CLI) development environment (which is compatible with ODBC from Microsoft)
- Java Support (JDBC)
- Embedded SQL for Java (SQLJ)
- Application programming interfaces to access database utilities.

The DB2 Software Developer's Kit supports several programming languages (which include COBOL, C, and C++) for application development, and provides precompilers for the supported languages. Refer to the *Building Applications for UNIX Environments* or the *Building Applications for Windows and OS/2 Environments* manuals for complete information on using the DB2 Software Developer's Kit, and for a list of the supported compilers.

Accessing Host Data from the Desktop Using DB2 Connect Enterprise Edition

DB2 Connect Enterprise Edition gives clients on your LAN access to data that is stored on host systems. It provides applications with transparent access to host data through a standard architecture for managing distributed data. This standard is known as Distributed Relational Database Architecture (DRDA). DRDA allows your applications to establish a fast connection to databases on MVS, OS/390, OS/400, VM, and VSE host systems.

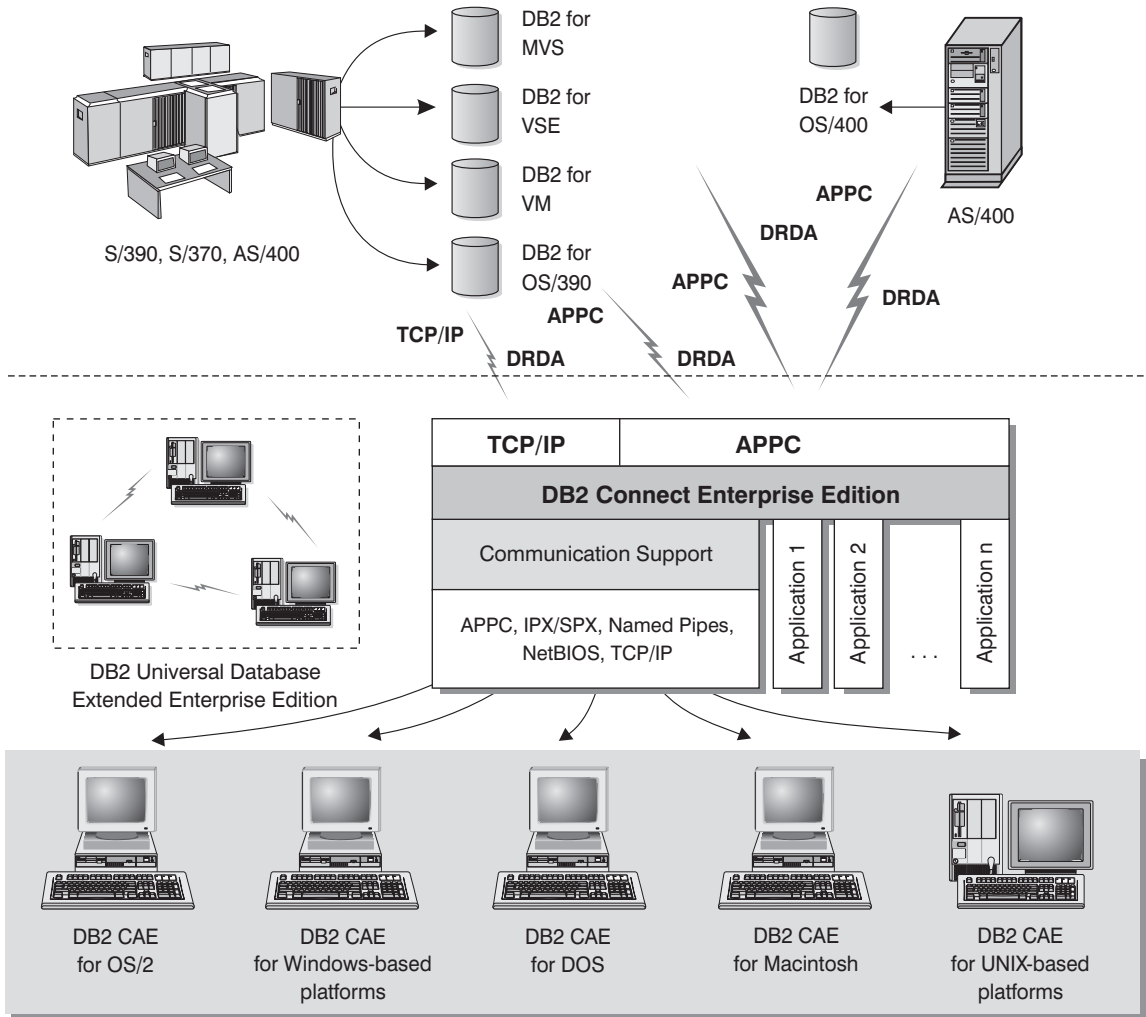
A great deal of the data in many large organizations is managed by DB2 for AS/400, DB2 for MVS/ESA, DB2 for OS/390, or DB2 for VSE & VM. Applications that run on any of the supported platforms can work with this data transparently, as if a local database server managed it.

The following tools and products can access host data easily using DB2 Connect:

- *Spreadsheets*, such as Lotus 1-2-3 and Microsoft Excel, to analyze real-time data without the cost and complexity of data extraction procedures and import procedures.
- *Decision support tools*, such as Business Objects, Intersolv Q+E Database Editor, and Crystal Reports, to provide real-time information.
- *Personal database products*, such as Lotus Approach and Microsoft Access.
- *Development tools*, such as PowerBuilder, Microsoft VisualBasic, and Borland Delphi to create client/server solutions.

DB2 Connect Enterprise Edition enables multiple clients to connect to host data and can significantly reduce the effort that is required to establish and maintain access to enterprise data. See Figure 4 on page 14 for an example of clients connecting to host databases through DB2 Connect Enterprise Edition.

DB2 Connect Enterprise Edition



NOTE: CAE is Client Application Enabler. Not all protocols are supported for all the clients.

Figure 4. DB2 Connect Enterprise Edition

Accessing DB2 Data from the Web

Java Support (JDBC), Embedded SQL for Java (SQLJ), and Net.Data are provided with DB2 to allow you to create applications that access data in DB2 databases from the Web.

Use *Java Support* to create applications or applets that access data in DB2 databases. You can run JDBC applets inside Hypertext Markup Language (HTML) web pages on any system with a Java-enabled browser, regardless of the platform of your client. Your client system requires no additional software beyond this browser. The client and the server share the processing of JDBC applets and applications.

The JDBC server and the DB2 Client Application Enabler must reside on the same machine as the Web server. The JDBC server calls the DB2 Client Application Enabler to connect to local, remote, or host databases. When the applet requests a connection to a DB2 database, the JDBC client opens a TCP/IP socket to the JDBC server on the machine where the Web server is running. See Figure 5 for an example of Java-enabled browsers accessing data from remote DB2 databases.

Accessing DB2 Data Using JDBC

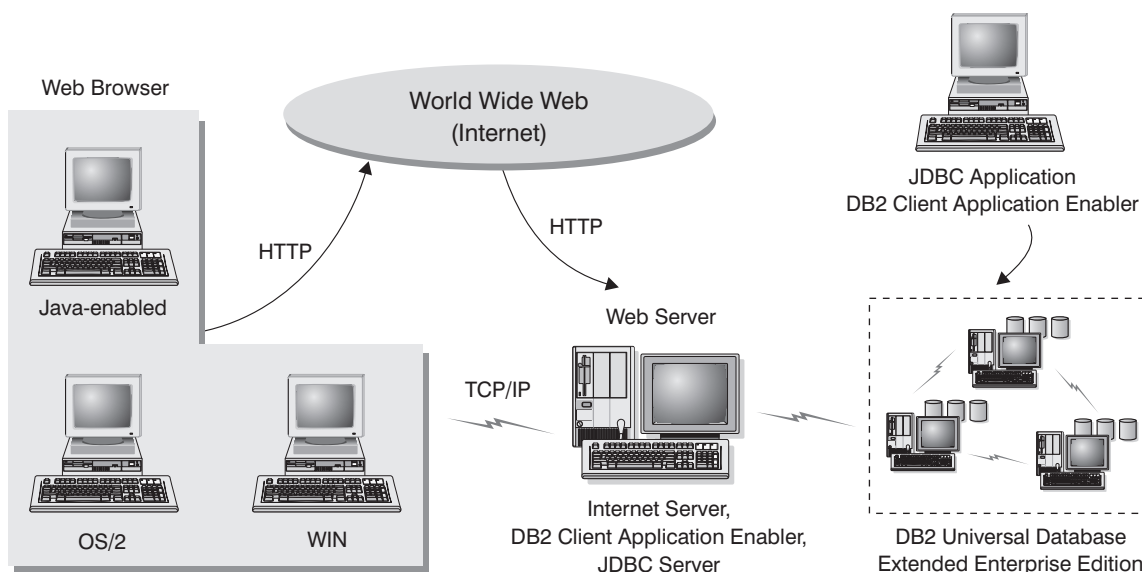


Figure 5. Accessing Internet Data Stored on DB2 Using JDBC

JDBC applications can be run from any system that has a DB2 Client Application Enabler installed; a Web browser and a Web server are not required.

For more information on Java enablement, refer to the Java Enablement web page (<http://www.software.ibm.com/data/db2/java/>). For more information on the JDBC API, refer to the URL <http://splash.javasoft.com/>.

Use *Net.Data* to create applications that are stored on a Web server and viewable from any Web browser. While viewing these documents, users can either select automated queries or define new ones that retrieve the specified information directly from a DB2 database.

Automated queries do not require user input; they are links in the HTML document and, when selected, they trigger existing SQL queries and return the results from a DB2 database. These links can be triggered repeatedly to access current DB2 data. Customized queries require user input. Users define the search characteristics on the Web page by selecting options from a list or by entering values in the entry fields. They submit the search by clicking on a push button. *Net.Data* uses the information that is supplied by the user to dynamically build a complete SQL statement, and it sends the query to the database.

A demonstration of *Net.Data* applications is available from the IBM Software Page (<http://www.software.ibm.com/>) under the Data Management link. Access the Data Management Link from the option on the Products drop-down window.

Net.Data can be installed with a DB2 server to allow local access to databases. *Net.Data* can be installed with a DB2 Client Application Enabler to allow remote access to databases. In both cases, *Net.Data* and a Web server must be installed on the same system. See Figure 6 on page 17 for an example of *Net.Data* that is being used to access data from remote DB2 databases.

Accessing DB2 Data Using DB2 Net.Data

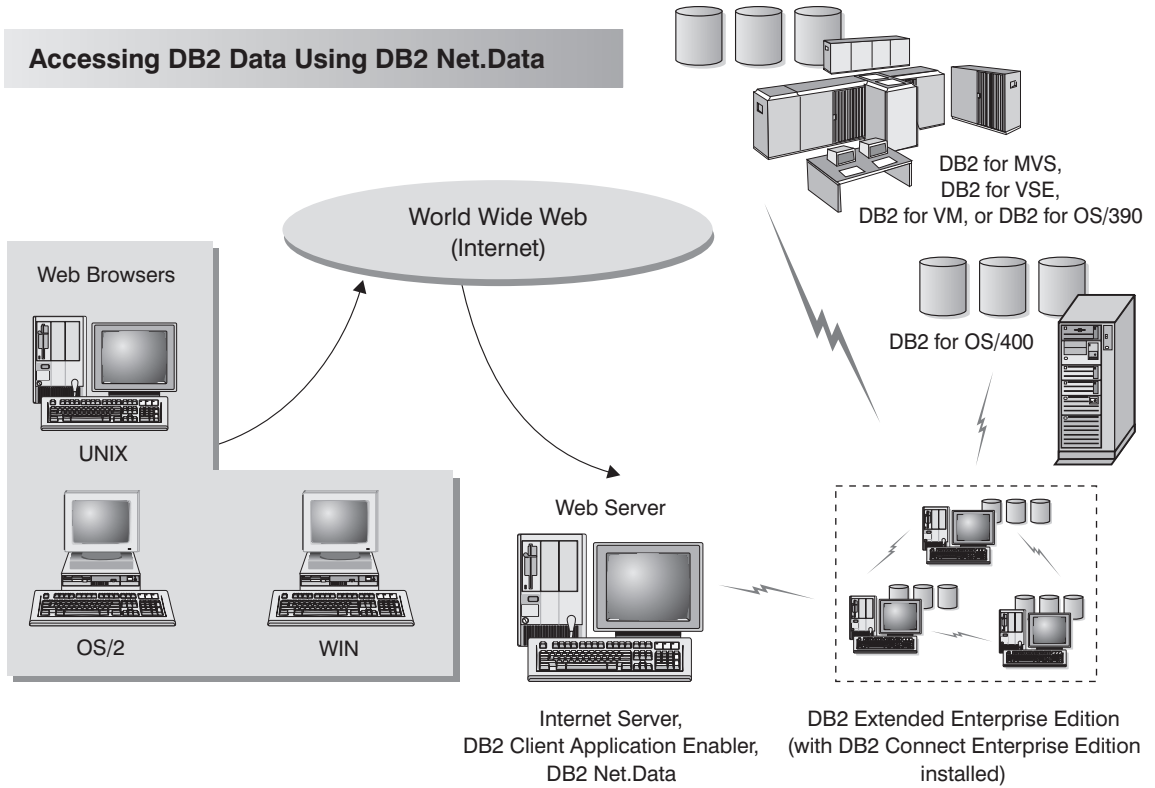


Figure 6. Accessing Internet Data Stored on DB2 Using Net.Data

Chapter 2. Introduction to DB2 Extended Enterprise Edition

A *database* is simply a collection of data. A *database manager* is the software that allows users to store and access data in a database. It achieves this function by using system resources, including CPU, memory, disk, and communications. In a partitioned database system, a single database manager and the collection of data and system resources that it manages are referred to collectively as a *database partition server (node)*. A *partitioned database system* is the collection of all the database partition servers that you create to handle data requests.

In DB2 Extended Enterprise Edition, the quality, functionality, reliability, and robustness of the database manager is extended to the entire partitioned database system.

In a partitioned database system, multiple database partition servers can be assigned to a machine (or to multiple machines), and the database manager at each machine is responsible for a portion of a database's total data (each database partition server houses a portion of the entire database). This portion of the database is known as a *database partition*. The fact that databases are partitioned across database partition servers is transparent to users and applications.

A partitioned database system can maintain very large databases and open opportunities for new applications. DB2 Extended Enterprise Edition provides fast response time for both decision-support (DSS) and online transaction processing (OLTP) applications.

DB2 Extended Enterprise Edition can be configured to execute on a *shared-nothing* hardware architecture, in which machines do not have to compete for resources. Each machine has exclusive access to its own disks and memory, and the database partition servers that run on the machines communicate with each other through the use of messages. For a database system to exploit shared-nothing architecture, typically one database partition server is assigned to each machine. Another possible configuration is running multiple logical nodes, in which more than one database partition server runs on a machine. For more information, see "Multiple Logical Nodes" on page 25.

Processing in a Partitioned Database Environment

In DB2 Extended Enterprise Edition, a database is distributed across multiple machines, and database partition servers are installed on a set of machines. Because the database is partitioned across multiple machines, you can use multiple CPUs across multiple machines to satisfy requests for information. The retrieval and update requests are decomposed automatically into sub-requests and executed in parallel on the database partition servers on each machine.

As an illustration of the power of processing in a partitioned database system, assume that you have 100 000 000 records that you want to scan in a single-partition database. This scan would require that a single database manager search 100 000 000 records. Now suppose that these records are spread evenly over 20 database partition servers; each database manager only has to scan 5 000 000 records. If each database partition server scans at the same time and with the same speed, the time required to do the scan should be approximately 5% of that of a single-partition system handling this task.

User interaction with DB2 Extended Enterprise Edition is handled through one database partition server (node). This database partition server is known as the *coordinator node* for that user. The coordinator node runs on the same machine as the application, regardless of whether the user or application is remote or local. Any database partition server can be used as a coordinator node. You should consider spreading out users across database partitions servers to distribute the coordinator function.

Cost-Based Query Optimization

DB2 Extended Enterprise Edition uses a *cost-based query optimizer*, which compares different methods for doing a unit of work, and selects the most efficient one. The optimizer provides the following features:

Transparent parallelism

Both new and existing applications that use data-manipulating SQL statements do not have to be changed when they are migrated to DB2 Extended Enterprise Edition. You only have to rebind them so the optimizer can generate the best plans for existing SQL queries.

Comprehensive use of data partitioning information

The optimizer uses information about how base tables, and the intermediate tables that result from queries, are partitioned across database partitions. This information is used to determine the best execution strategy.

Full-fledged, cost-based optimization

The optimizer considers different execution plans and chooses the one with the lowest cost. While comparing different strategies, it accounts for the inherent parallelism of different operations, and the costs introduced by messages.

Inter-partition and intra-partition parallelism of all relational operations.

All operations, including index and table scans, aggregation, set operations, joins, inserts, deletes, and updates can employ *inter-partition parallelism*. (Inter-partition parallelism means that the operator is executed in parallel by each database partition server; *intra-partition parallelism* means that different operators in the same query can be executed in parallel by the same database partition server.) When generating plans, the optimizer considers different parallel joining methods, including *collocated*, *directed*, and *broadcast* joins.

As a simple example, assume that you want to fetch specific columns from rows that meet some condition. The coordinator node sends a request to the other database partition servers to select the columns from rows that meet that condition. The other database partition servers then send this data to the coordinator node, which does the final processing.

If a row is being added to a table, the database partition server checks the *partitioning map*, which specifies the database partition server where the row is stored. The row is only sent to that database partition server, with the result that only the interested database partition servers take part in the insert. This keeps communications overhead as low as possible.

Configuration

Figure 7 on page 22 shows an example of a DB2 Extended Enterprise Edition hardware configuration.

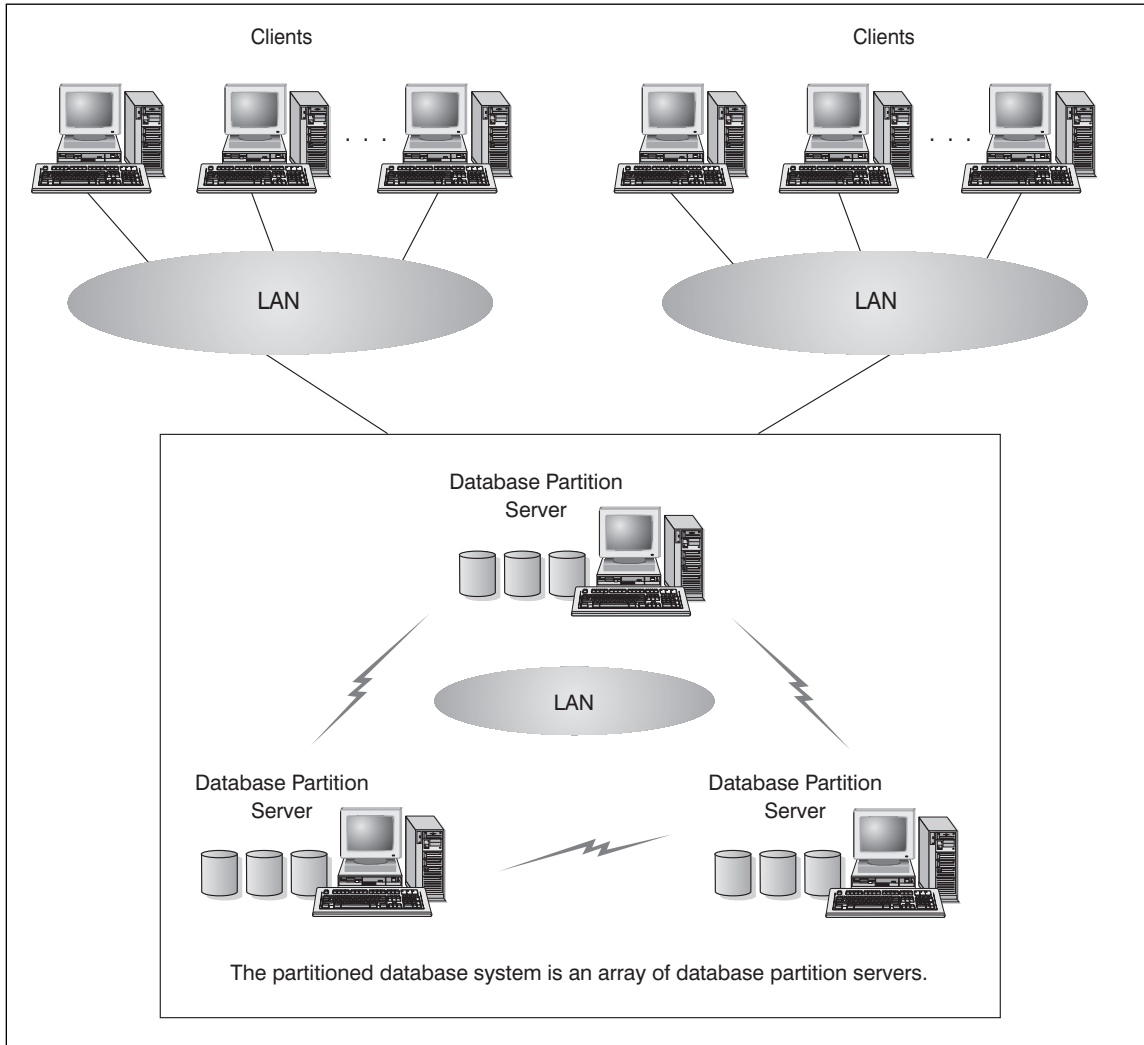


Figure 7. DB2 Extended Enterprise Edition Hardware Configuration

DB2 Extended Enterprise Edition can run on a cluster of individual machines interconnected by a LAN.

The following sections provide information that you should be familiar with before you configure your partitioned database system. Specifically, they describe:

- Physical storage
- How to combine database partitions into nodegroups, and how data is stored in them

- Logical nodes and their uses
- Instances
- Communications between database partition servers
- Agents

Machines and Storage

DB2 Extended Enterprise Edition implements a shared-nothing architecture, therefore each database partition server is the equivalent of a single-partition database system. Thus, the database storage capacity for the partitioned database system is equal to that provided by a single-partition database system multiplied by the number of nodes. You can store tables of up to 64 GB (gigabytes) per database partition, where giga is 10^9 . For example, in a 128-database partition (node) system, the maximum size of one table is 8 TB (terabytes, where tera is 10^{12}).

Nodegroups and Data Partitioning

You can define named subsets of one or more database partitions in a database. Each subset you define is known as a *nodegroup*. Each subset that contains more than one database partition is known as a *multipartition nodegroup*. Multipartition nodegroups can only be defined within database partitions that belong to the same database.

Three default nodegroups are created when you create a database: IBMDEFAULTGROUP, IBMCATGROUP and IBMTEMPGROUP.

If you want, you can create tables in the default nodegroups, IBMDEFAULTGROUP and IBMCATGROUP.

The IBMDEFAULTGROUP nodegroup contains all the database partitions for the database. When you create a database, a database partition is created at each database partition server (node) in the instance.

The IBMCATGROUP nodegroup for the database is created at the database partition server where you enter the **create database** command. This nodegroup only contains the database partition that is local to the database partition server where the command was entered. This database partition server is referred to as the *catalog node* of the database because the IBMCATGROUP nodegroup contains the catalog tables for the database.

You cannot directly work with the third default nodegroup, IBMTEMPGROUP. Like the IBMDEFAULTGROUP nodegroup, it also contains all the database partitions of a database. This nodegroup is used to contain all temporary table spaces.

Figure 8 shows an example of a database in which there are three nodegroups. Nodegroup 1 is a multipartition nodegroup made of four database partitions, and nodegroups 2 and 3 are both single-partition nodegroups.

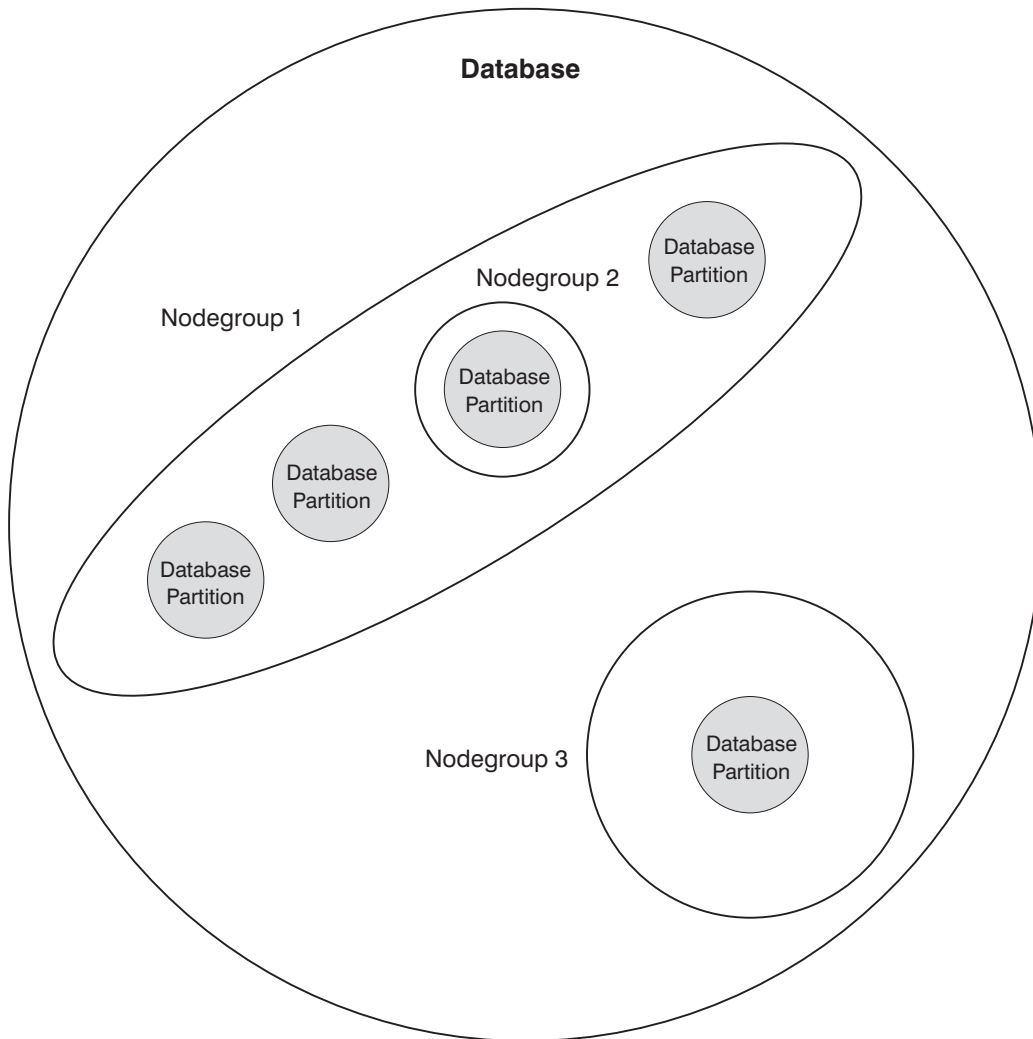


Figure 8. Nodegroups in a Database

When you want to create table spaces for a database, you first create the nodegroup where the table spaces will be stored, then create a table space in the nodegroup. After this, you create the tables in the table space.

You can drop database partitions from a nodegroup, or if you have increased the size of the database system you can add new database partitions to existing nodegroups. For information about adding and dropping nodes in nodegroups, refer to the *Administration Guide*.

As your database increases in size, you can add database partition servers to the database system for improved performance. This is known as scaling the database system. When you add a database partition server, a database partition is created for each database that already exists in the database system. You then add the new database partition to an existing nodegroup that belongs to that database. Finally, you redistribute data in that nodegroup to utilize the new database partition. For information about scaling a populated database system, refer to the *Administration Guide*. For information about scaling a partitioned database system without databases, see “Adding a Database Partition Server to an Instance” on page 88.

Each table defined in a multipartition nodegroup has a *partitioning key* associated with it. The partitioning key is an ordered set of columns whose values are used in conjunction with a *partitioning map* to determine the database partition on which a row of a given table resides. The partitioning map is an array of 4096 database partition numbers.

Columns of any data type (except LONG VARCHAR, LONG VARGRAPHIC, or LOB) can be used as the partitioning key. A table defined in a single-partition nodegroup may or may not have a partitioning key. Tables with only long-field columns can be defined only in single-partition nodegroups, and they cannot have a partitioning key. For more information about creating tables, refer to the *SQL Reference*.

The use of nodegroups and partitioning keys means that:

- Data can be distributed across multiple database partitions to reduce I/O and processing bottlenecks.
- Data can be redistributed when large volumes of system activity or increases in table size require the addition of more machines.

Refer to the *SQL Reference* for more information about creating nodegroups. Refer to the *Administration Guide* for more information about using nodegroups.

Multiple Logical Nodes

Typically, you configure DB2 Extended Enterprise Edition to have one database partition server assigned to each machine. There are situations, however, in which it would be advantageous to have several database partition servers running on the same machine. This means that the configuration can contain more nodes than machines. In these cases, the machine is said to be running *multiple logical nodes* if they participate

in the *same* instance. If they participate in different instances, this machine is *not* running multiple logical nodes.

With multiple logical node support, you can choose from three types of configurations:

- A standard configuration, where each machine has only one database partition server.
- A multiple logical node configuration, where a machine has more than one database partition server.
- A configuration, where several logical nodes run on each of several machines.

Configurations that use multiple logical nodes are useful when the system runs queries on a machine that has symmetric multiprocessor (SMP) architecture. The ability to configure multiple logical nodes on a machine is also useful if a machine fails. If a machine fails (causing the database partition server or servers on it to fail), you can restart the database partition server (or servers) on another machine, using the **db2start nodenum** command. This ensures that user data remains available.

Another benefit is that multiple logical nodes can exploit SMP hardware configurations. In addition, because database partitions are smaller, you can obtain better performance when performing such tasks as backing up and restoring database partitions and table spaces, and creating indexes.

Instances

In DB2 Extended Enterprise Edition, an instance contains all the database partition servers (nodes) that were defined to take part in a given partitioned database system. An instance has its own databases and instance directory. The instance directory contains the database manager configuration file, system database directories, node directories, and the node configuration file. See “Instance Directory” on page 83 for more information.

Each instance has different security from other instances on the same machine. This is shown in Figure 9 on page 27, which shows two separate instances. Instance 1 contains six database partition servers and Instance 2 contains eight database partition servers. (Multiple database partition servers are indicated when more than one line is shown between a database partition server and the instance directory.) The two instances appear to overlap, but this is due to the assignment of two database partition servers to each of the three machines in the middle of the figure. Database partition servers only belong to one instance. For example, Instance 1 does not contain database partition servers that belong to Instance 2.

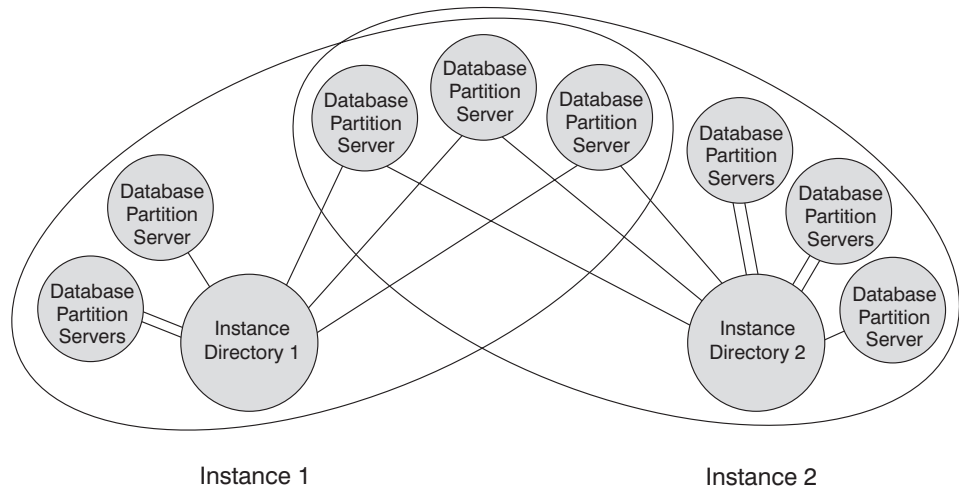


Figure 9. Two Instances

You can have multiple instances on the same machine, with each configured differently:

- To have distinct test and production environments
- To restrict access to specific databases
- To exploit different database configurations

Each instance is owned by the machine where the instance was first created (known as the instance-owning machine, node 0). The instance-owning machine stores information that is common to all of the database partition servers. For information about creating instances, see Chapter 11, “Working with Instances” on page 83.

Fast Communication Manager

The fast communication manager (FCM) provides communication support for DB2 Extended Enterprise Edition.

Each database partition server has one FCM thread to provide communications between database partition servers to handle agent requests, and to deliver message buffers. The FCM thread is started when you enter the **db2start** command.

You can specify the number of FCM message buffers with the *fcnum_buffers* database manager configuration parameter. For a description of this and other FCM parameters, refer to the *Administration Guide*.

If communications fail between database partition servers or if they re-establish communications, the FCM thread updates information (that you can query with the

database system monitor) and causes the appropriate action (such as the rollback of an affected transaction) to be performed. You can use the database system monitor to help you set the FCM configuration parameters. For information about FCM-related database system monitor output, refer to the *System Monitor Guide and Reference*.

Agents

All connection requests from client applications, whether they are local or remote, are allocated a corresponding coordinator agent. In symmetric multiprocessor (SMP) systems, multipartition database systems, and single-partition database systems in which the *intra_parallel* database manager configuration parameter is enabled, the coordinator agent distributes the database requests to subagents, and these subagents perform the requests for the application. Once the coordinator agent is created, it handles all database requests on behalf of its application, coordinating the subagents that perform requests on the database.

A coordinator agent may be:

- Connected to the database with an alias
- Attached to an instance.

Idle and associated agents reside in an agent pool. These agents are available for requests from coordinator agents operating on behalf of client programs, or from subagents operating on behalf of existing coordinator agents. The number of available agents is dependent on the *maxagents* and *num_poolagents* database manager configuration parameters.

For additional information about agents and the configuration parameters used to control them, refer to the *Administration Guide*. For information about the DB2 process model, refer to the *Troubleshooting Guide*.

Part 2. Installing DB2 Server

Chapter 3. Planning for Installation



If you know that your system meets DB2's hardware and software requirements, and you want to start installing DB2 right away, go directly to Chapter 4, "Installing and Configuring DB2 Extended Enterprise Edition" on page 43.

DB2 provides many components that you might want to use in your environment. Use the product and planning information in this section to confirm that your system fulfills the system requirements, and to decide which components you want to install.

Memory Requirements

Before you install DB2 Extended Enterprise Edition, decide what system configuration you need.

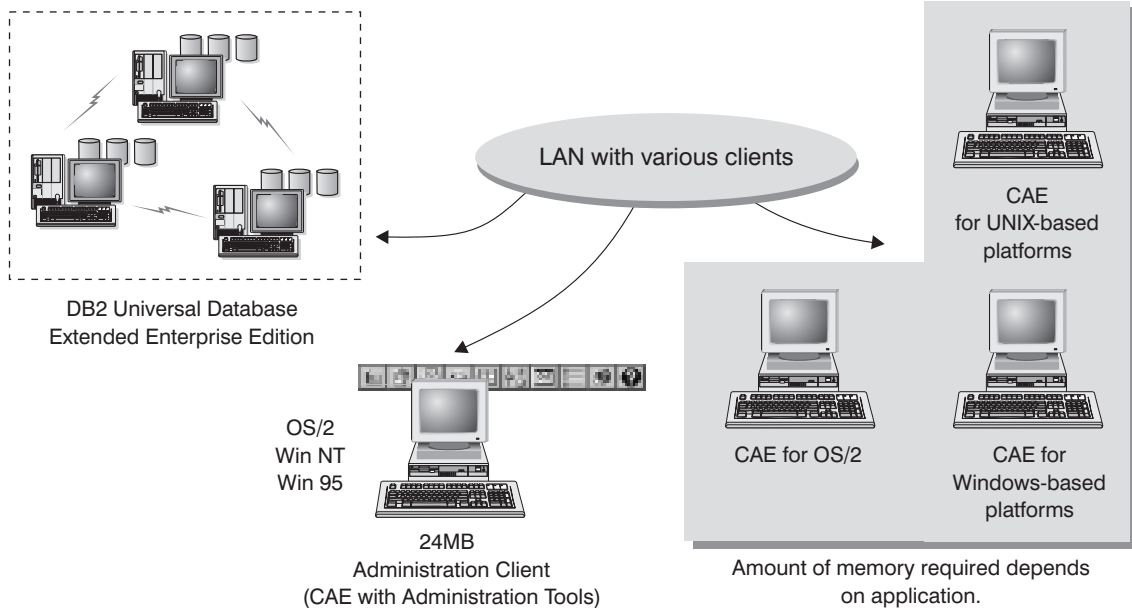
This section shows diagrams of the most common configurations that use DB2. Each diagram shows the products or components that are required for each system in the network (excluding operating system and communication requirements), and the approximate amount of memory required for such a configuration.

Configuration for Accessing DB2 Servers on the LAN

If you want to access data across a LAN from a remote database server, use the configuration shown in Figure 10 on page 32. This diagram shows DB2 Universal Database Extended Enterprise Edition installed on a machine and various clients connected to this server to access data. You must install the appropriate DB2 Client Application Enabler on each client.

Clients connect to a DB2 Universal Database Extended Enterprise Edition server using one of the supported protocols: APPC, IPX/SPX, Named Pipes, NetBIOS, or TCP/IP.

DB2 Extended Enterprise Edition - Memory Requirements



Note: CAE=Client Application Enabler

Figure 10. Configuration for Accessing DB2 Servers on the LAN

The amount of virtual memory (RAM) you need depends on the configuration that you want. For the server, it is recommended that a minimum of 32 MB of RAM be installed to accommodate five concurrent clients. We suggest that you have a minimum of 48 MB of RAM to accommodate 25 concurrent clients and 64 MB of RAM for 50 concurrent clients. If your system contains more database partition servers, more users, or you have specific performance requirements, you may require more memory. Use these guidelines to determine the amount of memory necessary in your environment.

An administration client is shown in the diagram. This client can be run on OS/2 or Windows 32-bit operating systems and can be used to manage DB2 servers on your network. Typically, 30 MB of memory is required to run all the graphical tools that you will need for an administration client.

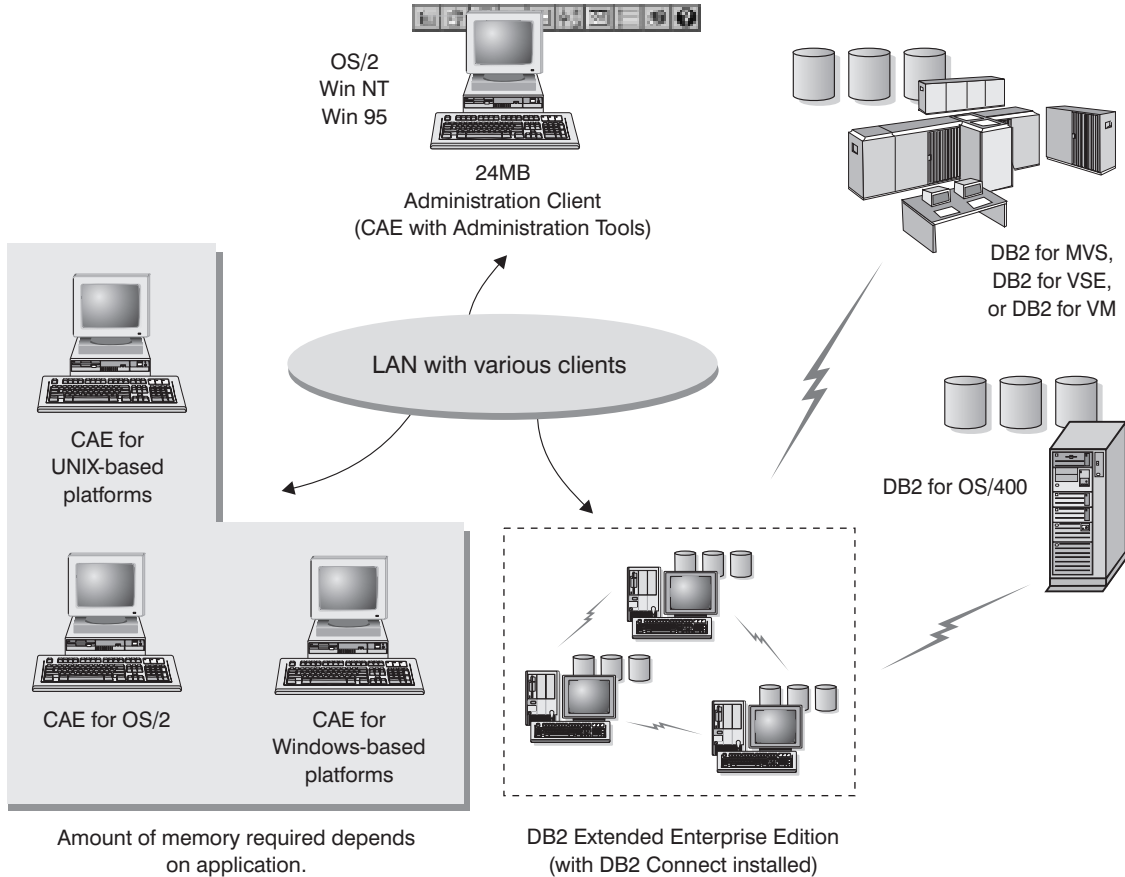
For the DB2 clients in your environment, the amount of memory required depends on the database applications you are running.

Configuration for Accessing LAN and Host Data

If you want to access databases on both your local LAN and on host systems such as OS/390, OS/400, MVS, VM, or VSE, use the configuration shown in Figure 11 on page 34. You will need to install the DB2 Connect Support option of DB2 Extended Enterprise Edition.

DB2 client applications connect to DB2 Extended Enterprise Edition servers using one of the supported protocols: APPC, IPX/SPX, Named Pipes, NetBIOS, or TCP/IP. DB2 then connects to the appropriate host database management system using DRDA over either APPC or TCP/IP.

DB2 Extended Enterprise Edition - Memory Requirements



Note: CAE=Client Application Enabler

Figure 11. Configuration for Accessing LAN and Host Data

The amount of virtual memory (RAM) you need depends on the configuration that you want. For the server, it is recommended that a minimum of 32 MB of RAM be installed to accommodate five concurrent clients. We suggest that you have a minimum of 48 MB of RAM to accommodate 25 concurrent clients and 64 MB of RAM for 50 concurrent clients. If your system contains more database partition servers, more users, or you have specific performance requirements, you may require more memory. Use these guidelines to determine the amount of memory necessary in your environment.

An administration client is shown in the diagram. This client can be on an OS/2 or Windows 32-bit operating systems and can be used to manage DB2 servers on your

network. Typically, 30 MB of memory is required to run all the graphical tools you will need for an administration client.

For the DB2 clients in your environment, the amount of memory required depends on the database applications you are running.

Configuration for Accessing from the Web

If you want Internet clients to use Web browsers to access data in databases on both LAN and host-based DB2 servers, use the configuration shown in Figure 5 on page 15 or Figure 6 on page 17.

Follow the guidelines in “Configuration for Accessing DB2 Servers on the LAN” on page 31 and “Configuration for Accessing LAN and Host Data” on page 33 to determine the memory required on your DB2 server and administration client.

Disk Requirements

This section shows the *minimum* amount of disk space that is required to install DB2 products and components. It does not include the disk requirements necessary for the operating system, application development tools, and communications products. Consult each product's documentation for these values. Estimates for disk space requirements are listed here; the actual amounts required depend on the functions you are using. Use this information to plan for systems with a large number of concurrent clients or databases. For information about space requirements for data, refer to the *Administration Guide*.

Estimating Fixed Disk Requirements

To estimate the disk requirements for a particular configuration, add the recommended minimum disk sizes for the products and components that you want to install. Include an allowance for your application data.

Server Components

Use Table 1 on page 36 to estimate the amount of disk space you need to install DB2 and associated components on your operating system.

Table 1. Estimating Disk Requirements

	Recommended Minimum Disk (MB)
DB2 Universal Database Extended Enterprise Edition	
DB2 Universal Database Extended Enterprise Edition — this amount includes the following components:	130 MB
<ul style="list-style-type: none"> • DB2 Connect Support • Graphical Tools (Client Configuration Assistant, Control Center, Performance Monitor, and Visual Explain) • East Asian Conversion Support • DB2 ODBV Driver 	<ul style="list-style-type: none"> 1 MB 45 MB 3 MB 3 MB
Online documentation in HTML format	35 MB

Note: The disk amounts listed are approximations.

Client Components

Use Table 2 to estimate the amount of disk space you need on each of your client workstations.

Table 2 (Page 1 of 2). Estimating Disk Requirements for Client Components

	Recommended Minimum Disk (MB)
DB2 Client Application Enabler for OS/2	
DB2 Client Application Enabler for OS/2 — this amount includes the following components:	70 MB
<ul style="list-style-type: none"> • Client Configuration Assistant • Graphical Tools • Control Center • Performance Monitor • Visual Explain 	<ul style="list-style-type: none"> 2 MB 10 MB 24 MB 8 MB 4 MB
<ul style="list-style-type: none"> • Online documentation in HTML format 	25 MB
DB2 Client Application Enabler for Windows 3.x	
DB2 Client Application Enabler for Windows 3.x	10 MB
<ul style="list-style-type: none"> • Online documentation in HTML format 	10 MB
DB2 Client Application Enablers for Windows 32-bit Operating Systems	
DB2 Client Application Enablers for Windows 32-bit operating systems — this amount includes the following components:	94 MB
<ul style="list-style-type: none"> • Client Configuration Assistant • Control Center • Performance Monitor • Visual Explain 	<ul style="list-style-type: none"> 2 MB 18 MB 6 MB 3 MB

Table 2 (Page 2 of 2). Estimating Disk Requirements for Client Components

	Recommended Minimum Disk (MB)
• Online documentation in HTML format	30 MB

Software Requirements

This section outlines the software required to run DB2 Universal Database and its associated products or components.

Server Product Requirements

Table 3 on page 38 lists the operating system and communications software required for DB2 Universal Database.

Table 3. Software Requirements for DB2 Servers

Product	Operating System	Communications
<ul style="list-style-type: none"> DB2 Universal Database Extended Enterprise Edition 	<ul style="list-style-type: none"> Windows NT Version 4.0 or later 	<p>APPC, IPX/SPX, NetBIOS, TCP/IP, or Named Pipes</p> <ul style="list-style-type: none"> For APPC connectivity, you require one of: <ul style="list-style-type: none"> IBM Communications Server for NT Version 5.01 or later IBM Personal Communications Server for NT Version 4.02 or later Microsoft SNA Server Version 2.11 or later The Windows NT base operating system provides NetBIOS, IPX/SPX, Named Pipes, and TCP/IP connectivity. <p>Notes:</p> <ol style="list-style-type: none"> The instance-owning machine should be running on Windows NT Server Version 4.0 or later. DB2 Connect for Windows NT supports APPC, NetBIOS, IPX/SPX, TCP/IP and Named Pipes for LAN connections. If SNA two-phase commit is necessary, then IBM Communications Server for Windows NT Version 5.01 is required. If you want to use DCE (Distributed Computing Environment) with Version 5.2 of DB2 Universal Database, you require a DCE product that is at OSF DCE level 1.1 with IBM DCE for Windows NT Version 2.0. In addition, DB2 also supports Gradient PC-DCE for Windows NT or 95 Version 2.0 Release Runtime Media Kit. For DB2 Connect support, you require DB2/MVS Version 5.1 plus its prerequisite, OS/390 DCE Base Services version 3 for DCE support. <p>Note: With DB2 Connect, you must install DCE Directory Services on the client and the DRDA server. You do not need DCE installed on a DB2 Connect Enterprise Edition server.</p> If you plan to use the ADSTAR Distributed Storage Manager (ADSM) facilities for backup and restore of your databases, you require the ADSM Client Version 2.1.x.7. If you have the IBM Antivirus program installed on your operating system, it must be Version 3.0 or later. If you want to use the Simple Network Management Protocol (SNMP) subagent, you require DPI 2.0 provided by IBM Netfinity Agent. If you plan to set up Failover support, then you require Windows NT Enterprise Edition V4.0 with the Microsoft Cluster Server support.

Client Product Requirements

Table 4 on page 39 lists the software requirements needed for the DB2 client products.

Table 4 (Page 1 of 2). Software Requirements for Clients

Component	Operating System	Communications
<ul style="list-style-type: none"> DB2 Client Application Enabler for OS/2 DB2 Software Developer's Kit for OS/2 Net.Data for OS/2 	<ul style="list-style-type: none"> OS/2 Warp Version 3 or Version 4 OS/2 Warp Connect Version 3 OS/2 Warp Server Version 4 OS/2 Warp Server Advanced V4 OS/2 Warp Server Advanced V4 with SMP Feature 	<p>APPC, IPX/SPX, Named Pipes (Local), NetBIOS, or TCP/IP</p> <ul style="list-style-type: none"> For APPC connectivity, you require IBM Communications Manager Version 1.x or later. For NetBIOS connectivity, you require IBM NTS/2 Version 1.0, IBM Communications Manager Version 1.1, or IBM OS/2 LAN Requester. For IPX/SPX connectivity, you require the Novell NetWare client for OS/2 Version 2.10 or later. For TCP/IP connectivity, you require IBM TCP/IP Version 2.0 or later. The OS/2 base operating system provides Named Pipes (Local) connectivity. Named Pipes is supported in DOS and WIN-OS/2 sessions. <p>Notes:</p> <ol style="list-style-type: none"> OS/2 Warp Server provides the Netfinity LAN client that Netfinity LAN needs for installing clients and servers. Net.Data requires OS/2 Warp Version 3 or later and a Web server such as IBM's Internet Connection Server. For DCE Cell Directory Services Support (CDS) for DB2 Client Application Enabler for OS/2, you must install IBM Distributed Computing Environment Cell Directory Service client, Version 2.10, on each client workstation.
<ul style="list-style-type: none"> DB2 Client Application Enabler for Windows 3.x DB2 Software Developer's Kit for Windows 3.x 	<ul style="list-style-type: none"> Microsoft Windows 3.x 	<p>IPX/SPX, NetBIOS, or TCP/IP</p> <ul style="list-style-type: none"> For IPX/SPX connectivity, you require Novell NetWare client for Windows, which comes with Novell Netware Server Version 3.x or Version 4.x; or Novell NetWare Workstation for Windows Version 1.0 or later with the latest update kit installed. For NetBIOS connectivity, you require the LAN Adapter and Protocol Support in any of these products: <ul style="list-style-type: none"> IBM LAN Support Program Version 1.2.1 or later DOS LAN Services (LAN Server 4.0) Banyan Vines NetBIOS Version 5.52 NetBEUI (available with Microsoft Windows for Workgroups 3.11) For TCP/IP connectivity, you require a TCP/IP stack that is compliant with the Windows Sockets Version 1.1 specification.
<ul style="list-style-type: none"> DB2 Client Application Enabler for Windows 3.x using WIN-OS/2 DB2 Software Developer's Kit for Windows 3.x using WIN-OS/2 	<ul style="list-style-type: none"> OS/2 Warp Version 3 or Version 4 OS/2 Warp Connect Version 3 OS/2 Warp Server Version 4 OS/2 Warp Server Advanced V4 OS/2 Warp Server Advanced V4 with SMP Feature 	<p>NetBIOS, IPX/SPX, TCP/IP, or Named Pipes (Local)</p> <ul style="list-style-type: none"> For NetBIOS connectivity, you require IBM NTS/2 Version 1.0, IBM Communications Manager Version 1.1, or IBM OS/2 LAN Requester. For IPX/SPX connectivity, you require Novell NetWare client for OS/2 Version 2.10 or later with virtual session support. For TCP/IP connectivity, you require IBM TCP/IP for OS/2 with the Windows Access kit.

Table 4 (Page 2 of 2). Software Requirements for Clients

Component	Operating System	Communications
<ul style="list-style-type: none"> DB2 Client Application Enabler for Windows 95 DB2 Software Developer's Kit for Windows 95 	<ul style="list-style-type: none"> Windows 95 4.00.950 or later 	IPX/SPX, Named Pipes, NetBIOS, or TCP/IP <ul style="list-style-type: none"> The Windows 95 base operating system provides NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. <p>Note: IPX/SPX connectivity is supported between Windows 95 clients and Windows NT servers. There is no support for IPX/SPX connectivity between Windows 95 clients and OS/2 and UNIX servers.</p>
<ul style="list-style-type: none"> DB2 Client Application Enabler for Windows 98 DB2 Software Developer's Kit for Windows 98 	<ul style="list-style-type: none"> Windows 98 	IPX/SPX, Named Pipes, NetBIOS, or TCP/IP <ul style="list-style-type: none"> The Windows 98 base operating system provides NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. <p>Note: IPX/SPX connectivity is supported between Windows 98 clients and Windows NT servers. There is no support for IPX/SPX connectivity between Windows 98 clients and OS/2 and UNIX servers.</p>
<ul style="list-style-type: none"> DB2 Client Application Enabler for Windows NT DB2 Software Developer's Kit for Windows NT Net.Data for Windows NT 	<ul style="list-style-type: none"> Windows NT Version 3.51 Windows NT Version 4.0 or later 	APPC, IPX/SPX, Named Pipes, NetBIOS, or TCP/IP <ul style="list-style-type: none"> The Windows NT base operating system provides NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. For APPC connectivity, you require one of the following products: <ul style="list-style-type: none"> IBM Communications Server for Windows NT Version 5.01. Microsoft SNA Server Version 2.11 or later on the LAN. You do not have to install the SNA Server on the same workstation as the DB2 Client Application Enabler for Windows NT. PCOMM AS/400 and 3270 V4.1 or later Wall Data Rumba <p>Note: If you have the IBM Antivirus program installed on your operating system, it must be Version 3.0 or later.</p>

Possible Client-to-Server Connectivity Scenarios

The following table shows the communication protocols that can be used when connecting a specific DB2 Client to a specific DB2 Server.

Table 5 (Page 1 of 2). Possible Client-to-Server Connectivity Scenarios	
Client	DB2 for Windows NT Server
OS/2	APPC IPX/SPX(1) NetBIOS TCP/IP

<i>Table 5 (Page 2 of 2). Possible Client-to-Server Connectivity Scenarios</i>	
Client	DB2 for Windows NT Server
Windows 3.x	IPX/SPX(1) NetBIOS TCP/IP
Windows 95 or Windows 98	IPX/SPX(1) Named Pipes NetBIOS TCP/IP
Windows NT	APPC IPX/SPX(1) Named Pipes NetBIOS TCP/IP
(1)	Direct Addressing

Understanding System Administrative Authority

System Administrative (SYSADM) authority is required to perform administration tasks such as cataloging, starting the database manager, or creating the sample database. By default, any user belonging to the Administrators group where the user account is defined has SYSADM authority. Throughout the book, IDs that have this authority are referred to as having SYSADM authority. To change the group that has SYSADM authority on the default instance, see “Working with the System Administrative Group” on page 80.

Chapter 4. Installing and Configuring DB2 Extended Enterprise Edition

This section describes how to install DB2 Universal Database Extended Enterprise Edition for Windows NT.



If you want to install DB2 clients, see Chapter 5, "Installing DB2 Clients" on page 55.

When you are creating a partitioned database system, the installation of DB2 Universal Database Extended Enterprise Edition differs between the first machine and subsequent machines. When you install DB2 Universal Database Extended Enterprise Edition on the first machine, you create the database partition server that becomes the instance-owning machine. When you perform each subsequent install, you add a new node to an existing partitioned database system.

Before You Begin

Read and perform the steps in this section before you begin the installation to ensure that you have the items and information that you will need.

Pre-Installation Requirements

Before you install DB2 Extended Enterprise Edition, for each of the machines that will make up your partitioned database system, you need to:

- Determine how each of the machines will be used.
- Ensure that all machines are connected to a LAN with TCP/IP configured, so that the machines can communicate with each other using TCP/IP.

To resolve a machine's hostname, enter the **hostname** command. To ensure that communications exist between any two machines, enter the **ping hostname** command from any machine within the network, where *hostname* represents the name of the machine you are trying to communicate with.

- Ensure that all the machines that will participate in this partitioned database system belong to the same Windows NT domain.
- Ensure that all machines are set to a consistent date and time.



To be considered consistent, the difference in GMT time between the machines must not exceed the value configured for the *max_time_diff* parameter in the database manager configuration file. The default time for the *max_time_diff* configuration parameter is 1 hour. For information on setting database manager configuration parameters, refer to the *Administration Guide*.

- Determine the user accounts required to install DB2 and its components. See “Determining User Accounts for DB2 Installation and Configuration.”
- Ensure that a TCP/IP port range is available on each machine, to be used by the fast communication manager (FCM). FCM is a feature of DB2 that handles communications between database partition servers.

To determine the port range, check the services file on each of the machines that will participate in your partitioned database system. The services file is located in the *x:\winnt\system32\drivers\etc* directory, where *x*: is the drive on which you installed Windows NT. Determine a port range that is available on all of the machines. You will need to specify this port range during installation.

Use the following criteria to determine the port range:

- If any of the machines that will participate in your partitioned database system will be running multiple logical nodes (MLNs), then the port range you set must be equal to or greater than the number of logical nodes on the machine running the most MLNs.
- If you are not planning to run multiple logical nodes, a single port is sufficient.
- Ensure that a TCP/IP port is available on each machine, to be used by the DB2 Performance Monitor. Use the instructions above to determine a port that is available on all machines. The same port and service name must be reserved on all machines that will make up the partitioned database system. The service name used for the DB2 Performance Monitor is **db2ccmsv**.

Determining User Accounts for DB2 Installation and Configuration

This section describes the user accounts required to install DB2 and its components.

Determining a Domain User Account for Installing DB2

To install DB2 Extended Enterprise Edition for Windows NT, you must be logged on as a domain user. You need this user account to install the instance-owning database partition server and each node that will make up the partitioned database system. If the domain user account is not part of the Domain Administrator’s group, it must have local administrator authority for each machine in the partitioned database system. If the

Domain user account is part of the Domain Administrator's group, it will automatically have local administrator authority of any of the machines within the domain.



Use the same domain user account to install each database partition server that will participate in the partitioned database system.

Determining a Domain User Account Using DB2

You need to have a domain user account that will be used to create a database partition server. DB2 will use this domain user account to log on when DB2 is started as a Windows NT service. This user account must meet the following requirements:

- It must be an existing domain user account.
- It must be a valid DB2 username. A valid DB2 username is eight characters or less, and complies with DB2's naming rules. For more information on DB2's naming rules, see Appendix D, "Naming Rules" on page 233.



You must grant SYSADM authority to the domain user account specified for the database partition server after the installation completes. For more information, see "Working with the System Administrative Group" on page 80.

Determining a User Account for the Administration Server



If you want to create an Administration Server on a backup domain controller, you must specify a user account that already exists on the Primary Domain Controller.

When you create the database partition server that will become the instance-owning machine, the setup program automatically creates the Administration Server. During installation, you will be asked to provide a user account that will be used by the Administration Server to log on to the system and to start itself as a service.

The user account must belong to the local Administrators group. By default, the setup program will default both the username and the password to *db2admin*.

The setup program will check to see if the user account specified for the Administration Server exists. If it does not exist, it will be created. If it does exist, the setup program will verify that this user account is a member of the Administrators group and that the password supplied is valid.

Notes:

1. If the *db2admin* user account already exists on your system, you must use the same password specified for this account.

If the *db2admin* user account uses the password *db2admin*, for security reasons you should change this password (since this password is the default for any DB2 install and therefore is well known).

2. If the *db2admin* user account *does not* already exist, for security reasons, you should provide your own password (since this password is the default for any DB2 install and therefore is well known). Enter the new password in the **Password** and **Confirm Password** fields. You must ensure that it is eight characters or less, and complies with DB2's naming rules. For more information on DB2's naming rules, see Appendix D, "Naming Rules" on page 233.

Performing the Installation

To install DB2 Extended Enterprise Edition, you need to perform steps to set up your system environment. You also need to determine certain information about your system in order to configure your partitioned database system. Before you begin the installation, make sure that you have all of the information that you require. See "Before You Begin" on page 43 for a list of pre-installation requirements.

Installing DB2 Extended Enterprise Edition

To install DB2 Extended Enterprise Edition on a Windows NT workstation:

- 1 Log on as a user that meets the requirements for installing DB2. For more information, see "Determining User Accounts for DB2 Installation and Configuration" on page 44.
- 2 Shut down any other programs so that the setup program can update files as required.
- 3 Insert the CD-ROM into the drive. The auto-run feature automatically starts the setup program. The setup program will determine the system language, and launch the setup program for that language. If you want to run the setup program in a different language, see the tip that follows.



To manually invoke the setup program, do the following:

1 Click on **Start** and select the **Run** option.

2 Type the following in the **Open** field:

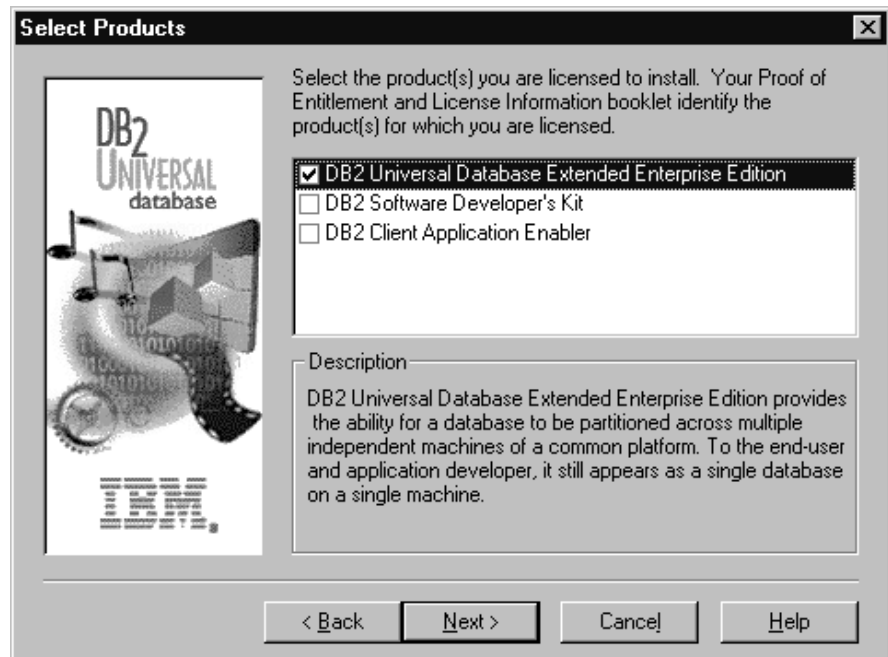
```
x:\setup /i language
```

where:

- *x*: represents your CD-ROM drive
- *language* represents the two-character country code for your language (for example, EN for English). Table 14 on page 232 lists the code for each available language.

3 Click on **OK**.

4 The Welcome window opens. Click on the **Next** push button to open the Select Products window, as shown below. Select the product that you want to install and click on the **Next** push button.



5 The Selection Installation Option window opens.

The setup program allows you to choose different installation options. All of these options install DB2 Extended Enterprise Edition.

Use the following information to determine the option you want to select.

- If there is no previous version of DB2 on the machine, you have two installation options:

This machine will be the instance-owning database partition server

The setup program creates a default instance, called DB2MPP, on what will become the instance-owning database partition server.

This machine will be a new node on an existing partitioned database system

The setup program adds this machine as a node to an existing partitioned database system.

- If there is a previous version of DB2 on the machine, you have three installation options:

This machine will be the instance-owning database partition server

The setup program will prompt you if you want to use the existing default instance as the default instance in the partitioned database system. Choose one of the following options:

Yes, use the instance

If you choose to use the existing default instance, the setup program migrates the existing instance to the DB2 Version 5 multipartition format (the instance retains its name). This machine becomes the instance-owning database partition server. All other existing DB2 instances will be migrated to the DB2 Version 5 single-partition format.

No, create a new default instance

If you choose to create a new default instance, the setup program creates a default instance, called DB2MPP, on what will become the instance-owning database partition server. All existing DB2 instances will be migrated to the DB2 Version 5 single-partition format.

This machine will be a new node on an existing partitioned database system

The setup program adds this machine as a node to an existing partitioned database system. All existing DB2 instances will be migrated to the DB2 Version 5 single-partition format.



If you are adding a node to an existing partitioned database system that contains databases, you must run the **db2 add node** command after the installation. This creates a database partition on the new node for any existing databases in your partitioned database system.

For more information, refer to the *Administration Guide*.

This machine will be a single-partition database server

This option does not allow you to exploit the scalability and power of a multipartition database system. All existing DB2 instances are migrated to the DB2 Version 5 single-partition format.

- 6 Respond to the setup program's prompts. Online help is available to guide you through the remaining steps. Invoke online help by clicking on the **Help** push button at any time.

You can click on the **Cancel** push button at any time to end the installation.

- 7 After you install DB2, you must restart your workstation before you can begin to use DB2. Select a restart option and click on the **Finish** push button.

Note: The instance directory is located in the `\sql11ib` directory. During the installation, the DB2 setup program gives the instance directory the shared name `DB2-INSTANCE_NAME`. Read and write permissions are automatically granted to everyone in the domain. After completing the installation, you can change the permissions to restrict access to the directory.



If you want to add a logical node to an existing instance, see “Adding a Database Partition Server to an Instance” on page 88 or refer to the *Administration Guide* for further information. If you want to add a new physical node to an existing instance, see “Performing the Installation” on page 46.

The setup program has:

- Created DB2 program groups and items (or shortcuts)
- Registered a security service
- Registered the DB2 Remote Command service
- Registered the DB2 Governor service
- Updated the Windows registry
- Created a DB2 Administration Server on the instance-owning machine, added it as a service, and configured it so that DB2 tools can administer the server. This service's start type was set to **Automatic**. If you installed a node, a DB2 Administration Server was not created. See “Creating an Administration Server” on page 95 for more information.



You have completed all the installation steps. Go to “Verifying the Installation” on page 50 to verify the installation.

By default, any user belonging to the Administrators group where the user account is defined has System Administrative (SYSADM) authority on the instance.

If you log on to the system with a domain account, in order to have SYSADM authority, the domain user account must belong to the Administrators group at the Primary Domain Controller; if you log on to the system with a local account, you must belong to the local Administrators group.

To change the group that has SYSADM authority on the default instance, see “Granting Users Authorization” on page 80.

Verifying the Installation



This section describes how you can create a sample database to verify that DB2 is properly installed. If you want, you can verify installation immediately after you create the instance on the instance-owning machine (node 0). However, it is recommended that you wait and create the sample database after you have added all of the machines that will make up your partitioned database system.

You can verify that DB2 is installed correctly by creating the SAMPLE database on your system, and accessing data from the database.

If you installed the DB2 tools, you can verify the installation as follows:

- 1 Log on to the system with a valid DB2 user account that has System Administrative (SYSADM) authority.
- 2 Invoke First Steps by clicking on **Start** and selecting **Programs->DB2 for Windows NT->First Steps**
- 3 Click on the **Create the SAMPLE Database** graphic button on the main panel of First Steps.

This command may take a few minutes to process. When the sample database has been created, you will receive a completion message; click on **OK**.
- 4 Once the database is created, click on the **View the SAMPLE database** graphic button to select data from the SAMPLE database. This starts the Command Center allowing you to use the supplied script that shows some of the data from the database. Click on the **Start or stop execution** icon (the gears icon in the top left corner of the screen) to begin the query.
- 5 Click on the **Work with the SAMPLE database** graphic button to start the Control Center. This allows you to see the tables that are in the SAMPLE database and enables you to perform actions on them.

If you did not install the DB2 tools, you can verify the installation as follows:

- 1 Log on to the instance-owning machine with a user account that has SYSADM authority.
- 2 Create the sample database using the **db2sampl** command at the operating system command prompt.

By default, the sample database will be created on the drive where DB2 was installed; however, you can specify a specific drive on which to create this database. It is automatically cataloged with the database alias SAMPLE when it is created.



You must ensure that the drive where the sample database will be created is available on each machine in your partitioned database system.

For example, to create the sample database on the F: drive, enter:

```
db2samp1 F:
```

This command may take a few minutes to process. There is no completion message; when the command prompt returns, the process is complete.

- 3 Start the database manager by entering the **db2start** command.
- 4 Once the database is created, enter the following commands in the command line processor to connect to the sample database, retrieve a list of all the employees that work in department 20, and reset the database connection.

```
connect to sample
"select * from staff where dept = 20"
connect reset
```



After you have verified the installation, you can remove the sample database to free up disk space. Enter the **drop database sample** command to drop the sample database.

Creating a Server Access Profile



You should have at least one database on your server before creating a server access profile.

To make it easy to set up a client, you can create a profile of your server that contains detailed information that the client needs to make a connection. This information includes the names and locations of all databases and instances on your server system. Creating a profile makes it easy to set up a single client as well as hundreds of clients. Follow these steps to create a profile that you can use to configure your clients:

- 1 Click on **Start** and select **Programs->DB2 for Windows NT->Administration Tools->Control Center**.
- 2 Click on the **[+]** sign beside the **Systems** icon to list all the known systems.
- 3 Select the system for which you want to create an access profile and click on the right mouse button.
- 4 Select the **Generate access profile** option from the pop-up menu.

| **5** Select the path and type in a name for this profile, then click on the **Generate**
| push button.

| **6** Click on **OK**.



Now that you have successfully installed DB2, go to Chapter 5, "Installing DB2 Clients" on page 55 to install remote clients to access DB2 databases.

If you are migrating from a previous version, go to Chapter 17, "Migrating from Previous Versions and Releases" on page 141 for information on how to complete the migration process.

Part 3. Installing Clients

Chapter 5. Installing DB2 Clients

Use the instructions in this section to install a DB2 client.

A DB2 Client Application Enabler is installed with any DB2 Version 5 product. If a DB2 Version 5 product is already installed, there is no need to install a DB2 Client Application Enabler.

Clients on a LAN that will connect to a DB2 Universal Database server must have the appropriate DB2 Client Application Enabler or a DB2 Software Developer's Kit installed.

DB2 Client Application Enablers (except the DOS client) are on the DB2 Client Application Enablers CD-ROM.

DB2 Client Application Enablers and DB2 Software Developer's Kits are available for the following: AIX, DOS, HP-UX, Macintosh, OS/2, SCO UnixWare 7, SCO OpenServer, Silicon Graphics IRIX, SINIX, Solaris, Windows 3.x, and Windows 32-bit operating systems.

You may install DB2 Client Application Enablers on any number of workstations; licensing restrictions are controlled at the server.



Go to the section that gives instructions for the DB2 client that you want to install.

- Chapter 6, "Installing DB2 Clients on Windows 32-Bit Operating Systems" on page 57.
- Chapter 7, "Installing DB2 Clients on OS/2 Operating Systems" on page 61.
- Chapter 8, "Installing DB2 Clients on Windows 3.x Operating Systems" on page 63.

To install and configure UNIX or Macintosh clients, refer to the online *Installing and Configuring DB2 Clients* manual, available on the DB2 Client Application Enablers CD-ROM.

To obtain DB2 Client Application Enablers for all clients, including DOS, connect to the IBM Web site (<http://www.software.ibm.com/data/db2/db2tech/clientpak.html>), search for the Client Application Enablers, and follow the instructions provided.

The DB2 Software Developer's Kits on the DB2 Extended Enterprise Edition CD-ROM have been upgraded so that it can communicate with a DB2 Extended Enterprise Edition server. If you installed a DB2 Software Developer's Kit from the DB2 Software Developer's Kit CD-ROM, you must apply a patch to use it with a DB2 Extended Enterprise Edition server. To obtain this patch, referred to as a maintenance package, connect to the IBM Web site (<http://www.software.ibm.com/data/db2/db2tech>).

Chapter 6. Installing DB2 Clients on Windows 32-Bit Operating Systems

This section contains the information you need to install the DB2 Client Application Enabler on Windows 32-bit operating systems.

To install a DB2 Client Application Enabler:

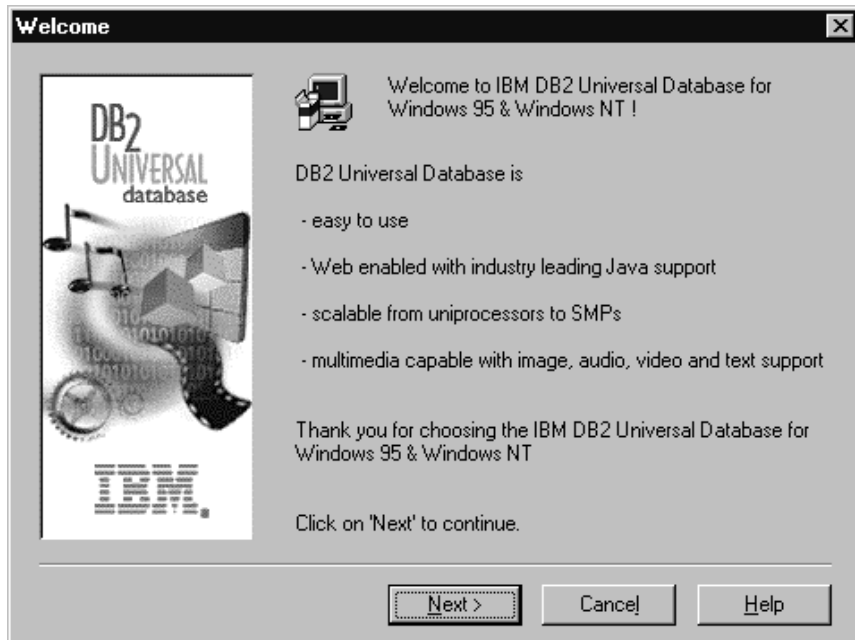
- 1 Log on as a local Administrator.
- 2 Shut down any other programs so that the setup program can update files as required.
- 3 Insert the DB2 Client Application Enablers CD-ROM that contains the Windows 95, Windows 98, or Windows NT client into the drive. The auto-run feature automatically starts the setup program. The setup program will determine the system language, and launch the setup program for that language. If you want to run the setup program in a different language, see the tip that follows.



To manually invoke the setup program, do the following:

- 1 Click on **Start** and select the **Run** option.
- 2 Type the following in the **Open** field:
`x:\setup /i language`
where:
 - *x*: represents your CD-ROM drive
 - *language* represents the two-character country code for your language (for example, EN for English). Table 14 on page 232 lists the code for each available language.
- 3 Click on **OK**.

- 4 The Welcome window opens.



Click on the **Next** push button to continue.

- 5 The Enable Remote Administration window opens. Select the **Install components required to administer remote servers** check box if you would like to administer remote servers from this client. Click on the **Next** push button.
- 6 Select the installation type you prefer:



Typical Install: Installs those DB2 components that are used most often, including all required components, ODBC support, documentation, and commonly used DB2 tools such as the Client Configuration Assistant and the Information Center. A DB2 instance is created.



Compact Install: Installs only the required DB2 components and ODBC support. A DB2 instance is created.



Custom Install: Installs only those components that you select. A DB2 instance is created.

- 7 Respond to the setup program's prompts. Online help is available to guide you through the remaining steps. Invoke online help by clicking on the **Help** push button at any time.

You can click on the **Cancel** push button at any time to end the installation.

- 8** After you install the product, you must reboot before you can begin to use it. Select a reboot option and click on the **Finish** push button. This completes the installation.

The installation program has:

- Created DB2 program groups and items (or shortcuts).
- Updated the Windows registry.
- Created a default client instance called DB2.
- Registered a security service.



To configure your client to access remote servers, go to Chapter 9, "Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 69.

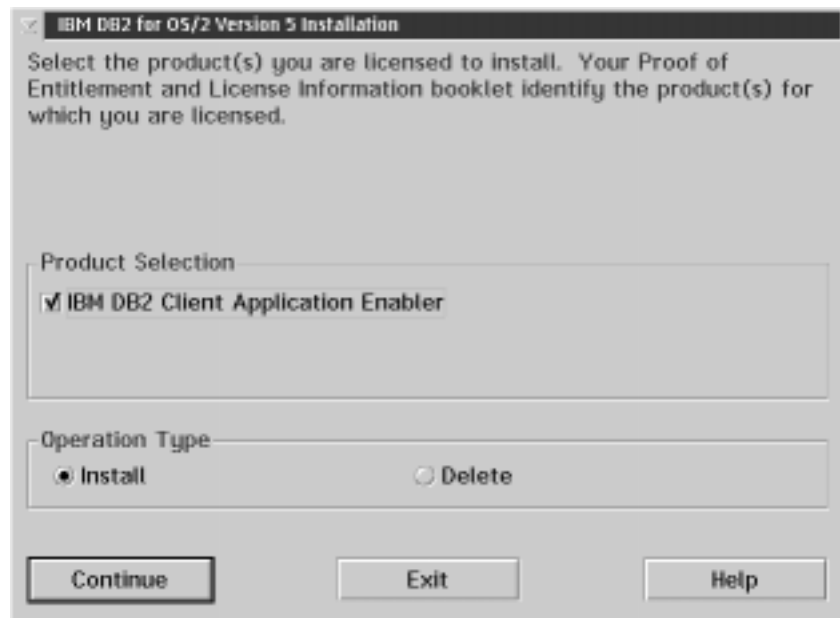
Chapter 7. Installing DB2 Clients on OS/2 Operating Systems

This section contains the information that you need to install the DB2 Client Application Enabler on OS/2 operating systems.

To install the DB2 Client Application Enabler for OS/2:

- 1 Insert the DB2 Client Application Enablers CD-ROM that contains the OS/2 client into the drive.
- 2 Begin the installation program as follows:
 - a Open an OS/2 window and set the drive to x:, where x represents your CD-ROM drive.
 - b Set the current directory to `\db2\os2\language\install`, where *language* is the two-character country code that represents your language (for example, EN for English). Table 14 on page 232 lists the codes for each available language.
 - c Enter the **install** command.

The IBM DB2 for OS/2 Version 5 Installation window opens.



Select the **IBM DB2 Client Application Enabler** check box, and click on the **Continue** push button.

Invoke the online help by clicking on the **HELP** push button at any time. The Install window opens.

- 3 Indicate whether or not you want to update the `config.sys` file as part of the installation.
 - If you want the `config.sys` file updated, click on **OK**. A backup copy of your existing file is kept as `config.0xx`. This is the recommended method.
 - If you do not want the `config.sys` file updated, clear the **Update CONFIG.SYS** check box and click on **OK**. You will need to manually update your `config.sys` file with the values that are stored in the `config.add` file.

The Install - directories window opens.

- 4 The Install - directories window provides a list of components. You can choose which of the components you want to install on your system.

For the Install - directories window, complete the entries as follows:

- a Select the components you want to install.
- b In the **File directory** field, type the directory where you want the product installed. The default directory is `c:\sql11ib`.

Note: If a DB2 Version 5 product is already installed on this workstation, you must install on the same drive and directory.
- c Click on the **Disk Space** push button to see how much space you have on each of the drives on your machine, and to change disks if necessary.
- d Click on the **Install** push button to begin the installation.

A progress window appears until the installation is complete. You can click on the **Stop** push button at any time to end the installation.

- 5 At the end of the installation, click on the **Exit** push button to exit the installation program. You must shut down and reboot before you can use the client.

Note: If you want to run Windows 3.x applications on your OS/2 client, you must also install the DB2 Client Application Enabler for Windows 3.x on your system. You should install this Client Application Enabler in the same directory as DB2 Client Application Enabler for OS/2. Having both DB2 Client Application Enablers in the same directory merges the DB2 directories and allows the CCA to configure the access to remote DB2 databases for both OS/2 and Windows 3.x applications.



To configure your client to access remote servers, go to Chapter 9, "Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 69.

Chapter 8. Installing DB2 Clients on Windows 3.x Operating Systems



If you have a previous version of the DB2 Client Application Enabler for Windows 3.x installed on your system, it is recommended that you back it up before beginning the installation.

To install the DB2 Client Application Enabler on Windows 3.x:

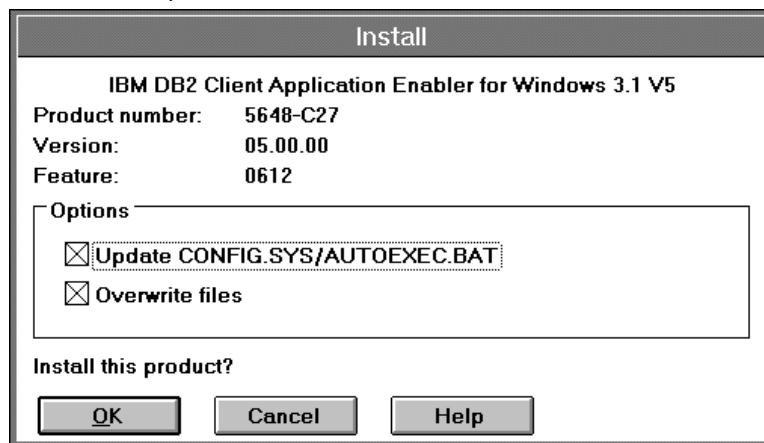
- 1 Insert the DB2 Client Application Enablers CD-ROM that contains the Windows 3.x client into the drive.

- 2 To begin the installation program enter:

```
x:\db2\windows\language\install\install.exe
```

where x: represents your CD-ROM drive and where *language* is the two-character country code that represents your language (for example, EN for English). Table 14 on page 232 lists the codes for each available language.

The Install window opens.



- 3 Indicate whether or not you want to update the autoexec.bat file as part of the installation.

Note: The config.sys file will not be modified by the installation.

- If you want the autoexec.bat file updated, check that the **Update CONFIG.SYS/AUTOEXEC.BAT** check box is selected, then click on **OK**. A backup copy, called autoexec.bak, is created in the same directory as your autoexec.bat file. This is the recommended method.

- If you do not want the autoexec.bat file updated, clear the **Update CONFIG.SYS/AUTOEXEC.BAT** check box. You will need to manually update your autoexec.bat file with the values that are stored in the autoexec.add file.



The **Overwrite files** check box is selected by default, it is recommend that you do not change this setting.

- 4 Click on **OK** to continue. The Install - directories window opens.

- 5 Select the components that you want to install.
- 6 In the **File directory** field, type the directory where you want the product installed. The default directory is c:\sqllib.



If you are installing the DB2 Client Application Enabler for Windows 3.x on an OS/2 system that already has the Version 5 DB2 Client Application Enabler for OS/2 installed, enter the same directory in the **File directory** field as the DB2 Client Application Enabler for OS/2. This will merge the two DB2 directories together, allowing tools like the Client Configuration Assistant to be used to configure clients to access a remote DB2 database for both Windows 3.x and OS/2 applications.

- 7 Click on the **Disk space** push button to see how much space you have on the selected drive.

8 Click on the **Install** push button to start the installation. A progress window appears until the installation is complete.

You can click on the **Stop** push button at any time to exit the installation program.

9 Click on the **Exit** push button to end the installation.

10 Once you have completed the installation, restart your WIN-OS/2 session before using the DB2 for Windows 3.x Client Application Enabler.

Part 4. Configuring DB2 Communications

Chapter 9. Configuring Client-to-Server Communications Using the Client Configuration Assistant



The information in this section describes how to use the Client Configuration Assistant to configure OS/2 and Windows 32-bit operating systems to access remote DB2 Universal Database servers.

Before you can use the Client Configuration Assistant, be sure that you have completed the instructions in "Before You Begin" on page 43.

To configure all other clients, refer to the online *Installing and Configuring DB2 Clients* manual.

Use the Client Configuration Assistant to configure your OS/2 and Windows 32-bit operating systems to access remote DB2 servers. The Client Configuration Assistant provides three configuration methods:

- One method makes use of a server's access profile
- One method searches the network for databases
- One method requires that you enter the database name and the communication protocol parameters for the DB2 server.

Perform the following steps to configure your workstation to access remote servers:

- 1** Start the Client Configuration Assistant.

For OS/2:

Double-click on the **Client Configuration Assistant** icon, located in the **DB2 for OS/2** folder.

For Windows 32-bit operating systems:

Click on **Start** and select **Programs->DB2 for Windows->Client Configuration Assistant**.

The Welcome panel opens each time you start the CCA, until you have added at least one database to your client.

- 2** Click on the **Add Database** or **Add** push button to configure connections using the Add Database SmartGuide. Choose one of the following configuration methods:

- If your administrator provided you a server profile, select the **Use an access profile** radio button and click on the **Next** push button.
 - a** Click on the **Browse** push button and select the appropriate access profile.

- b** You are presented with a list of systems, instances, and databases. Select the database that you want to add and proceed to Step 3.
- If you want to search the network for databases, select the **Search the network** radio button and click on the **Next** push button.

Note: Discovery cannot find information about pre-Version 5 DB2 systems.

- a** Click on the **[+]** sign beside the **Known Systems** icon to list all the systems known to your client.
- b** Click on the **[+]** sign beside a system to get a list of the instances and databases on it. Select the database that you want to add and proceed to Step 3.



The **Other Systems (Search the network)** icon will only appear if the client's *DISCOVER* parameter is set to *SEARCH* (this is the default setting).

- c** If the system that contains the database that you want to add is not listed, click on the **[+]** sign beside the **Other Systems (Search the network)** icon to search the network for additional systems. Click on the **[+]** sign beside a system to get a list of the instances and databases in it. Select the database that you want to add and proceed to Step 3.
 - d** If the system you want is still not listed, it can be added to the list of systems by clicking on the **Add System** push button. Enter the required communication protocol parameters for the remote Administration Server and click on **OK**. For more information, click on the **Help** push button.
- Select the database that you want to add and proceed to Step 3.



The Client Configuration Assistant may be unable to detect the remote system if:

- The Administration Server *DISCOVER* configuration parameter on the remote system is set to *DISCOVER=DISABLE*.
- The *DB2COMM* registry value and the *DISCOVER_COMM* configuration parameter are not configured on the Administration Server of the remote system with a discovery protocol being used on the client (the client discovery protocol is specified by the *DISCOVER_COMM* configuration parameter in the client's configuration).
- The Administration Server is not running on the remote system.
- By default, the Discovery function will search the network for 40 seconds; this may not be long enough to detect the remote system. You can configure the *DB2DISCOVERYTIME* registry value to specify a longer period of time.
- The network that the Discovery request is running on may have been configured so that the Discovery request does not reach the remote system desired.
- If you are using NetBIOS as the Discovery protocol, you may need to configure the *DB2NBDISCOVERRCVBUFS* registry value to a larger value to enable the client to receive more concurrent Discovery replies.

- If you have the protocol information for the database that you want to connect to, select the **Manually configure a connection to a DB2 database** radio button and click on the **Next** push button.
 - a** Select the radio button that corresponds to the protocol that you want to use from the **Protocol** list and click on the **Next** push button. If you selected the **TCP/IP** or **APPC** radio button, select the radio button that corresponds to the type of system where the database that you are trying to connect to resides.
 - b** Enter the required communication protocol parameters and click on the **Next** push button. For more information, click on the **Help** push button.
 - c** Enter the database alias name of the database that you want to connect to in the **Target database** field and click on the **Next** push button.



When a database is created on the remote server, if a database alias is not specified during database creation, the database is created with a database alias=*database_name*; otherwise, the database alias is the name specified.

- 3** To specify a local database alias name or to add a description, click on the **Next** push button. If you do not want to specify a database alias name, or add a description, click on the **Done** push button. If you do not specify a database alias name, the default will be the same as the remote database alias name.

- 4 If you want to run ODBC-enabled applications, click on the **Next** push button. If you do not plan to use ODBC, click on the **Done** push button to finish using the Add Database SmartGuide and proceed to Step 6.
- 5 If you are using ODBC applications:
 - a Select the **Register this database for ODBC** check box, if it is not pre-selected.
 - b Select the radio button that describes the type of data source that you want to register this database as. For more information, click on the **Help** push button.
 - c Click on the **Application** drop down box and select the application that you want to use.
 - d Click on the **Done** push button to finish using the Add Database SmartGuide and add the database that you selected.
- 6 The Confirmation window opens. Click on the **Test Connection** push button to test the connection to this database.
- 7 Enter your user ID and password to access the database and click on **OK**. If the connection is successful, a message confirming the connection appears. If the connection fails, click on the **Help** push button for more information. If you require additional information, refer to the *Troubleshooting Guide*.
- 8 You are now able to use this database. If you want to access another database, click on the **Add Another** push button. To finish using the Add Database SmartGuide, click on the **Close** push button.

Verifying the Connection

When the configuration of the server and client is complete, use the following steps to verify that you can access data from a remote database:



You will need to connect to a remote database to test the connection. If you do not have a database on the server, create the SAMPLE database on the server to test the connection. See "Creating the Sample Database" on page 78 for more information.

- 1 Start the database manager by entering the **db2start** command on the server (if it was not automatically started at boot time).
- 2 Issue the following command in the client's Command Center or command line processor to connect the client to the remote database:

```
connect to database_alias user userid using password
```

The database alias (*database_alias*) is an arbitrary local nickname for the remote database on the client. If you did not provide an alias when configuring the client, the default is the same as the database name (*database_name*). The database alias (*database_alias*) is the name that you use when connecting to a database from a client.

For more information, see “Connecting to a Database” on page 79.

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the SERVER. If the database manager is configured for CLIENT authentication, the *userid* and *password* must be valid on the client.

If the connection is successful, you will get a message showing the name of the database to which you have connected. You are now able to retrieve data from that database. For example, to retrieve a list of all the table names listed in the system catalog table, enter the following SQL command in the Command Center or command line processor:

```
"select tablename from syscat.tables"
```

When you are finished using the database connection, enter the **connect reset** command to end the database connection.



You are ready to start using the DB2. See Chapter 10, “Getting Started with DB2 Universal Database” on page 77 for details.

Part 5. Using DB2 Universal Database

Chapter 10. Getting Started with DB2 Universal Database

This section describes how to use DB2 Universal Database in a variety of ways.



Go to the section that gives instructions for the tasks that you want to perform.

- “Logging on to the System.”
 - “Creating the Sample Database” on page 78.
 - “Connecting to a Database” on page 79.
 - “Working with the System Administrative Group” on page 80.
 - “Software Registration” on page 81.
 - “Upgrading from a Try-and-Buy License” on page 82.
-

Logging on to the System

To use DB2, log on with a valid DB2 user ID that has the appropriate authority level for the commands you want to execute. If the system administrative group parameter (*sysadm_group*) is defined in the database manager configuration file, the user performing administration tasks must belong to that group to inherit System Administrative (SYSADM) authority on the instance.

The *sysadm_group* parameter is not set when DB2 is first installed; therefore, the following criteria are used to determine default SYSADM authority:

OS/2

Any valid DB2 user ID which belongs to the UPM Administrator or Local Administrator group.

Windows 95 and Windows 98

Any Windows 95 or Windows 98 username that is a valid DB2 user ID.

Windows NT

A valid DB2 user account which belongs to either the local Administration group or the Domain Admins group. Refer to the *Administration Guide* for more information.

See Appendix D, “Naming Rules” on page 233 for a list of naming restrictions.

As well as using the local operating system security to authenticate users, there is also the option of using DCE security. Refer to the *Administration Guide* for information on DCE security.

Creating the Sample Database

If you did not create the sample database after installation, you might want to create the sample database, called `SAMPLE`, that is provided with DB2 Universal Database. It contains tables, views, and fictitious data that you can use to verify that DB2 is correctly installed or test queries or commands. For a detailed description of the sample database, refer to the *Administration Guide*.

The sample database requires approximately 9 MB of disk space and by default is created in the drive where DB2 is installed. It is automatically cataloged with the database alias `SAMPLE` when it is created.



You must ensure that the drive where the sample database will be created is available on each machine in your partitioned database system.

To create the sample database, use one of the following methods:

- If you installed the DB2 tools, click on the **Create the SAMPLE Database** graphic button in the First Steps main panel. (To start First Steps, click on **Start** and select **Programs->DB2 for Windows NT->First Steps**.)

This command may take a few minutes to process. When the sample database has been created, you will receive a completion message. Click on **OK**.

- Enter the **db2samp1** command from the DB2 Command Window or in the operating system's command prompt. By default, the sample database will be created on the drive where DB2 was installed (you can optionally create the database on a different drive). For example, to create it on the F: drive, type:

```
db2samp1 F:
```

This command may take a few minutes to process. There is no completion message; when the command prompt returns, the process is complete.



To remove the `SAMPLE` database, enter the following command in the Command Center or the command line processor:

```
drop database sample
```

Connecting to a Database

You need to connect to a database before you can use SQL statements to query or update it. The `CONNECT` statement associates a database connection with a user ID.

A database is created in an instance using one of three different authentication types: `CLIENT`, `SERVER`, or `DCS`. If an authentication type is not specified when the instance is created, the default `SERVER` authentication is used. Depending on the authentication type and where the database resides, you connect to remote databases differently. While DCE authentication is an option, you cannot specify it when you create an instance. Refer to *Administration Guide* for more information on all authentication types.

Connecting to a local database

To connect to a local database called `SAMPLE`, enter the following command in the Command Center or the command line processor:

```
connect to sample
```

Connecting to a remote database

- To connect a client to a remote database called `SAMPLE`, using `SERVER` authentication, enter the following command in the Command Center or the command line processor:

```
connect to sample user userid using password
```

(Be sure to choose values for *userid* and *password* that are valid on the server system.)

- To connect a client to a remote database called `SAMPLE`, using `CLIENT` authentication, enter the following command in the Command Center or the command line processor:

```
connect to sample
```



If you are connecting to the database from an *OS/2* client, and are not currently logged on, you will be prompted to log on using a user ID and password.

If you are connecting to a database from a *Windows NT* client using `CLIENT` authentication, and you provide a user ID and password on the **connect** command, you must start the DB2 for Windows NT Security Service on the system. The Security Service is installed by DB2 and set up to run as a Windows NT service; however, it is not started automatically. To start the DB2 Security Service, enter the **net start db2ntsecserver** command.

When a successful connection is made, a message similar to the following is displayed:

Database product = DB2/NT 5.2.0
SQL authorization ID = USERID
Local database alias = SAMPLE

If you receive an error message, make sure that the SAMPLE database exists on the server and the database manager was started on the server.

Refer to the *Administration Guide* for more information on authentication.

Working with the System Administrative Group

By default, System Administrative (SYSADM) authority is granted to the following:

OS/2

Any valid DB2 user ID which belongs to the Administrator or Local Administrator group.

Windows 95 and Windows 98

Any Windows 95 or Windows 98 user.

Windows NT

Any valid DB2 user account which belongs to the local Administrators group.

You can change the users who have SYSADM privileges for each DB2 instance by changing the *sysadm_group* parameter; but before you do, ensure that the group exists.

Granting Users Authorization

DB2 Extended Enterprise Edition requires that all machines that participate in a partitioned database system belong to a single Windows NT domain. Only a domain account should be used when accessing databases in a partitioned database system.

For applications to access databases, DB2 performs two types of checking:

Authentication Ensures that the user account and password are valid.

Authorization Ensures that the user has sufficient authority to perform a task.

This section discusses the authorization process used in a Windows NT domain environment. For more information on authentication and authorization, refer to the *Administration Guide*.

When performing administration tasks (such as cataloging the database directory or creating a database) System Administrative (SYSADM) authority is required. By default,

any user belonging to the Administrators group where the user account is defined has SYSADM authority.

In a Windows NT domain environment, only domain users that belong to the Administrators group at the Primary Domain Controller (PDC) have SYSADM authority. Adding a domain user to the local Administrators group on the server machine does not grant the domain user SYSADM authority, since DB2 always performs authorization at the machine where the account is defined. To avoid adding a domain user to the Administrators group of the Primary Domain Controller, perform the following steps:

1. Create a new global group. The name of the global group must be eight characters or less and comply with DB2's naming rule. For more information, see Appendix D, "Naming Rules" on page 233.
2. Add a domain user to this global group.
3. Grant SYSADM authority to this global group by entering the following command:

```
db2 update database manager configuration using sysadm_group global_group_name
```

where *global_group_name* is the name of the global group that you created.

Each database partition server is implemented as an Windows NT service. For security reasons, a domain account must be specified as the Logon Account for this service so that each database partition server can access data on a network drive.

A domain account is specified whenever a new database partition server is created; for example, during installation, you were prompted to enter a domain account for the new database partition server. The domain user account specified for the database partition server must have SYSADM authority.



The **db2start** command requires local administrator authority and does not require SYSADM authority.

Software Registration

The **Registration** icon, located in the **DB2** folder, is used to access and update the IBM Software Customer Profile and to register IBM Software with IBM.

- 1 Click on **Start**, and select **Programs->DB2 for Windows NT->Registration**.

The customer profile contains identification and demographic information about the user of each IBM software product. The software registration tool contains information and instructions on how to register IBM software by modem, telephone, Internet, FAX, or postal service.

- 2 Enter the appropriate information and complete the IBM Software Registration Smart Guide. Click on **Help** push button if you need assistance.

Upgrading from a Try-and-Buy License

Try-and-Buy versions are available for some DB2 products. If you install a product originally as Try-and-Buy, you do not need to reinstall the same product to upgrade the license. Simply add the license from the DB2 CD-ROM that comes in the product box.

Note: *Your Proof of Entitlement* and *License Information* booklets identify the products for which you are licensed.

The following steps describe how to upgrade a license:

- 1 Click on **Start** and select **Programs->License Use Runtime-> Nodelock Administration Tool** to start the Nodelock Administration Tool.
- 2 Select **Products->New** from the menu toolbar.
- 3 Click on the **Import** push button.
- 4 Select the **db2eent.lic** license file from the `x:\db2\lic` directory, where `x:` represents the CD-ROM drive.
- 5 Click on **OK** to return to the main screen.

Chapter 11. Working with Instances

The information in this section describes DB2 instances and how to work with them.

An instance is a logical database manager environment where you catalog databases and set configuration parameters. Depending on your needs, you can create more than one instance. You can use multiple instances to do the following:

- Use one instance for a development environment and another instance for a production environment.
- Tune an instance for a particular environment.
- Restrict access to sensitive information.
- Control the assignment of SYSADM, SYSCTRL, and SYSMANT authority for each instance.
- Tune an instance for a particular environment.
- Limit the impact of an instance failure. In the event of an instance failure, only one instance is affected. Other instances can continue to function normally.

Each additional instance requires additional system resources (virtual storage and disk space) and more administration.

Instance Directory

The instance directory stores all information that pertains to a database instance. You cannot change the location of the instance directory once it is created. The directory contains:

- The database manager configuration file
- The system database directory
- The node directory
- The DB2 diagnostic file (db2diag.log)
- The node configuration file
- Any other files that contain debugging information, such as the exception/register dump or the call stack for the DB2 processes.

The instance directory is located in the `x:\sql11ib` directory, where `x:` is the drive where you installed DB2.

In a partitioned database system, the instance directory is shared between all database partition servers belonging to the instance.

Creating Instances

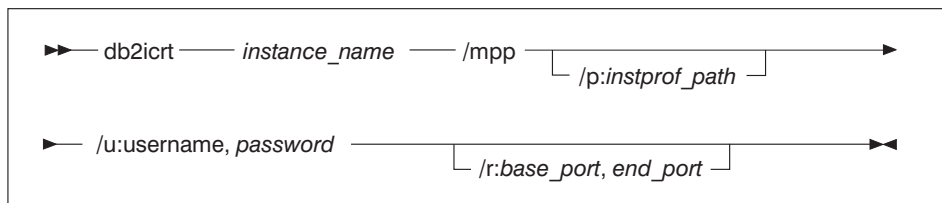
To create an instance, enter the **db2icrt** command. The machine that you enter the **db2icrt** command on will become the instance-owning machine (node 0).



You can choose to update an existing single-partition instance to the multipartition format using the **db2iupdt** command. For more information, refer to the Chapter 17, "Migrating from Previous Versions and Releases" on page 141.

To create an instance, perform the following steps:

- 1 Log on with a user account that belongs to the local Administrators group.
- 2 The syntax of the **db2icrt** command is as follows:



where:

instance_name

Is the name of the instance.

/mpp

Specifies that this instance is for use in a partitioned database system.

/p: instprof_path

Specifies the instance profile path. To specify a different instance profile path, you must create a shared drive or directory. For example: /p:\\machineA\DB2-DB2MPP

This parameter is optional. If you do not specify the instance profile path, the instance directory is created under the x:\sql11b directory (where x: is the drive where you installed DB2), and given the shared name DB2_INSTANCE_NAME. Read and write permissions will automatically be granted to everyone in the domain. You can change the permissions to restrict access to that directory.

/u:domain_name\username, password

Specifies the domain, logon account name, and password for the DB2 service. This parameter is required to create the multi-partition instance.

lr: base_port, end_port

Specifies the TCP/IP port range for the FCM; for example: /r:9010,9015. This parameter is optional; however, if you choose to specify the TCP/IP port range, you must ensure that this port range is available on all machines that will make up the partitioned database system.

For example, to create a multipartition instance called TESTMPP, enter the following command:

```
db2icrt TESTMPP /mpp /u:QBPAULZ\paulz,g1reeky
```

The **db2icrt** command grants the user account used to create the instance the following Windows NT user rights:

- Act as part of the operating system
- Create a token object
- Increase quota
- Log on as a service
- Replace a process level token

The instance requires these user rights to access the shared drive, authenticate the user account, and run DB2 as a Windows NT service.

- 3** We recommend that an Administration Server exists on any instance-owning machine. For more information, see Chapter 12, "Using the Administration Server" on page 95.

Listing Instances

To get a list of all the instances that are available on a system, enter:

```
db2ilist
```

To determine which instance applies to the current session, enter:

```
db2 get instance
```

Listing Database Partition Servers in an Instance

Use the **db2nlist** command to obtain a list of database partition servers that participate in an instance.

The syntax of the command is as follows:

```
db2nlist [/i:instance_name] [/s]
```

where:

- /i:instance_name** Is the instance name; the default is the current instance (set by the *DB2INSTANCE* environment variable).
- /s** Displays the status of each node. The status of each node can be one of the following:
- starting
 - running
 - stopping
 - stopped

For example, to list the status of every node in the **DB2MPP** instance, enter the following command:

```
db2nlist /i:DB2MPP /s
```

Starting and Stopping an Instance

You must start an instance before you can perform the following tasks:

- Connect to a database on the instance.
- Precompile an application.
- Bind a package to a database.
- Access host databases.

To start an instance:

- 1 Log on to the system with a user account that belongs to the local Administrators group on each machine in your partitioned database system.



If you have trouble logging on to the system, see “Logging on to the System” on page 77.

2 Use one of these methods to start the instance:

- From the Control Center, select the instance that you want to start, click with the right mouse button, and select the **Start** option from the pop-up menu.
- From a command line, enter the **db2start** command.

Note: The **db2start** command, or **start** option, will start the instance as defined by the *DB2INSTANCE* environment variable. For more information, see “Setting the Current Instance” on page 92.

To stop an instance use one of these methods:

- From the Control Center, select the instance that you want to stop, click with the right mouse button, and select the **Stop** option from the pop-up menu.
- From a command line, enter the **db2stop** command.

Note: If command line processor sessions are attached to an instance, you must enter the **terminate** command in that session to end each session before running the **db2stop** command or selecting the **stop** option from the Control Center. The **db2stop** command, or **stop** option, will stop the instance defined by the *DB2INSTANCE* environment variable. For more information, see “Setting the Current Instance” on page 92.



To start or stop a specific node in a partitioned database system, you can:

- From the Control Center:
 1. Select the instance that contains the node that you want to stop or start and click on the right mouse button.
 2. Select the **Open->Nodes** option.
 3. Select the node the you want to stop or start and click on the right mouse button.
 4. Select the **Stop** or **Start** option.
 5. Click on **OK**
- Enter the **db2start** or **db2stop** commands with the *nodenum* option

For example, to stop node 3 in your partitioned database system, enter the following command:

```
db2stop nodenum 3
```

Adding a Database Partition Server to an Instance

Use the **db2nrcrt** command to add a database partition server (node) to an instance. If you add a database partition server to a machine where an instance already exists, the **db2nrcrt** command adds a database partition server as a logical node to the machine. If you add a database partition server to a machine where an instance does not exist, the instance is added to this machine, and this machine becomes a new physical node.

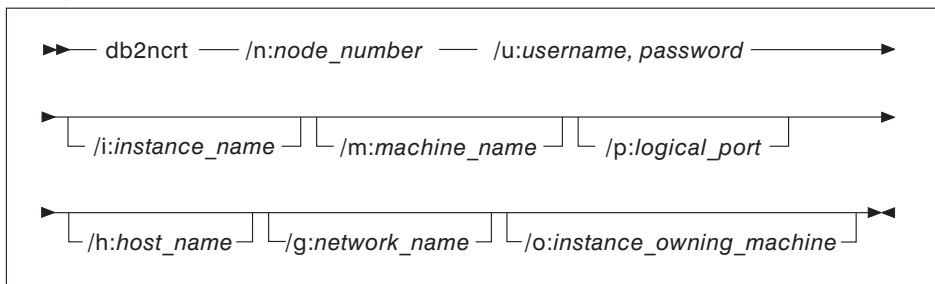


Do not use the **db2nrcrt** command if there are databases in an instance.

Instead, you should use the **db2start addnode nodenum** command. This ensures that the database is correctly added to the new database partition server. If you want to add a node to an instance in which a database has been created, refer to the *Administration Guide*.

Do not edit the `db2nodes.cfg` file, since changing the file may cause inconsistencies in the partitioned database system.

The syntax of the **db2nrcrt** command is as follows:



where:

/n:node_number

Is a unique node number to identify the database partition server. This number can be from 1 to 999. This parameter is required.

/u:domain_name\username, password

Is the domain, logon account name, and password for the DB2 service. This parameter is required.

/i:instance_name

Is the instance name. This parameter is optional. If you do not specify this parameter, the default is the current instance.

/m:machine_name

Is the computer name of the Windows NT workstation on which the node resides. This parameter is required if you add the database partition server on a remote machine.

/p:logical_port

Specifies the logical port number used for the database partition server. This parameter is optional. If you do not specify this parameter, the logical port number assigned will be 0.

If you are creating a logical node, you *must* specify this parameter and select a logical port number that is not in use. Note the following restrictions:

- On every machine, there must be a database partition server that has a logical port 0.
- The port number cannot exceed the port range reserved for FCM communications in the services file in `x:\winnt\system32\drivers\etc\` directory. For example, if you reserve a range of 4 ports for the current instance, then the maximum port number would be 3 (ports 1, 2, and 3; port 0 is for the default logical node).

/h:host_name

Specifies the TCP/IP host name that is used by FCM for internal communications (port is not a node). This parameter is required if you add the database partition server on a remote machine.

/g:network_name

Specifies the network name for the database partition server. This parameter is optional. If you do not specify this parameter, DB2 will use the first IP address that it detects on your system.

Use this parameter if you have multiple IP addresses on a machine and you want to use a specific IP address for the database partition server. You can enter the *network_name* parameter using the network name or IP address.

/o:instance_owning_machine

Is the computer name of the instance-owning machine; the default is the local machine. This parameter is required when the **db2nrcrt** command is invoked on any machine that is not the instance-owning machine.

For example, if you want to add a new database partition server to the instance *TESTMPP* (so that you are running multiple logical nodes) on the instance-owning machine *SHAYER*, and you want this new node to be known as node 2 using logical port 1, enter the following command:

```
db2nrcrt /n:2 /u:QBPAULZ\paulz,g1reeky /i:TESTMPP /m:TEST /p:1 /o:SHAYER
```

Changing the Database Partition Server Configuration

Use the **db2nchg** command to do the following:

- Move the database partition server (node) from one machine to another.
- Change the TCP/IP host name of the machine.

- Use a different logical port number.
- Use a different network name for the database partition server (node).



You can only use this command if the database partition server is stopped.

The syntax of the **db2nchg** command is as follows:

```

db2nchg - /n:node_number [ /i:instance_name ] [ /u:user, password ]
[ /p:logical_port ] [ /h:hostname ]
[ /m:machine_name ] [ /g:network_name ]

```

where:

/n:node_number

Specifies the node number of the database partition server's configuration you want to change. This parameter is required.

/i:instance_name

Specifies the instance that this database partition server participates in. This parameter is optional. If you do not specify this parameter, the default is the current instance.

/u:domain_name\username, password

Specify this option to change the domain, logon account name, and password for the DB2 service. This parameter is optional. If you do not specify this parameter, the domain, logon account, and password will remain the same.

/p:logical_port

Specify this option to change the logical port for the database partition server. This parameter must be specified if you move the database partition server to a different machine. If you do not specify this parameter, the logical port number will remain unchanged.

/h:host_name

Specify this option to change the TCP/IP hostname used by FCM for internal communications. This parameter is optional. If you did not specify this parameter, the hostname is unchanged.

/m:machine_name

Specify this option to move the database partition server to another machine. The database partition server can only be moved if there are no existing databases in the instance. This parameter is optional.

/g:network_name

Specify this option to change the network name for the database partition server. This parameter is optional.

Use this parameter if you have multiple IP addresses on a machine and you want to use a specific IP address for the database partition server. You can enter the *network_name* using the network name or IP address.

For example, to change the logical port assigned to node 2, which participates in the instance *TESTMPP*, to use the logical port 3, enter the following command:

```
db2nchg /n:2 /i:TESTMPP /p:3
```

Dropping a Database Partition Server from an Instance

Use the **db2ndrop** command to drop a database partition server from an instance that has no databases. If you drop a database partition server, its node number can be reused for a new database partition server.



You can only use this command if the database partition server is stopped.

Exercise caution when you are dropping database partition servers from an instance. If you drop the instance-owning database partition server (node 0) from the instance, the instance will become unusable. If you want to drop the instance, use the **db2idrop** command. See “Removing Instances” on page 93 for more information.



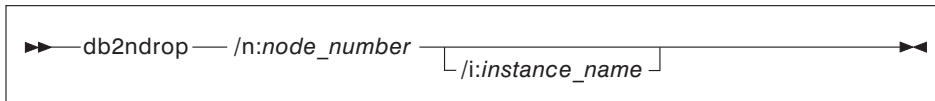
Do not use the **db2ndrop** command if there are databases in this instance.

Instead, you should use the **db2stop drop nodenum** command. This ensures that the database partition server is correctly removed from the partition database system. If you want to drop a node in an instance where a database exists, refer to the *Administration Guide*.

Do not edit the `db2nodes.cfg` file, since changing the file may cause inconsistencies in the partitioned database system.

If you want to drop a node that is assigned the logical port 0 from a machine that is running multiple logical nodes, you must drop all the other nodes assigned to the other logical ports before you can drop the node assigned to logical port 0. Each database partition server must have a node assigned to logical port 0.

The syntax of the **db2ndrop** command is as follows:



where:

/n:node_number Is the unique node number to identify the database partition server. This parameter is required.

/i:instance_name Is the instance name. This parameter is optional. If you do not specify this parameter, the default is the current instance.

For example, assume the **db2nlist** command returned the following output:

```
List of nodes for instance "DB2MPP" is as follows:
Node: "0" Host: "SHAYER" Machine: "SHAYER" Port: "0"
Node: "1" Host: "PAULZ" Machine: "GREECE" Port: "0"
Node: "2" Host: "PAULZ" Machine: "GREECE" Port: "1"
Node: "3" Host: "DENISEJ" Machine: "DENISEJ" Port: "0"
```

Since both nodes 1 and 2 are running on the same machine, and node 1 is using logical port 0, you would first need to drop node 2 before dropping node 1. To drop node 2 which participated with an instance called SHAYER, enter the following command:

```
db2ndrop /n:2 /i=SHAYER
```

Setting the Current Instance

When you enter commands to start or stop an instance's database manager, DB2 applies the command to the current instance. DB2 determines the current instance as follows:

- Read the setting for the *DB2INSTANCE* environment variable for the current session. To set the *DB2INSTANCE* environment variable, enter:


```
set db2instance=new_instance_name
```

 where *new_instance_name* represents the instance that you want to work with.
- If the *DB2INSTANCE* environment variable is not set for the current session, DB2 uses the setting for the *DB2INSTANCE* environment variable from the system environment variables:
 - On Windows NT, system environment variables are set in the System Environment. See "Setting Your Environment on Windows 32-Bit Operating Systems" on page 122 for instructions on setting *DB2INSTANCE*.
- If the *DB2INSTANCE* environment variable is not set at all, DB2 uses the registry value, *db2instdef*.

To set the `db2instdef` registry value globally, enter:

```
db2set db2instdef=new_instance_name -g
```

where `new_instance_name` represents the instance that you want to be the default for your system.

For more information on setting registry values, refer to Chapter 15, “Controlling Your DB2 Environment” on page 119.

To determine which instance applies in the current session, enter:

```
db2 get instance
```

Running Multiple Instances Concurrently

To run multiple instances concurrently, use one of the following methods:

- From the Control Center, select the other instance that you want to start, click the right mouse button, and select the **Start** option.
- From the command line do the following:

- 1 Set the `DB2INSTANCE` variable to the name of the other instance that you want to start, enter:

```
set db2instance=another_instance_name
```

- 2 Start the instance by entering the **db2start** command.

See Chapter 15, “Controlling Your DB2 Environment” on page 119 for information on configuring each instance with different environment variables.

Removing Instances



Ensure that there are no applications using the instance before removing it.

To remove an instance, perform the following steps:

- 1 Close all applications that are currently using the instance.

- 2 Stop the Command Line Processor by entering the **db2 terminate** command in each DB2 command window.
- 3 Stop the instance by entering the **db2stop** command.
- 4 Back up the instance directory indicated by the *DB2INSTPROF* registry value.
- 5 Enter the **db2idrop** command as follows:
 `db2idrop instance_name`
 where *instance_name* represents the instance you want to drop.

Chapter 12. Using the Administration Server

The Administration Server enables you to administer DB2 server systems from a local or remote client, using the Control Center. You must have an Administration Server running on the server if you want to use the Client Configuration Assistant or the Control Center to administer a DB2 server. For example, you can start or stop DB2, or schedule jobs to be run at the server from a local or remote system.

The Administration Server is also used to support client configuration using the Client Configuration Assistant, and system reporting using the Control Center. You can create only one Administration Server on a machine.

This section shows you how to manually create an Administration Server. It also describes how to start, stop, and remove an Administration Server, set up an Administration Server to use the CCA and the Control Center, and set an Administration Server's user account.

Creating an Administration Server

The setup program creates an Administration Server on the instance-owning machine during installation.



We recommend that you create an Administration Server on any instance-owning machine.

By adding a system that has an Administration Server to the Control Center or Client Configuration Assistant (CCA), you are designating the corresponding database partition server (that resides on the same machine as the Administration Server) as the connection point to the entire instance. The system specified becomes the coordinator node for requests that are issued to the instance from the Control Center or CCA.

You may want to consider distributing the coordinator function among other database partition servers in the instance. To do this:

1. Create a new Administration Server on the other machines that are in the partitioned database system.
2. Catalog each Administration Server as a separate system in the Control Center or CCA.

3. Catalog the same instance under each new system, and each time specify the same machine that you used to catalog the Administration Server.



If you want a node to act as a coordinator node, but do not want to install an Administration Server on that node, you can manually catalog the node using the Client Configuration Assistant, the Control Center, or the command line processor.

Creating additional Administration Servers on other machines allows the CCA or the Control Center to set up multiple nodes as coordinator nodes using DB2 Discovery. This distributes the connection workload among many nodes. Further information is provided below; also see Chapter 18, “Using the Client Configuration Assistant” on page 159.

You can create additional Administration Servers using the **db2admin** command. To do this, perform the following steps:

- 1 Log on to the machine that you want to create the Administration Server on using an account that has local Administrator authority.
- 2 Create a Windows NT account that has local Administrator authority. This account will be used by the Administration Server. Ensure that the username of the account adheres to the DB2 naming convention, for example, *db2admin*. See Appendix D, “Naming Rules” on page 233 for further information.

When creating the account for the Administration Server, note the following:

- Do not use the account for the Administration Server as a User Account.
- Set the password for the account to Password Never Expires.

- 3 Run the following command:

```
db2admin create /user:username /password:password
```

where *username* and *password* are the username and password for the Administration Server.

Once you create the Administration Server, you can verify that it is working by entering the following command:

```
db2admin start
```

Setting Up the Administration Server to Use the CCA and the Control Center

You must configure DB2 Discovery, a feature that is used by the CCA and the Control Center, to retrieve information about systems on your network. Specifically, you may need to update instance lists and the Administration Server configuration to ensure that DB2 Discovery retrieves the correct information.

Update Instance Lists

An Administration Server may not be aware of all the instances in a partitioned database system because instance lists are maintained on a per-machine basis. When an instance is created, only the Administration Server on the instance-owning machine is aware of the instance, because only the instance list on that machine is updated.

Each instance that you create should be known by at least one Administration Server. If you created an instance on a machine that does not have an Administration Server, you can make that instance known to an Administration Server by performing one of the following:

- Create an Administration Server on that machine. See “Creating an Administration Server” on page 95 for more information.
- Enter the **db2ncrt** command to add an Administration Server node to the instance.

This will partition the instance across the new node. For more information on the **db2ncrt** command, see “Adding a Database Partition Server to an Instance” on page 88.

Perform the following steps if you created more than one Administration Server, and you want each Administration Server to be aware of all the instances in your partitioned database system.

1 For each Administration Server

Run the **db2ilist** command on the Administration Server machine to display a list of instances known to this Administration Server.



If this instance list includes the instances that you want the Administration Server to recognize, go to “Update the Administration Server Configuration” on page 98 for detailed information on how to configure Discovery.

2 For each instance that is missing from the instance list in the step above

On the instance-owning machine, run the **db2nlist** command to see if there is an entry for each machine that has an Administration Server. If there isn't an entry for the Administration Server machine, you must run the **db2ncrt** command to add this machine to the instance. For more information on the **db2ncrt** command, see "Adding a Database Partition Server to an Instance" on page 88.

Note: The network shared drive for the instance must be available on the Administration Server machine.

Update the Administration Server Configuration

By default, the setup program sets the *DB2SYSTEM* registry value to the Windows NT computer name. The system names that are retrieved by Discovery are the systems on which an Administration Server resides. Discovery uses these systems as coordinator nodes when connections are established.

There are two ways of updating an Administration Server's configuration:

- If you want to be able to select a coordinator node from a list of DB2 systems, set *DISCOVER=SEARCH* (which is the default) in each Administration Server's configuration file.

When there are multiple Administration Servers, the same instance may appear in more than one system on the CCA or Control Center's interface; however, each system will have a different communications access path to the instances. Users can select different DB2 systems as coordinator nodes for communications and thereby redistribute the workload.

- If you do not want users to be able to select the coordinator node, set *DISCOVER=KNOWN* on all Administration Servers, except set *DISCOVER=SEARCH* on just one Administration Server in the Administration Server configuration. Discovery uses the database partition server where the Administration Server resides as a coordinator node when connections are established.

For example, to set *DISCOVER=SEARCH* in the Administration Server's configuration, enter the following commands:

```
db2 update admin configuration using DISCOVER SEARCH
db2admin stop
db2admin start
```

Starting the Administration Server

To manually start the Administration Server, use the **db2admin** command as follows:

- 1 Log on to the machine using an account that has local Administrator authority.
- 2 From a command window, enter:

```
db2admin start
```

Stopping the Administration Server

To manually stop the Administration Server, use the **db2admin** command as follows:

- 1 Log on to the machine using an account that has local Administrator authority.
- 2 From a command window, enter:

```
db2admin stop
```

Setting the Logon Account for the Administration Server



You should not use the **Services** utility in the **Control Panel** to change the logon account for the Administrator Server, since some of the required access rights will not be set for the logon account. Always use the **db2admin** command to set or change the logon account for the Administration Server.

After creating an Administration Server, you can set or change the logon account using the **db2admin** command as follows:

- 1 Log on to the machine using an account that has local Administrator authority.
- 2 From a command window, enter:

```
db2admin setid username password
```

where *username* and *password* are the user name and password of the account that has local Administrator authority, and that you want to use for the Administration Server.

Removing the Administration Server



If you remove the Administration Server, remote administration, such as communications, backup, or restore, will no longer be possible.

To remove the Administration Server, use the **db2admin** command as follows:

- 1 Log on to the machine using an account that has local Administrator authority.
- 2 Stop the Administration Server by entering the following command:
`db2admin stop`
- 3 Back up the instance directory indicated by the *DB2INSTPROF* registry value.
- 4 Drop the Administration Server by entering the following command:
`db2admin drop`

Chapter 13. Running Your Own Applications

Various types of applications can access DB2 databases:

- Applications developed using the DB2 Software Developer's Kit that include embedded SQL, APIs, stored procedures, user-defined functions or calls to the DB2 CLI.
- ODBC applications such as Lotus Approach.
- JDBC applications and applets.
- Net.Data macros containing HTML and SQL.

An application on a DB2 client can access a remote database without knowing its physical location. The DB2 client determines the location of the database, manages the transmission of the requests to the database server, and returns the results.

In general, to run a database client application, use the following steps:

- 1 Ensure the server is configured and running.
Be sure that the database manager is started on the database server to which the application program is connecting. If it is not, you must issue the **db2start** command at the server before starting the application.
- 2 Ensure that you can connect to the database that the application uses.
- 3 Bind the utilities and the applications to the database. See "Binding Database Utilities" for information about binding the utilities.
- 4 Run the application program.

Binding Database Utilities

You must bind the database utilities (import, export, reorg, the command line processor) and DB2 CLI bind files to each database before they can be used with that database. In a network environment, if you are using multiple clients that run on different operating systems or are at different versions or service levels of DB2, you must bind the utilities once for each operating system and DB2-version combination.

Binding a utility creates a *package*, which is an object that includes all of the information that is needed to process specific SQL statements from a single source file.

The bind files are grouped together in different .lst files in the bnd directory, under the installation directory (typically sql11b). Each file is specific to a server.

How you bind the database utilities to a database depends on your workstation's operating system:

- On OS/2 and Windows 32-bit operating systems, you can use the Client Configuration Assistant:
 - 1 Start the Client Configuration Assistant (CCA).
 - 2 Select the database to which you want to bind the utilities.
 - 3 Click on the **Bind** push button.
 - 4 Select the **Bind DB2 Utilities** radio button.
 - 5 Click on the **Continue** push button.
 - 6 Enter a user ID and password to connect to the database. The user ID must have the authority to bind new packages against the database.
 - 7 Select the utilities you want to bind and click on **OK**.
- On all operating systems, you can use the command line processor:
 - 1 Change to the bnd directory in the install path. For example: x:\sql11b\bnd, where x: is the drive where you installed DB2
 - 2 Connect to the database using the command:

```
connect to database_alias
```

where *database_alias* is the name of the database to which you want to connect.
 - 3 Enter the following commands in the Command Center or the command line processor:

```
"bind @db2ubind.lst messages bind.msg grant public"  
"bind @db2cli.lst messages cllibind.msg grant public"
```

In this example, bind.msg and cllibind.msg are the output message files, and EXECUTE and BINDADD privileges are granted to *public*.
 - 4 Reset the connection to the database by entering the following command:

```
connect reset
```

For more information on the **bind** command, refer to the *Command Reference*.

Notes:

1. The db2ubind.lst file contains the list of bind (.bnd) files required to create the packages for the database utilities. The db2cli.lst file contains the list of bind (.bnd) files required to create packages for the DB2 CLI and the DB2 ODBC driver.
2. Binding may take a few minutes to complete.
3. To bind to databases that reside on OS/390, MVS, VM, or AS/400, refer to the *DB2 Connect User's Guide* for instructions.
4. If you have BINDADD authority, the first time you use the DB2 CLI or ODBC driver, the DB2 CLI packages will be bound automatically.



If the applications that you are using require binding to the database, you can use the Client Configuration Assistant's Bind facility, or the command line processor, to perform the bind action.

Running CLI/ODBC Programs

The DB2 Call Level Interface (CLI) run-time environment and the ODBC driver are included with the DB2 Client Application Enabler. This is contained on the DB2 Client Application Enablers CD-ROM or can be downloaded from the Web page at <http://www.software.ibm.com/data/db2>.

This support enables applications developed using ODBC and DB2 CLI APIs to work with any DB2 server. DB2 CLI application development support is provided by the DB2 Software Developer's Kit (DB2 SDK) which is part of the separately orderable DB2 Application Development Kit product.

Before DB2 CLI or ODBC applications can access DB2, the DB2 CLI packages must be bound on the server. Although this will occur automatically on the first connection if the user has the required authority to bind the packages, it is recommended that the administrator do this first with each version of the client on each platform that will access the server. See "Binding Database Utilities" on page 101 for specific details.

The following general steps are required on the client system to give DB2 CLI and ODBC applications access to DB2 databases. These instructions assume that you have successfully connected to DB2 using a valid user ID and password. Depending on the platform many of these steps are automatic. For complete details, see the section that deals specifically with your platform.

- 1 Use the CCA to add the database (if you have separate client and server machines) so that its instances and databases can be made known to the Control Center, then add the instances and databases for that system. (Your local system

is represented by **Local** icon.) If you do not have access to this program you can use the **catalog** command in the command line processor.

- 2 On all platforms other than OS/2 and Windows 3.x, the DB2 CLI/ODBC driver is automatically installed when the DB2 Client Application Enabler is installed, and therefore nothing needs to be done. On OS/2 and Windows 3.x you must use the **Install ODBC Driver** icon to install both the DB2 CLI/ODBC driver and the ODBC driver manager.
- 3 To access the DB2 database from ODBC:
 - a The Microsoft, Visigenic, or other ODBC Driver Manager must already be installed (this is done by default during the installation of DB2).
 - b The DB2 databases must be registered as ODBC data sources. The ODBC driver manager does not read the DB2 catalog information; instead it references its own list of data sources.
 - c If a DB2 table does not have a unique index then many ODBC applications will open it as read-only. A unique index should be created for each DB2 table that is to be updated by an ODBC application. Refer to the **CREATE INDEX** statement in the *SQL Reference*. Using the Control Center you would alter the settings of the table, then select the **Primary Key** tab and move one or more columns from the available columns list over to the primary key columns list. Any column you select as part of the primary key must be defined as NOT NULL.
- 4 Various CLI/ODBC Configuration Keywords can be set to modify the behavior of DB2 CLI/ODBC and the applications using it.
- 5 If you followed the above steps to install ODBC support, and added DB2 databases as ODBC data sources, your ODBC applications will now be able to access them.

Platform Specific Details for CLI/ODBC Access



The platform specific details on how to give DB2 CLI and ODBC applications access to DB2 are divided into the following categories:

- "Windows 3.x, Windows 95, Windows 98, and Windows NT Client Access to DB2 using CLI/ODBC" on page 105
 - "OS/2 Client Access to DB2 using CLI/ODBC" on page 107
-

Windows 3.x, Windows 95, Windows 98, and Windows NT Client Access to DB2 using CLI/ODBC

Before DB2 CLI and ODBC applications can successfully access a DB2 database from a Windows client, perform the following steps on the client system:

- 1** The DB2 database (and node if the database is remote) must be cataloged. To do so, use the CCA (or the command line processor). See “Configuring Database Connections” on page 159 for more details.

For more information refer to the on-line help in the CCA (or the **CATALOG DATABASE** and **CATALOG NODE** commands in the *Command Reference*).

- 2** Verify that the Microsoft ODBC Driver Manager and the DB2 CLI/ODBC driver are installed. On Windows 32-bit operating systems they are both installed with DB2 unless the ODBC component is manually unselected during the install. On Windows 3.x you must use the **Install ODBC Driver** icon to install the Microsoft ODBC Driver Manager and the DB2 CLI/ODBC driver.

To verify that they both exist on the machine:

- a** Run the Microsoft ODBC Administrator from the icon in the Control Panel, or issue the appropriate command from the command line: **odbcad32.exe** for Windows 32-bit operating systems, **odbcadm.exe** on Windows 3.x.
- b** Click on the **Drivers** push button, or the **ODBC Drivers** tab (depending on the platform).
- c** Verify that "IBM DB2 ODBC Driver" is shown in the list.

If either the Microsoft ODBC Driver Manager or the IBM DB2 CLI/ODBC driver is not installed, then rerun the DB2 install and select the ODBC component on Windows 32-bit operating systems, or double-click on the **Install ODBC Driver** icon in the DB2 program group in Windows 3.x.

- 3** Register the DB2 database with the ODBC driver manager as a *data source*. On Windows 32-bit operating systems you can make the data source available to all users of the system (a system data source), or only the current user (a user data source). Use either of these methods to add the data source:

- Using the CCA:
 - a** Select the DB2 database alias that you want to add as a data source.
 - b** Click on the **Properties** push button. The Database Properties window opens.
 - c** Select the **Register this database for ODBC** check box.
 - d** On Windows 32-bit operating systems you can use the radio buttons to add the data source as either a user or system data source.

- Using the Microsoft **32-bit ODBC Administration tool**, which you can access from the icon in the Control Panel or by running **odbcad32.exe** from the command line:
 - a** On Windows 32-bit operating systems the list of user data sources appears by default. If you want to add a system data source click on the **System DSN** button, or the **System DSN** tab (depending on the platform).
 - b** Click on the **Add** push button.
 - c** Double-click on the IBM DB2 ODBC Driver in the list.
 - d** Select the DB2 database to add and click on **OK**.
- On Windows 32-bit operating systems there is a command that can be issued in the command line processor to register the DB2 database with the ODBC driver manager as a data source. An administrator could create a command line processor script to register the required databases. This script could then be run on all of the machines that require access to the DB2 databases through ODBC.

The *Command Reference* contains more information on the CATALOG command:

```
CATALOG [ user | system ] ODBC DATA SOURCE
```

4 Optional: Configure the DB2 CLI/ODBC driver using the CCA:

- a** Select the DB2 database alias you want to configure.
- b** Click on the **Properties** push button. The Database Properties window opens.
- c** Click on the **Settings** push button. The CLI/ODBC Settings window opens.
- d** Click on the **Advanced** push button. You can set the configuration keywords in the window that opens. These keywords are associated with the database *alias name*, and affect all DB2 CLI/ODBC applications that access the database. The online help explains all of the keywords, as does an appendix in the *Installing and Configuring DB2 Clients* manual.

5 Optional: Using 16-bit ODBC applications:

Although not directly related to DB2, some users have experienced problems when running 16-bit ODBC applications on Windows 32-bit operating systems.

Both 16- and 32-bit applications use the same 32-bit IBM DB2 CLI/ODBC Driver. The Microsoft ODBC Driver Manager takes care of the conversion between the 16-bit application and the 32-bit ODBC driver.

Some 16-bit applications were shipped with an old set of 16-bit Driver Manager DLLs (that is, before 1995). These DLLs do not function well in the 32-bit environment. The following symptoms may occur when running the old DLLs:

- Traps occur in odbc.dll or other ODBC related DLLs.

- 16-bit applications do not see data sources cataloged using the 32-bit ODBC Administrator (from the Control Panel).

If (and only if) you are experiencing these problems you may want to update the Microsoft 16-bit ODBC Driver Manager DLLs. These files are provided with DB2 in the `SQLLIB\MS0DBC16` subdirectory. Use these new DLLs to replace the older versions that are located in the system subdirectory of the Windows operating system directory.

- 6 If you have installed ODBC access (as described above), you can now access DB2 data using ODBC applications. Start the ODBC application and go to the Open window. Select the **ODBC databases** file type. The DB2 databases that you added as ODBC data sources will be selectable from the list. Many ODBC applications will open the table as read-only unless a unique index exists.

OS/2 Client Access to DB2 using CLI/ODBC

Before DB2 CLI and ODBC applications can successfully access a DB2 database from an OS/2 client, perform the following steps on the client system:

- 1 The DB2 database (and node if the database is remote) must be cataloged. To do so, use the CCA (or the command line processor).

For more information see the on-line help in the CCA or "Configuring Database Connections" on page 159.

- 2 If you are using ODBC applications to access DB2 data, perform the following steps. (If you are using only CLI applications, skip this step and go to the next step.)
 - a Check that the ODBC Driver Manager (Visigenic or Intersolv) and the DB2 CLI/ODBC driver are installed:

- 1 Run the ODBC Administration tool in one of two ways:

- Double-click on the **Control Panel** icon from the **Main** Folder in WIN-OS/2 or the **ODBC** Folder in OS/2, and double-click on the **ODBC Administrator** icon.
- Run **odbcadm.exe** from the command line.

The Data Sources window opens.

- 2 Click on the **Drivers** push button. The Drivers window opens.

- 3 Verify that "IBM DB2 ODBC Driver" is shown in the list.

If either the ODBC Driver Manager or the IBM DB2 CLI/ODBC driver is not installed then double-click on the **Install ODBC Driver** icon in the DB2 folder to install both the DB2 CLI/ODBC driver and the ODBC driver manager.

- b Register the DB2 database with the ODBC driver manager as a *data source* using either of these methods:

- Using the CCA:
 - 1** Select the DB2 database alias that you want to add as a data source.
 - 2** Click on the **Properties** push button.
 - 3** Select the **Register this database for ODBC** check box.
- Using the Visigenic **ODBC Administration tool**, which you can access from the **ODBC** folder or by running **odbcadm.exe** from the command line:
 - 1** Click on the **Add** push button from the Data Sources window. The Add Data Source Window opens.
 - 2** Double-click on the IBM DB2 ODBC Driver in the list.
 - 3** Select the DB2 database to add and click on **OK**.
- 3** Optional: Configure the DB2 CLI/ODBC driver using the CCA:
 - a** Select the DB2 database alias you want to configure.
 - b** Click on the **Properties** push button. The Database Properties window opens.
 - c** Click on the **Settings** push button. The CLI/ODBC Settings window opens.
 - d** Click on the **Advanced** push button. You can set the configuration keywords in the window that appears. These keywords are associated with the database *alias name*, and affect all DB2 CLI/ODBC applications that access the database. The online help explains all of the keywords, as does an appendix in the *Installing and Configuring DB2 Clients* manual.
- 4** If you have installed ODBC access (as described above), you can now access DB2 data using ODBC applications. Start the ODBC application and go to the Open window. Select the **ODBC databases** file type. The DB2 databases that you added as ODBC data sources will be selectable from the list. Many ODBC applications will open the table as read-only unless a unique index exists.

Detailed Configuration Information

The section “Platform Specific Details for CLI/ODBC Access” on page 104 should provide you with all of the information you require. *Installing and Configuring DB2 Clients* includes additional information on setting up and using DB2 CLI and ODBC applications. This information is useful where DB2 tool support is not available, and for administrators who require more detailed information.

The following topics are covered in the *Installing and Configuring DB2 Clients* manual:

- How to Bind the DB2 CLI/ODBC Driver to the Database

- How to Set CLI/ODBC Configuration Keywords
- Configuring db2cli.ini

Running Java Programs

You can develop Java programs to access DB2 databases with the appropriate Java Development Kit (JDK) on AIX, HP-UX, OS/2, SCO UnixWare, Solaris, or Windows 32-bit operating systems. The JDK includes Java Database Connectivity (JDBC), a dynamic SQL API for Java.

DB2 JDBC support is provided by the DB2 Client Application Enabler (DB2 CAE). With DB2 JDBC support you can build and run JDBC applications and applets. These contain dynamic SQL only, and use a Java call interface to pass SQL statements to DB2.

The DB2 Software Developer's Kit (DB2 SDK) provides support for Java embedded SQL (SQLJ). With DB2 SQLJ support and DB2 JDBC support you can build and run SQLJ applications and applets. These contain static SQL and use embedded SQL statements that are bound to the DB2 database.

Java can also be used on the server to create JDBC and SQLJ stored procedures and user-defined functions (UDFs).

Building and running different types of Java programs requires support from different components of DB2:

- JDBC applications do not require any other component of DB2 on the client to be built. To run JDBC applications requires the DB2 Client Application Enabler (CAE) to connect to DB2.
- To build SQLJ applications requires the DB2 SDK. To run SQLJ applications requires the DB2 CAE to connect to DB2.
- JDBC applets do not require any other DB2 component on the client to be built or run.
- SQLJ applets require the DB2 SDK to be built. They do not require any other DB2 component to be run on the client.

For detailed information on building and running JDBC and SQLJ programs on UNIX platforms see *Building Applications for UNIX Environments*. Detailed information on building and running JDBC programs on Windows and OS/2 can be found in *Building Applications for Windows and OS/2 Environments*. Building and running SQLJ programs on Windows and OS/2 is explained in the *What's New* book. For more information on DB2 programming in Java, refer to the *Embedded SQL Programming Guide*, chapter 15, "Programming in Java". This covers creating and running JDBC

applications, applets, stored procedures and UDFs. Information on SQLJ applications, applets, stored procedures and UDFs can be found in the *What's New* book.

For the latest, updated DB2 Java information, visit the Web Page at:

<http://www.software.ibm.com/data/db2/java>

Configuring the Environment

To build and run DB2 Java programs, you need to install and configure the appropriate version of the Java Development Kit (JDK) on your development machine:

AIX:	The Java Development Kit (JDK) Version 1.1 for AIX from IBM.
HP-UX:	The HP-UX Developer's Kit for Java Release 1.1 from Hewlett-Packard.
OS/2:	The Java Development Kit (JDK) Version 1.1 for OS/2 from IBM.
SCO UnixWare:	No installation necessary: Java Development Kit (JDK) Version 1.1.3 for SCO UnixWare is already installed in /usr/java.
Solaris:	The Java Development Kit (JDK) Version 1.1.4 for Solaris, and the Solaris Native Thread pack, from Sun Microsystems.

Windows 32-bit operating systems:

The Java Development Kit (JDK) Version 1.1 for Win32 from Sun Microsystems.

For information on installing and configuring any of the above JDKs, please refer to:

<http://www.software.ibm.com/data/db2/java>

For all supported platforms, you must also install and configure the DB2 Client Application Enabler for your platform from the DB2 Client Pack. It must be Version 2.1.2 or later, except for SCO UnixWare which must be Version 5.2 or later.

To run DB2 Java stored procedures or UDFs, you also need to update the DB2 database manager configuration to include the path where the JDK version 1.1 is installed on your development machine. You can do this by entering the following on the command line:

```
db2 update dbm cfg using JDK11_PATH /home/smith/jdk11
```

where /home/smith/jdk11 is the path where the JDK version 1.1 is installed.

You can check the DB2 database manager configuration to verify the correct value for the JDK11_PATH field by entering the following command:

```
db2 get dbm cfg
```

You may want to pipe the output to a file for easier viewing. The `JDK11_PATH` field appears near the beginning of the output. For more information on these commands, see the *Command Reference*.

Note: On Solaris, some Java Virtual Machine implementations do not work well in programs that run in a "setuid" environment. The shared library that contains the Java interpreter, `libjava.so`, may fail to load. As a workaround, you can create symbolic links for all needed JVM shared libraries in `/usr/lib`, with a command similar to the following (depending on where Java is installed on your machine):

```
ln -s /opt/jdk1.1.3/lib/sparc/native_threads/*.so /usr/lib
```

For more information on this and other workarounds available, please visit:

<http://www.software.ibm.com/data/db2/java/v5/faq.html>

To run Java programs, the following environment variables are automatically updated during DB2 installation on Windows and OS/2, and during instance creation on UNIX platforms.

On UNIX platforms:

- `CLASSPATH` includes "." and the file `sql1lib/java/db2java.zip`
- On AIX, SCO UnixWare, and Solaris: `LD_LIBRARY_PATH` includes the directory `sql1lib/lib`; on HP-UX: `SHLIB_PATH` includes the directory `sql1lib/lib`
- On Solaris only: `THREADS_FLAG` is set to "native"

On Windows and OS/2 platforms:

- `CLASSPATH` includes "." and the file `%DB2PATH%\java\db2java.zip`

In order to build and run SQLJ programs, `CLASSPATH` is also automatically updated to include these files:

On UNIX platforms:

- `sql1lib/java/sqlj.zip` (required to build SQLJ programs)
- `sql1lib/java/runtime.zip` (required to run SQLJ programs)

On Windows and OS/2 platforms:

- `%DB2PATH%\java\sqlj.zip` (required to build SQLJ programs)
- `%DB2PATH%\java\runtime.zip` (required to run SQLJ programs)

Java Applications

Start your application from the desktop or command line by running the Java interpreter on the executable program with this command:

```
java prog_name
```

where `prog_name` is the name of the program.

The DB2 JDBC driver handles the JDBC API calls from your application and uses the DB2 CAE to communicate the requests to the server and receive the results.

Note: An SQLJ application must be bound to the database before it is run.

Java Applets

Because Java applets are delivered over the web, a web server must be installed on your DB2 machine (server or client).

To run your applet, make sure your `.html` file is properly configured. Start the JDBC applet server on the TCP/IP port specified in the `.html` file. For example, if you specified:

```
param name=port value='6789'
```

then you would enter:

```
db2jstrt 6789
```

You must ensure that your working directory is accessible to your web browser. If it is not, copy your applet's `.class` and `.html` files into a directory that is accessible. For SQLJ applets, you must also copy the profile `.class` and `.ser` files as well.

Copy the `sql11ib/java/db2java.zip` file into the same directory as these other files. For SQLJ applets, also copy the `sql11ib/java/runtime.zip` file into this directory.

Then on your client machine start your web browser (which supports JDK 1.1) and load the `.html` file.

When your applet calls the JDBC API to connect to DB2, the JDBC driver establishes separate communications with the DB2 database through the JDBC applet server residing on the DB2 server.

Note: An SQLJ applet must be bound to the database before it is run.

Chapter 14. Entering DB2 Commands and SQL Statements

You can use the Command Center to enter DB2 commands and SQL statements if you are working with DB2 on OS/2 or Windows 32-bit operating systems. Otherwise, you can enter commands in the command line processor or at a system command prompt.

Task: Entering DB2 commands, SQL statements, and operating system commands	
Operating System	Tools
OS/2	<ul style="list-style-type: none">• Command Center• command line processor• command prompt
Windows 3.x	<ul style="list-style-type: none">• command line processor
Windows 32-bit operating systems	<ul style="list-style-type: none">• Command Center• command line processor• command window

Using the Command Center

On OS/2 and Windows 32-bit operating systems, the Command Center provides an interactive window that allows you to:

- Run SQL statements, DB2 commands, and operating system commands.
- See the execution result of SQL statements and DB2 commands in a results window. You can scroll through the results and save the output to a file.
- Save a sequence of SQL statements and DB2 commands to a script file. You can then schedule the script to run as a job. When a saved script is modified, all jobs dependent on the saved script inherit the new modified behavior.
- Recall and run a script file.
- See the execution plan and statistics associated with a SQL statement before execution. You do this by invoking Visual Explain in the interactive window.
- Get quick access to database administrative tools such as the Control Center and the Journal from the main tool bar.
- Display all the command scripts known to the system through the Script Center, with summary information listed for each.

To invoke the Command Center on OS/2:

- 1 Double-click on the **DB2 for OS/2** folder.
- 2 Double-click on the **Command Center** icon.

To invoke the Command Center on Windows 32-bit operating systems:

- 1 Click on **Start**, and select **Programs->DB2 for Windows-> Command Center**.

The Command Center contains a large input area in which you type your commands. To run the commands you have entered, select the **Start or stop execution** icon from the menu toolbar, or use CTRL+Enter.



It is no longer necessary to enter the commands with a db2 prefix; instead, enter the command as follows:

```
list node directory
```

To enter operating system commands in interactive mode, precede the operating-system command with an exclamation mark (!). For example:

```
!dir
```

If you want to enter multiple commands, you must end each command with the termination character, then press Enter to start the next command on a new line. (The default termination character is a semicolon(;).) To specify a different termination character, click on the **Tools Settings** icon in the menu toolbar.

After you enter a command, the Command Center displays the Results window, which informs you how the commands are proceeding.

To recall commands that you have entered, select the **Script** Tab, click on the drop down box, and select a command. To save commands as scripts, select **Script->Save as** from the menu bar.



If you want to store commonly used SQL statements or DB2 commands as scripts, click on the **Script Center** icon from the main tool bar.

Using the Command Line Processor

You can use the command line processor to enter SQL statements, DB2 commands, and operating system commands. It operates in the following modes:

Interactive Input Mode	You can enter operating systems commands, DB2 commands or SQL statements and view their output.
Command Line Mode	The DB2 command line processor behaves like a command window from your operating system. You can enter operating system commands, DB2 commands, or SQL statements and view their output.
File Input Mode	Refer to the <i>Command Reference</i> for information on the file input mode.

Interactive Input Mode

To invoke the command line processor in interactive input mode, do the following:

- On OS/2:
 - 1** Double-click on the **DB2 for OS/2** folder.
 - 2** Double-click on the **Command Line Processor** icon.
- On Windows 32-bit operating systems:
 - 1** Click on **Start** and select **Programs ->DB2 for Windows->Command Line Processor**.
- On Windows 3.x use one of the following methods:
 - From a DOS full screen, enter the **win db2clpw** command.
 - From Windows:
 - 1** Double-click on the **DB2** folder.
 - 2** Double-click on the **Command Line Processor** icon.

You can also invoke the command line processor in interactive input mode by entering the **db2cmd** command followed by the **db2** command at a command prompt. In interactive input mode, the prompt looks like this:

```
db2 =>
```

In interactive mode, you do not have to enter DB2 commands with a db2 prefix; instead, you just enter the DB2 command. For example:

```
db2 => list node directory
```

To enter operating system commands in interactive mode, precede the operating-system command with an exclamation mark (!). For example:

```
db2 => !dir
```



Windows 3.x does not support this method of running operating system commands.

If you need to enter a long command that does not fit on one line, use the line continuation character, \. For example:

```
db2 => select empno, lastname, birthdate, from \  
db2 (cont.) => employee where sex='F' order by empno desc
```

Note: You do not need to use a line continuation character when entering long commands in interactive input mode on Windows 3.x workstations.

To end interactive input mode, enter the **quit** command.

For more information on advanced topics using the command line processor, refer the *Command Reference*.

Command Line Mode

On OS/2, you can enter DB2 commands from a command prompt. You must include the db2 prefix. For example:

```
db2 list node directory
```

To invoke the command line processor in command line mode on Windows 32-bit operating systems, use one of the following methods:

- Click on **Start** and select **Programs-> DB2 for Windows->Command Window**.
- Enter the **db2cmd** command at a command prompt.

After you invoke the DB2 command environment, you can enter DB2 commands at the command prompt. You must include the db2 prefix.

Notes:

1. If the DB2 command contains characters that have special meaning on the operating system you are using, you will need to enter the command in quotation marks to ensure that it is run properly. For example, the following command will successfully retrieve all the information from the *employee* table, even if the * character has a special meaning on the operating system:

```
db2 "select * from employee"
```

2. The command line mode is not available on Windows 3.x systems.

For more information on using the command line processor, refer to the *Command Reference*.

Connecting to a Logical Node

The *default logical node* is the node that has the *logical_port* parameter set to 0. By default, a client connects to the default logical node unless the *DB2NODE* environment variable is set to another node number. For all commands and applications that connect to a database, you must ensure that the *DB2NODE* environment variable is set to reference the node number of the logical node that you want the operation to run on. You can find this node number by entering the **db2nlist** command. If the database partition server being connected to is not started on the machine, the application will return an error.

Use the **set client** command to target a specific node within a partitioned database system. For more information, refer to the *Command Reference*.



Use the **db2nlist** command to see the list of database partition servers that participate in an instance.

Remote applications always connect to the default node (logical port 0), unless the *DB2NODE* environment variable is set to another node number. Because all logical nodes on an instance share the same database manager configuration file, only one logical node per instance on a machine can service remote applications. The default node starts up the communication subsystem, if it was configured to do so. If the default node is shut down, the communication subsystem also ends. You will have to stop and restart a logical node on the machine, using the **db2start nodenum node_number** command to re-establish the communication subsystem.

Assume that you want to update the configuration for a database called TEST for a specific database partition server (for example, MACH1). Also assume that you created logical nodes 0 and 1 on MACH1. Because database configuration is defined at the database partition level, you would connect to each logical node on the machine and change the database configuration. Each database partition at a node has its own database configuration file for each database. You could do the following in two sessions on MACH1:

1 In session 1:

- a** Set the default connection node to 0 by entering the following command:

```
db2 set client connect_node 0 attach_node 0
db2 terminate
```

- b** Update the database configuration file for the remote server. For example, enter the following command:

```
db2 update database configuration for
database_alias using maxxappls 45
```

2 In session 2:

a Set the default connection node to *1* by entering the following command:

```
db2 set client
connect_node 1 attach_node 1
db2 terminate
```

b Update the database configuration file for the remote server by entering the following command:

```
db2 update database
configuration for database_alias using maxappls 65
```

If you want, you can run session 1 and 2, one after the other. For more information, refer to the *Command Reference*.



If you want to update the database configuration for *all* database partition servers in your partitioned database system, use the **db2_all** command. For example:

```
db2_all db2 update database configuration for
database_alias using maxappls 45
```

You can list the local database directory by entering the **list database directory on path** command. The output will show the catalog node number for the database and the current node number where you are connected.

Chapter 15. Controlling Your DB2 Environment

Registry values, environment variables, and configuration parameters control your database environment.



- To set registry values through the DB2 profile registry, see “Controlling the DB2 Profile Registry” on page 120.
 - To set DB2 environment variables, see:
 - “Setting Your Environment on OS/2” on page 121.
 - “Setting Your Environment on Windows 32-Bit Operating Systems” on page 122.
 - “Setting Your Environment on Windows 3.x” on page 123.
 - To set database manager configuration parameters on a client instance, see “Configuration Parameters” on page 130.
-

Prior to the introduction of the DB2 profile registry in Version 5, changing your DB2 environment on OS/2 or Windows 32-bit operating systems required you to change your system's environment variables.

With Version 5, almost all of the environment variables have been moved to the DB2 profile registry. Users with system administrative (SYSADM) authority for a given instance can update registry values for that instance. Use the **db2set** command to update DB2 registry values without rebooting your system. The DB2 registry applies the updated information to DB2 server instances and DB2 applications started after the changes are made.

Note: There is no DB2 profile registry on Windows 3.x. See “Setting Your Environment on Windows 3.x” on page 123 for more information.

DB2 configures its operating parameters by checking for variable values according to the following search order:

- The environment variable settings.
- Profile registry values set with the **db2set** command in the node-level profile.
- Profile registry values set with the **db2set** command in the instance-level profile.
- Profile registry values set with the **db2set** command in the global-level profile.

See “DB2 Registry Values and Environment Variables” on page 124 for descriptions of the subset of registry values and environment variables that you may want to adjust to get DB2 up and running. For more detailed information about all registry values and environment variables, refer to the *Administration Guide*.

Controlling the DB2 Profile Registry

The DB2 profile registry stores DB2 registry values. The "levels" of registry values are as follows:

DB2 node-level profile:

This profile is only used with DB2 Universal Database Extended Edition. It contains node-level variable settings and overrides. Values defined in this level will override their settings in the instance and global levels.

DB2 instance-level profile:

This profile contains instance level variable settings and overrides. Values defined in this level will override their settings in the global level.

DB2 global-level profile:

This profile contains machine global variable settings. Any variable not defined at the node or instance levels will be evaluated at this level. (In DB2 Extended Enterprise Edition, one global-level profile exists at each machine.)

To modify registry variable values, enter the **db2set** command.

The syntax of the **db2set** command is as follows:

- To set a parameter, enter:

```
db2set parameter=value
```

This command will set the parameter that you specify at the default level for that parameter.

- To set a parameter for a specific DB2 Extended Enterprise Edition node, enter:

```
db2set parameter=value -i instance_name node_number
```

- To set a parameter's value for a specific instance, enter:

```
db2set parameter=value -i instance_name
```

- To set a parameter at the global profile level, enter:

```
db2set parameter=value -g
```

Note: Some parameters will always default to the global level profile. They cannot be set at the instance or node level profiles; for example, *DB2SYSTEM* and *DB2INSTDEF*.

- To delete a parameter's value at a specified level, you can use the same command syntax to set the parameter, but specify nothing for the parameter value. For example, to delete the parameter's setting at the global level, enter:

```
db2set parameter= -g
```

- To delete a parameter's setting and to restrict its use, if it is defined at a higher profile level, enter:

```
db2set parameter= -null -i instance
```

This command will delete the setting for the parameter you specify and restrict high level profiles from setting this parameter (in this case, the DB2 global-level profile). However, the parameter you specify could still be set by a lower level profile (in this case, the DB2 node-level profile).

- To view the current session's parameter's value, enter:

```
db2set parameter
```

- To view the parameter's value at all levels, enter:

```
db2set parameter -all
```

- To view a list of all values defined in the profile registry, enter:

```
db2set -all
```

For more information on the **db2set** command and the administration of the DB2 profile registry, refer to the *Command Reference*.

Setting Your Environment on OS/2

It is strongly recommended that all DB2 specific registry values be defined in the DB2 profile registry. If DB2 variables are set outside of the registry, remote administration of those variables will not be possible, and the workstation will have to be rebooted in order for the variable values to take effect.

On OS/2, there are three system environment variables that are not stored in the DB2 profile registry: *DB2INSTANCE*, *DB2PATH*, and *DB2INSTPROF*. *DB2INSTANCE* and *DB2PATH* are set when DB2 is installed; *DB2INSTPROF* can be set after installation. The environment variable *DB2PATH* must be set; this environment variable is set during install and you should not modify it. Setting *DB2INSTANCE* and *DB2INSTPROF* is optional. Because the system environment variables are not set in the profile registry, you will need to reboot if you change their settings.

To determine the setting of an environment variable, enter:

```
set variable
```

To change the setting of an environment variable, enter the following command:

```
set variable=value
```

To modify system environment variables you must edit the `config.sys` file, then reboot to make the changes take effect.

Setting Your Environment on Windows 32-Bit Operating Systems

It is strongly recommended that all DB2 specific registry values be defined in the DB2 profile registry. If DB2 variables are set outside of the registry, remote administration of those variables will not be possible, and the workstation will have to be rebooted in order for the variable values to take effect.

Windows 95 has one system environment variable, *DB2INSTANCE*, that can only be set outside the profile registry; however, you are not required to set *DB2INSTANCE*. The DB2 profile registry variable *DB2INSTDEF* may be set in the global level profile to specify the instance name to use if *DB2INSTANCE* is not defined.

DB2 Extended Enterprise Edition servers on Windows NT have two system environment variables, *DB2INSTANCE* and *DB2NODE*, that must be set outside the profile registry. You are not required to set *DB2INSTANCE*. The DB2 profile registry variable *DB2INSTDEF* may be set in the global level profile to specify the instance name to use if *DB2INSTANCE* is not defined.

The *DB2NODE* environment variable is used to route requests to a target logical node within a machine. This environment variable must be set in the session in which the application or command is issued and not in the DB2 profile registry. If this variable is not set, the target logical node defaults to the logical node which is defined with port 0 on the machine. For more information, see "Connecting to a Logical Node" on page 117.

To determine the setting of an environment variable, enter:

```
set variable
```

To set system environment variables, do the following:

On Windows 95 and Windows 98:

Edit the autoexec.bat file and reboot to make the change take effect.

On Windows NT:

Set environment variables as follows:

- 1 Click on **Start** and select **Settings->Control Panel**.
- 2 Double-click on the **System** icon.
- 3 In the System Properties window, select the **Environment** tab and do the following:
 - a If the variable does not exist:
 - 1 Select any environment variable in the System Variables window.

- 2 Change the name in the **Variable** field to the name of the environment variable you want to set, for example *DB2INSTANCE*.
 - 3 Change the **Value** field to the value that you would like this parameter to take, for example, *db2inst*.
- b** If the variable already exists in the System Variables window, you can set a new value:
- 1 Select the environment variable you want to change or append, for example *DB2INSTANCE*.
 - 2 Change the **Value** field to the value that you would like this parameter to take, for example, *db2inst*.
- c** Click on the **Set** push button.
- d** Click on **OK**.
- 4 You may have to reboot your system for these changes to take effect.

Setting Your Environment on Windows 3.x

The DB2 environment on Windows 3.x is not controlled by profile registries. Instead, Windows 3.x clients define environment keywords in the *db2.ini* file (typically found in *C:\windows* directory).



On Windows 3.x, the parameters that control the DB2 environment are called environment keywords. However, because many Windows 3.x keywords are also used on operating systems that use the DB2 profile registries, environment keywords may also be referred to as registry values in this manual.

The *db2.ini* initialization file is an ASCII file that stores values for the Windows 3.x client environment keywords. Within this file, there is just one section header titled:

```
[DB2 Client Application Enabler]
```

The parameters are set by specifying a keyword with its associated keyword value in the form:

```
KEYWORD=keywordValue
```

The following is a sample *db2.ini* file:

```
[DB2 Client Application Enabler]
; This is for my Windows 3.x Client
DB2PATH=C:\SQLLIB\WIN
DB2INSTANCE=DB2
DB2INSTPROF=C:\SQLLIB
DB2TRACEON=N
```

Notes:

1. All the keywords and their associated values must be located below the section header.
2. The keywords are not case sensitive; however, their values can be if the values are character based.
3. Comment lines use a semicolon in the first position of a new line.
4. Blank lines are permitted. If duplicate entries for a keyword exist, the first entry is used (and no warning is given).

This file is located in the Windows product directory.

On Windows 3.x, the Client Application Enabler for DB2 Version 2.x, Version 4, and Version 5 must set this information *only* in the db2.ini file.

DB2 Registry Values and Environment Variables

The following subset of the DB2 registry values and environment variables are those that you may need to know about to get up and running. Each has a brief description; some may not apply to your environment.

You can view a list of all supported variables for your platform. On all operating systems except Windows 3.x, enter the following command:

```
db2set -lr
```

Note: Windows 3.x clients define environment variables in the file db2.ini. See “Setting Your Environment on Windows 3.x” on page 123 for details.

Parameter	Operating System	Values	Description
General			
DB2ACCOUNT	All	Default: (not set)	The accounting string that is sent to the remote host. Refer to the <i>DB2 Connect User's Guide</i> for details.

Parameter	Operating System	Values	Description
DB2BIDI	All	Default: (not set)	Enables bi-directional CCSID processing by UDB. Refer to the <i>DB2 Connect User's Guide</i> for details.
DB2CODEPAGE	All	Default: derived from the language ID, as specified by the operating system.	Specifies the code page of the data presented to DB2 for database client application. The user should not set <i>db2codepage</i> unless explicitly stated in DB2 documents, or asked to do so by DB2 service. Setting <i>DB2CODEPAGE</i> to a value not supported by the operating system can produce unexpected results. Normally, you do not need to set <i>DB2CODEPAGE</i> because DB2 automatically derives the code page information from the operating system.
DB2COUNTRY	All	Default: derived from the language ID, as specified by the operating system.	Specifies the country code of the client applications, which influences date and time formats.
DB2DBDFT	All	Default: (not set)	Specifies the database alias name of the database that will be implicitly connected to when applications are started and no implicit connect has been done. This keyword is ignored if it is not set.
DB2DBMSADDR	Windows 32-bit operating systems	Default= (not set), value for Windows 95 and Windows 98: (90000000), value for Windows NT: (20000000)	Specifies the default database manager shared memory address in hexadecimal format. If a shared memory address collision occurs, this value can be modified to force the database manager instance to allocate its shared memory at a different address. Use the <i>DB2DBMSADDR</i> registry profile variable to assign an address value in the range of x'20000000' to x'B0000000' on Windows NT or x'80000000' to x'B0000000' on Windows 95 using increments of x'40000000'. Use the db2set command to change the shared memory address. For example, to set <i>DB2DBMSADDR</i> to x'84000000', use the following command: db2set DB2DBMSADDR=84000000
DB2DISCOVERYTIME	OS/2 and Windows 32-bit operating systems	Default: (default time of 40 seconds is used when this parameter is not set)	Specifies the amount of time that a SEARCH discovery will search for DB2 systems.

Parameter	Operating System	Values	Description
DB2DMNBCKCTRL	Windows NT	Default: (not set)	If DB2 is installed on a backup domain controller, setting DB2DMNBCKCTRL=DOMAIN_NAME allows DB2 to use the security database on the backup domain controller, thereby reducing LAN traffic. (DOMAIN_NAME must be in upper case.) Note: A backup domain controller shadows the security database on the primary domain controller.
DB2INCLUDE	All	Default: (not set)	Specifies a path to be used during the processing of the SQL INCLUDE text-file statement during DB2 PREP processing. It provides a list of directories where the INCLUDE file might be found. Refer to the <i>Embedded SQL Programming Guide</i> for descriptions of how <i>db2include</i> is used in the different precompiled languages.
DB2INSTDEF	OS/2 and Windows 32-bit operating systems	Default=DB2 (on OS/2 and Windows 32-bit operating systems)	Sets the default instance to be used if <i>DB2INSTANCE</i> is not defined.
DB2INSTOWNER	Windows NT		The environment variable that is created in the DB2 profile registry when the instance is first created. This variable is set to the name of the instance-owning machine.
DB2OPTIONS	All	Default: (not set)	Sets command line processor options.
DB2SLOGON	Windows 3.x	Default (not set), Values: Y, N	Enables a secure logon in DB2 for Windows 3.x . If <i>db2slogon</i> =YES DB2 does not write user IDs and passwords to a file, but instead uses a segment of memory to maintain them. When <i>db2slogon</i> is enabled, the user must logon each time Windows 3.x is started.
DB2TIMEOUT	Windows 3.x and Macintosh	Disabled	Used to control the timeout period for Windows 3.x and Macintosh clients during long SQL queries. After the timeout period has expired a dialog box pops up asking if the query should be interrupted or allowed to continue. The minimum value for this variable is 30 seconds. If <i>db2timeout</i> is set to a value between 1 and 30, the default minimum value will be used. If <i>db2timeout</i> is set to a value of 0, or a negative value, the timeout feature is disabled. This feature is disabled by default.

Parameter	Operating System	Values	Description
DB2TRACENAME	Windows 3.x and Macintosh	Default (not set)	On Windows 3.x and Macintosh, specifies the name of the file where trace information is stored. The default on Windows 3.x is <i>db2tracename=</i> DB2WIN.TRC, and is saved in your current instance directory (for example, \sql11ib\db2). The default on Macintosh is <i>db2tracename=</i> DB2MAC.TRC. We strongly recommend that you specify the full path name when naming the trace file.
DB2TRACEON	Windows 3.x and Macintosh	Default (not set), Values: Y, N	On Windows 3.x and Macintosh, turn trace on to provide information to IBM in case of a problem. (It is not recommended that you turn trace on unless you encounter a problem you cannot resolve.) Refer to the <i>Troubleshooting Guide</i> for information on using the trace facility with DB2 Client Application Enabler.
DB2TRCFLUSH	Windows 3.x and Macintosh	Default (not set), Values: Y, N	On Windows 3.x and Macintosh, <i>db2trcf flush</i> can be used in conjunction with <i>db2traceon=Y</i> . <i>db2trcf flush=Y</i> will cause each trace record to be written immediately into the trace file. Setting this will slow down your DB2 system considerably, so the default setting is <i>db2trcf flush=N</i> . This setting is useful in cases where an application hangs the system and therefore needs to be rebooted. Setting this keyword guarantees that the trace file and trace entries are not lost by the reboot.
DB2TRCSYSERR	Windows 3.x and Macintosh	Default: (not set), Values: Y, N	Specifies the number of system errors to trace before the client turns off tracing. The default value traces one system error, after which, trace is turned off.

Parameter	Operating System	Values	Description
DB2YIELD	Windows 3.x	Default (not set), Values: Y, N	Specifies the behavior of the Windows 3.x client while communicating with a remote server. When N is set, the client will not yield the CPU to other Windows 3.x applications, and the Windows environment is halted while the client application is communicating with the remote server. You must wait for the communications operation to complete before you can resume any other tasks. When set to Y, your system functions as normal. It is recommended that you try to run your application with <i>db2yield=YES</i> . If your system crashes, you will need to set <i>db2yield=NO</i> . For application development, ensure your application is written to accept and handle Windows messages while waiting for a communications operation to complete.
System Environment			
DB2INSTANCE	All	Default= <i>DB2INSTDEF</i> on OS/2 and Windows 32-bit operating systems	The environment variable used to specify the instance that is active by default. On UNIX, users must set the environment variable <i>DB2INSTANCE</i> .
DB2INSTPROF	OS/2, Windows 3.x, and Windows 32-bit operating systems	Default (not set)	The environment variable used to specify the location of the instance directory on OS/2 and Windows 32-bit operating systems if different than <i>DB2PATH</i> .
DB2PATH	OS/2, Windows 3.x, and Windows 32-bit operating systems		The environment variable used to specify the directory where the product is installed on OS/2 and Windows 32-bit operating systems. By default it is set to <i>x:\sql11ib\win</i> on Windows 32-bit operating systems; where <i>x:</i> is the install drive.
DB2NODE	All		Used to specify the target logical node of a DB2 Extended Enterprise Edition database partition server that you want to connect to. If this variable is not set, the target logical node defaults to the logical node which is defined with port 0 on the machine. This applies only to local clients.

Parameter	Operating System	Values	Description
Communications			
DB2COMM	All, server only	Default (not set), values: any combination of APPC, IPXSPX, NETBIOS, NPIPE, TCPIP	Specifies the communication managers that are started when the database manager is started. If this is not set, no DB2 communications managers are started at the server.
DB2NBADAPTERS	OS/2 and Windows 32-bit operating systems	Default=0, Range: 0-15, Multiple values should be separated by commas	Used on the server to specify which local adapters to use for DB2 NetBIOS LAN communications. Each local adapter is specified using its logical adapter number. You can set <i>DB2NBADAPTERS</i> on the client if you want to perform a NetBIOS discovery on an adapter other than the default adapter 0. Note: On Windows 95 and Windows 98 <i>DB2NBADAPTERS</i> is only used for discovery.
DB2NETREQ	Windows 3.x	Default=3, Range: 0-25	Specifies the number of NetBIOS requests that can be run concurrently on Windows 3.x clients. The default value is 3. The number can range from 0 to 25. The higher you set this value, the more memory below the 1MB level will be used. When the concurrent number of requests to use NetBIOS services reaches the number you have set, subsequent incoming requests for NetBIOS services are held in a queue and become active as the current requests complete. If you enter 0 (zero) for <i>db2netreq</i> , the Windows database client issues NetBIOS calls in synchronous mode using the NetBIOS wait option. In this mode, the database client allows only the current NetBIOS request to be active and does not process another one until the current request has completed. This can affect other application programs. The 0 value is provided for backwards compatibility only. It is strongly recommended that 0 not be used.
DB2PORTRANGE	Windows NT		Sets the TCP/IP port range used by the fast communication manager (FCM). This ensures that any additional database partition servers created on other machines will have the same port range.

Configuration Parameters

Configuration parameters are values that affect the operating characteristics of a database or database management system.

Database manager configuration parameters exist on servers and clients; however, only certain database manager configuration parameters can be set on the client. These parameters are a subset of the database management configuration parameters that can be set on the server. In DB2 Extended Enterprise Edition, one database manager configuration file is shared between all database partition servers in the instance.

Database configuration parameters can be set only on the server, or remotely from the Control Center. In DB2 Extended Enterprise Edition, each database partition has its own configuration file.

For information on setting database or database manager configuration parameters, refer to the *Administration Guide*.

Setting Database Manager Configuration Parameters on a Client Instance

To control database management configuration parameters on a client instance on OS/2 or Windows 32-bit operating systems, use the Client Configuration Assistant (CCA).

- On OS/2:
 - 1 Double-click on the **Client Configuration Assistant** icon in the **DB2 for OS/2** folder.
 - 2 Click on the **Client Settings** push button.
- On Windows 32-bit operating systems:
 - 1 Click on **Start** and select **Programs->DB2 for Windows->Client Configuration Assistant**. The Client Configuration window opens.
 - 2 Click on the **Client Settings** push button.

To modify a configuration parameter:

- 1 Select the tab that corresponds to the type of configuration parameter that you want to modify.
- 2 Select the configuration parameter that you want to modify from the **Parameter** window.

| **3** Select the value for this parameter from the **Value** window.

| **4** Click on **OK**.

For more information, refer to the online help.



- For information on using Client Configuration Assistant, see Chapter 18, “Using the Client Configuration Assistant” on page 159.
-

Chapter 16. Using the Windows NT Performance Monitor

There are two performance monitors available to DB2 for Windows NT users:

DB2 Performance Monitor

The DB2 Performance Monitor provides snapshot and event data related to DB2 only. (For more information, click on the **Help** push button in the Control Center and see the Getting Started online help.)

Windows NT Performance Monitor

The Windows NT Performance Monitor enables you to monitor both database and system performance, retrieving information from any of the performance data providers registered with the system. Windows NT also provides performance data on all aspects of machine operation, including:

- CPU usage
- Memory utilization
- Disk activity
- Network activity



To set up and use the Windows NT Performance Monitor with DB2, refer to:

- “Registering DB2 with the Windows NT Performance Monitor”
 - “Enabling Remote Access to DB2 Performance Information” on page 134
 - “Displaying DB2 Performance Values” on page 135
 - “Accessing Remote DB2 Performance Information” on page 136
 - “Resetting DB2 Performance Values” on page 136.
-

Registering DB2 with the Windows NT Performance Monitor



The setup program automatically registers DB2 with the Windows NT Performance Monitor for you.

To make DB2 performance information accessible to the Windows NT Performance Monitor, you must register the DLL for the DB2 for Windows NT Performance Counters. This also enables any other Windows NT application using the Win32 performance APIs to get DB2 performance data.

To install and register the DB2 for Windows NT Performance Counters DLL (DB2Perf.DLL) with the Windows NT Performance Monitor, enter the following command:

```
db2perfi /i
```

This copies the DLL to the x:\winnt\system32 directory, where x: is the drive where DB2 is installed.

Registering the DLL also creates a new key in the services section of the registry. One entry gives the name of the DLL, which provides the counter support. Three other entries give names of functions provided within that DLL. These functions are:

- Open** Called when the DLL is first loaded by the system in a process.
- Collect** Called to request performance information from the DLL.
- Close** Called when the DLL is unloaded.

Enabling Remote Access to DB2 Performance Information

To see Windows NT performance objects from another DB2 for Windows NT machine, you must register a user account, that belongs to the local Administration group, with DB2. (The default Windows NT Performance Monitor username, *SYSTEM*, is a DB2 reserved word and cannot be used.) To register the name, enter the following command:

```
db2perfr -r username password
```

where *username* is the Administrator's username.

Note: The username you choose must conform to the naming rules in Appendix D, "Naming Rules" on page 233.

The user account data is held in a key in the registry, with security that allows access only by administrators and the *SYSTEM* account. The data is encoded, to prevent security concerns about storing an administrator password in the registry.

Once a user account has been registered with DB2, even local instances of the Performance Monitor will explicitly log on using that user account. This means that if the account information registered with DB2 does not match, local sessions of the Performance Monitor will not show DB2 performance information.

Deregistering the Username for the DB2 Performance Monitor

A user account must be maintained to match the username and password values stored in the Windows NT Security database. If the username or password is changed in the Windows NT Security database, the username and password combination used for remote performance monitoring must be reset.

To deregister, enter the following command:

```
db2perfr -u
```

Displaying DB2 Performance Values

To display DB2 performance values using the Performance Monitor, do the following:

- 1 Start the Performance Monitor by clicking on **Start** and selecting **Programs->Administrative Tools (Common)->Performance Monitor**.
- 2 Select **Edit->Add to Chart** from the Action bar, and choose the performance values you want displayed from the **Add to Chart** box. This box displays a list of performance objects providing performance data. Select an object to see a list of the counters it supplies.

A performance object can also have multiple instances. For example, the LogicalDisk Object provides counters such as 'Local Active Databases' and '% User Time'; it also has an instance for each logical drive in the machine, including 'C:' and 'D:'.

For more information on using the Performance Monitor, refer to your Windows NT documentation.

Windows NT provides the following performance objects:

Windows NT Database Manager

This object provides general server information for a single Windows NT server instance. The DB2 instance being monitored appears as the object instance. This object is only selectable if the database manager has been started. For example, after the `db2start` command has been issued.

You can only get performance information from one database partition server at a time. The database partition server that the Performance Monitor monitors is determined by the setting of the `DB2INSTANCE` and `DB2NODE` environment variables. If you have multiple logical nodes running and want to see performance information for each node,

you must start a separate session of the Performance Monitor with the *DB2INSTANCE* and *DB2NODE* environment variables set for each node.

Windows NT Databases

This object provides information for a particular database. Information is available for each currently active database. This object is only selectable if the database is active. For example, enter the `db2 activate database<database_alias>` command.

Windows NT Applications

This object provides information for a particular DB2 application. Information is available for each currently active DB2 application. This object is only selectable if an application is currently running. For example, enter the `db2connect to <database_alias>` command.

Accessing Remote DB2 Performance Information

Enable remote access to DB2 (see “Enabling Remote Access to DB2 Performance Information” on page 134) if you want the Performance Monitor to remotely access performance information on another machine. In the **Add to Chart** box, select another computer to monitor. This brings up a list of all the available performance counters on that computer.

Resetting DB2 Performance Values

When an application calls the DB2 monitor APIs, the information returned is normally the cumulative values since the DB2 server was started. However, often it is useful to:

1. Reset performance values
2. Run a test
3. Reset the values again
4. Re-run the test.

To reset a database performance value, enter the **db2perf** command.

By default, this resets performance values for all active databases. However, you can also specify a list of databases to reset. The command resets the values for *all* programs currently accessing database performance information for the relevant DB2 server instance. For more information, refer to the *Command Reference*.



The **db2perf** command only resets counters on a local machine. In the case where you have multiple logical nodes, the **db2perf** command will only need to be run once on that machine. However, to reset counters on *all* nodes, you must use the **rah** command. For example:

```
rah db2perf
```

For more information about the **rah** command, refer to the *Administration Guide*.

| The **db2perf** command also resets the values seen by anyone remotely accessing
| DB2 performance information when it is executed.

Part 6. Advanced Installation and Configuration

Chapter 17. Migrating from Previous Versions and Releases

This section discusses how to migrate from previous versions of DB2 to Version 5.2 and what you should do if you upgrade from Release 5.0 to Release 5.2 of DB2.

Migrating from Previous Versions of DB2

This section describes the steps required to migrate either a DB2 Version 2.x or a Database Server Version 4.x single-partition database system to a DB2 Universal Database Extended Enterprise Edition database system. It also includes information on how to update a Version 5 single-partition database system to a Version 5 multi-partition database system.

Note: The migration process for Database Server Version 4 is identical to that used for DB2 Version 2. Whenever DB2 Version 2 is mentioned in this section, the same information also applies to Database Server Version 4.

When you migrate from a Version 2.x database system to a DB2 Extended Enterprise Edition Version 5 database system, depending on the installation option you choose, the setup program will migrate all instances to the Version 5 multi-partition format or single-partition format. Refer to the *Administration Guide* for descriptions of how to add database partition servers to a system, and how to redistribute data across database partitions.

After you install DB2 Extended Enterprise Edition, any databases that you created with earlier versions of DB2 will not automatically work with the new version of DB2. You must follow the steps in this section to migrate your instances and databases.

After you migrate a DB2 Version 2.x database to DB2 Extended Enterprise Edition, the database is relocated from the `drive:\instance_name` directly to the `drive:\instance_name\NODE0000`. A node is created with a default name of NODE0000.

This section contains the following information:

- “Pre-Installation Steps” on page 142.
- “Installing DB2 Extended Enterprise Edition Version 5” on page 147.
- “Post-Installation Steps” on page 150.



To avoid difficulties during database migration, read this entire section before starting the migration process. It is strongly recommended that you follow all the steps listed in this section.

Upgrading from DB2 Release 5.0 to Release 5.2

If you are upgrading from Release 5.0 to Release 5.2 of DB2, we recommend that you run the **db2upd52** command to support the new functionality added in the SYSIBM.SYSDATATYPES catalog table. See “Updating Catalog Table Data” on page 156 for more information.

You should run the **db2upd52** command to update the databases after you run the **db2iupdt** command to update the instance. See “Updating Catalog Table Data” on page 156 for further information.

Pre-Installation Steps

This procedure helps you to ensure that all databases on your system can be migrated to Version 5. Before you install DB2 Extended Enterprise Edition, perform the following steps.

Step 1. Back Up Databases

You should back up all databases before installing Version 5 of DB2. To back up the databases, perform the following steps:

- 1 Complete all database transactions.
- 2 Ensure all applications disconnect from each database.

To list all applications that are connected to the database, enter the **db2 list applications** command. If all applications are disconnected, this command will return the following message:

```
SQL1611W No data was returned by the Database System Monitor
```

- 3 Ensure all databases are cataloged.



The **db2ckmig** database migration verification command does not verify uncataloged databases. See “Step 2. Verify that Databases Can Be Migrated” on page 143 for a description of how **db2ckmig** works.

To view a list of all the cataloged databases in the instance, enter the following command:

```
db2 list database directory
```

- 4 Make a backup copy of all databases. Refer to your current copy of the *Administration Guide* for information on making backup copies of databases and to the *Command Reference* for the syntax of the backup command.

Note: Make sure that this is the most recent backup copy of the database before you start the next procedure.

- 5 Stop the database manager by entering the **db2stop** command.

Step 2. Verify that Databases Can Be Migrated

DB2 Extended Enterprise Edition Version 5 provides the **db2ckmig** command to check that databases can be migrated, prior to installation. The command is located on the product CD-ROM.

To run this command:

- 1 Insert the CD-ROM into the CD-ROM drive.

Note: If your operating system is Windows NT 4.0 or Windows 95, the **setup** program is started automatically via the operating system's **autorun** feature. Do not proceed with the install. Instead, cancel and proceed to the next step.

- 2 Enter the **db2ckmig** command (located in the `x:\DB2\COMMON` directory, where `x:` represents your CD-ROM drive) to verify that the databases on your system can be correctly migrated. The syntax of the command is as follows:

```
db2ckmig database /l filename
           /e
           /u userid /p password
```

database Specifies an alias name of a database to be verified.

/e: Specifies that all cataloged databases are to be verified.

/l: filename Specifies a target file to keep a list of errors and warnings generated for the scanned database. A path may be specified, but it must exist. This is a required parameter.

/u userid Is an optional parameter that specifies the user account used to connect to the database.

/p password Is an optional parameter that specifies the password of the user account used to connect to the database.



You can enter the **db2ckmig** command on remote systems; the database parameter must specify the alias name of the remote database. The file will be written on your local system.

For example, to check that all databases cataloged on your system can be migrated and to log all the messages from this command to the `c:\temp\message.txt` file, enter the following command:

```
x:\DB2\COMMON\db2ckmig /e /1 c:\temp\message.txt
```

- 3 The command generates a log file and places it in the location specified by the `/1` option. If there are errors, see Table 6 for suggested corrective actions. Once the errors have been corrected, re-enter the **db2ckmig** command to ensure that the databases are ready to be migrated.

Error	Action
A database is in backup pending state	Perform a backup of the database.
A database is in roll-forward pending state	Recover the database as required; perform or resume a roll-forward database to end of logs and stop.
Table space ID is not in normal state	Recover the database and table space as required; perform or resume a roll-forward database to end of logs and stop.
A database is in an inconsistent state	Restart the database to return it to a consistent state.

Table 6 (Page 2 of 3). Correcting Error Messages

Error	Action
<p>The database contains database objects that have a schema name of SYSCAT, SYSSTAT, or SYSFUN</p>	<p>These schema names are reserved for the Version 5 database manager. To correct this error, do the following:</p> <ol style="list-style-type: none"> 1 Back up the database. 2 Export the data from the database object (catalogs or tables). 3 Drop the object. 4 Recreate the object with the corrected schema name. 5 Import/Load the data into the object. 6 Run the db2ckmig command against the database again, ensuring that the database passes the db2ckmig check. 7 Make a backup copy of the database.
<p>The database contains database objects that have a dependency on the SYSFUN.DIFFERENCE function. Possible violated database objects are:</p> <ul style="list-style-type: none"> • Constraint • Function • Trigger • View 	<p>The SYSFUN.DIFFERENCE function must be dropped and recreated during database migration. However, if there is a database object that is dependent on this function, migration will fail. To correct this error, do the following:</p> <p>Constraint Enter the alter table command to drop the constraint.</p> <p>Function Enter the drop function command to drop the function dependent on SYSFUN.DIFFERENCE.</p> <p>Trigger Enter the drop trigger command to drop the trigger.</p> <p>View Enter the drop view command to drop the view.</p> <p>Note: Any package dependent on the SYSFUN.DIFFERENCE function will be marked inoperative after migration. Therefore, the db2ckmig command will not report any package that is dependent on the SYSFUN.DIFFERENCE function.</p>

Table 6 (Page 3 of 3). Correcting Error Messages

Error	Action
<p>The database contains user-defined distinct types that use the type name of DATALINK or REFERENCE.</p>	<p>The data type names, DATALINK and REFERENCE are reserved for the Version 5 database manager. To correct this error, do the following:</p> <ol style="list-style-type: none"> 1 Back up the database. 2 Export the data from any tables that are dependent on the data types. 3 Drop any tables dependent on the data types, and then drop the data types. These drops may drop other objects such as views, indexes, triggers, or functions. 4 Create data types with different type names and recreate the tables using the new data type names. Recreate any dropped views, indexes, triggers, or functions. 5 Import/Load the data into the object. 6 Run the db2ckmig command against the database again, ensuring that the database passes the db2ckmig check. 7 Make a backup copy of the database.

Migration Considerations for the User Exit Program



Follow these instructions if you are using the **db2uexit** user exit program with previous versions of DB2.

DB2 Version 5 has changed the interface it uses to invoke the user exit program to archive and retrieve log files. For more information on these new interfaces, refer to the *Administration Guide*. The name for the user exit program has changed to **db2uex t2** in Version 5. (In previous versions, it was called **db2uexit**.)

Follow these instructions if you are using the **db2uexit** user exit program with previous versions of DB2.

The following should be considered before installing DB2 Extended Enterprise Edition Version 5 if you are using the **db2uexit** user exit program. The installation of Version 5 automatically migrates Version 2.x instances.

- If the Version 2.x user exit program, **db2uexit.exe** is found in the `\sqllib\bin` directory before installation, it will remain in this directory after the installation completes. The **db2uext2.exe** program will also be installed in this directory. Its function is to invoke the **db2uexit.cmd** or **db2uexit.exe** user exit programs using the Version 2.x interface. This allows the old user exit program to be used on Version 5.x.
- If **db2uexit.exe** is in a directory other than the `sqllib\bin` directory, it will remain there after installation, but **db2uext2.exe** will not be installed in the `sqllib\bin` directory. Following installation, if you want to use the old user exit, you will have to copy it to the `sqllib\bin` directory, then copy **db2uext2.v2** from the `sqllib\misc` directory to the `sqllib\bin` directory, and rename it to **db2uext2.exe**.

At a convenient time, you should modify your user exit program to use the new DB2 Version 5 interfaces. The new user exit program should replace **db2uext2** in the `sqllib\bin` directory, used to support the pre-Version 5 user exit program, **db2uexit**, which should be removed.

At this point, you are now ready to install DB2 Extended Enterprise Edition. Before starting, make sure that there are no applications running against the databases and that you stop all databases as described in “Step 1. Back Up Databases” on page 142.

Installing DB2 Extended Enterprise Edition Version 5

After you have successfully completed the pre-installation steps, you can now start installing DB2 Extended Enterprise Edition Version 5 using either the interactive or unattended method. During the installation of DB2 Extended Enterprise Edition Version 5, instance migration for instances created in previous versions of DB2 occurs. For installation instructions, see Chapter 4, “Installing and Configuring DB2 Extended Enterprise Edition” on page 43.

After DB2 is installed, all instances are migrated to the latest Version 5 single-partition format. If you chose to use an existing default instance as the default instance for the partitioned database system that you are creating, that instance was updated to the Version 5 multipartition format.

After installation, you must migrate all Version 5 databases for use in your new partitioned database system. If you want to use other existing instances, you must update them to the multipartition format.

The following table shows the steps that you must do after installing DB2 Extended Enterprise Edition.

Previously-Installed Version of DB2	Version of DB2 after Installation	Post Installation Steps
Version 2.x	<p>DB2 Extended Enterprise Edition Version 5</p> <p>On the Select the Default Instance window:</p> <ul style="list-style-type: none"> • If you select the default instance (that is, the instance specified by the <i>DB2INSTANCE</i> environment variable), it is migrated to the multipartition format. • If you do not select to migrate to the default instance, a default instance called DB2MPP is created. <p>Notes:</p> <ol style="list-style-type: none"> 1. Regardless of the option you select, all other instances are migrated to the latest single-partition format, but must be updated if you want to use them in a partitioned database system. 2. If you selected to add a node, the migration process will not create an Administration Server. 	<ol style="list-style-type: none"> 1. Migrate your databases. See “Migrating Databases” on page 151 for details. 2. Update all instances that you want to use in a partitioned database system. See “Updating Instances for use in Partitioned Database Systems” on page 151 for more information.

Previously-Installed Version of DB2	Version of DB2 after Installation	Post Installation Steps
Version 2.x	<p>DB2 Extended Enterprise Edition Version 5</p> <p>If you select the This machine will be a single-partition database server radio button on the Select an Installation Option for an Existing DB2 Install window in the setup program, all instances are migrated to the single-partition format.</p>	<ol style="list-style-type: none"> 1. Migrate your databases. See “Migrating Databases” on page 151 for more information. 2. Update all instances that you want to use in a partitioned database system. See “Updating Instances for use in Partitioned Database Systems” on page 151 for details.
DB2 Universal Database Version 5	<p>DB2 Extended Enterprise Edition Version 5</p> <p>On the Select the Default Instance window:</p> <ul style="list-style-type: none"> • If you select the default instance (that is, the instance specified by the <i>DB2INSTANCE</i> environment variable), it is migrated to the multipartition format. • If you do not select the default instance, a default instance called DB2MPP is created. 	<p>Update all instances that you want to use in a partitioned database system.</p> <p>See “Updating Instances for use in Partitioned Database Systems” on page 151 for more information.</p>

Previously-Installed Version of DB2	Version of DB2 after Installation	Post Installation Steps
DB2 Universal Database Version 5	<p>DB2 Extended Enterprise Edition Version 5</p> <p>If you select the This machine will be a single-partition database server radio button on the Select an Installation Option for an Existing DB2 Install window in the setup program, all instances are migrated to a single-partition database system.</p> <p>Note: Regardless of the option you select, all other instances are migrated to the Version 5 single-partition format. To use these instances in a partitioned databases system, they must be updated to the multipartition format.</p>	<p>Update all instances that you want to use as partitioned database systems.</p> <p>See "Updating Instances for use in Partitioned Database Systems" on page 151 for details.</p>

Post-Installation Steps

After installing the DB2 products, you can now migrate databases and complete other migration activities.



Do not scale the database system before you migrate databases; otherwise, the database migration will fail.

Migrating Databases

If you are updating a single-partition Version 5 instance to the Version 5 multipartition format, you do not need to migrate the databases in that instance. Databases that were created in any versions previous to Version 5 must be migrated. When these databases are migrated for use in a partitioned database system, default nodegroups are created for them. For more information about default nodegroups and how they are used, see “Nodegroups and Data Partitioning” on page 23.

To migrate databases owned by an instance, you need to perform the following steps:

- 1 Log in with a user account that has SYSADM authority.
- 2 Ensure that the databases you want to migrate are cataloged. To retrieve a list of all cataloged databases on your system, enter the `db2 list database` command.
- 3 Migrate the database. Refer to the *Version 5 Command Reference* for the database migration syntax command.

Updating Instances for use in Partitioned Database Systems

After migrating databases, you are then ready to update any instance that you want to use in a partitioned database system. To do this, use the `db2iupdt` command. The syntax of the `db2iupdt` command is as follows:

```
db2iupdt instance_name /u:username,password
        /p:instance_profile_path /r:baseport,endport
        /h:hostname
```

where:

instance_name

Specifies the name of the instance to be updated.

/u:domain_name\username,password

Specifies the domain, account name, and password for the DB2 service.

/p:instance_profile_path

This parameter is optional. If you do not specify the instance profile path, the instance directory is created under the `\sqllib` directory, and given the shared name, `DB2-instance_name`. Read and write permissions will automatically be granted to everyone in the domain. You can change the permissions to restrict access to that directory.

To specify a different instance profile path, you must create a shared object. For example: /p:\machineA\DB2-DB2MPP.

/r:baseport,endport

Specifies the range of TCP/IP ports to be used for communications among database partition servers. When this option is specified, the services file of the local machine is updated with the following entries:

```
DB2_instance_name      baseport/tcp
DB2_instance_name_END endport/tcp
```

If you do not specify a valid port range, DB2 will select one for you.

/h:hostname

Overrides the default TCP/IP hostname if there is more than one TCP/IP hostname for the current machine.

Post Database Migration Actions

There are optional activities you may want to undertake following database migration.



You can also apply these optional activities to a down-level database backup which is restored to Version 5, because at the end of the restore, the database is migrated to Version 5.

- **Unique index conversion to DB2 Universal Database Version 5 semantics**

Version 5 of DB2 supports deferred unique constraint checking until the end of a statement. This can result in correct processing of multiple row updates, that in previous releases of DB2, returned an error because the updates temporarily created duplicate values in the transient state. Deferred unique constraint checking will guarantee that updates which result in a table with only unique keys (for example, key = key + 1) will succeed, regardless of the order of the data.

Note: This change only applies for unique indexes that are created in Version 5 of DB2.

All unique indexes in a migrated database do not automatically migrate to Version 5 semantics during database migration because :

- Converting unique indexes is a very time-consuming operation.
- You may have applications that depend on the previous version's unique index semantics.
- You may want to manage the staged conversion of unique indexes on your own schedule, when needed.



All existing applications will continue to work even if the unique indexes are not converted to Version 5 semantics. You have to convert unique indexes to Version 5 semantics only if support for deferred uniqueness checking is required.

To convert unique indexes, you need to perform the following steps:

- 1 Log in with a user account that has SYSADM authority.
- 2 Start the database manager by entering the **db2start** command.
- 3 Run the **db2uiddl** command against your migrated database. Refer to the *Version 5 Command Reference* for the syntax of this command.

The **db2uiddl** command searches the database catalog tables and generates all the CREATE UNIQUE INDEX statements for user tables in an output file.

- 4 Review the output generated from the **db2uiddl** command, and make changes, if needed. Comments in the output will flag any situations that require your attention.
- 5 Connect to the database by entering the **db2 connect to database_alias** command.
- 6 Execute the output file, generated by the **db2uiddl** command, as a DB2 Command Line Processor command file, using a command similar to the following:

```
db2 -tvf filename
```

where `filename` is the name of the file generated by the **db2uiddl** command.

- 7 Disconnect from the database by entering the **db2 connect reset** command.

DB2 interprets the re-creation of an existing unique index to signal that the index is ready to be converted to Version 5 semantics.

- **Update Statistics**

When database migration is completed, the old statistics that are used to optimize query performance are retained in the catalogs. However, Version 5 of DB2 has statistics that are modified or do not exist in the previous version. To take advantage of these, you may want to enter the **runstats** command on tables, particularly those tables that are critical to the performance of your SQL queries.

Refer to the *Version 5 Command Reference* for the syntax of the **runstats** command. For details on the statistics, refer to the *Version 5 Administration Guide*.

- **Rebind Packages**

During database migration, all existing packages are invalidated. After the migration process, each package is rebuilt when it is used for the first time by the Version 5 database manager. However, for better performance, we recommend that you run the **db2rbind** command to rebuild all packages stored in the database, after database migration is complete. Refer to the *Version 5 Command Reference* for the syntax of this command.

- **Update database and database manager configuration**

Some of the database configuration parameters are changed to Version 5 defaults or to other values during database migration. The same is true for database manager configuration parameters which may have changed, during instance

migration, to Version 5 defaults or to other values. Refer to the *Administration Guide* for more information about configuration parameters.

The following database manager configuration parameters are changed to Version 5 defaults:

- *Sort heap threshold (sheapthres)*

If the migrating database configuration file has this parameter at a value which is less than the Version 5 default, the parameter is reset to its Version 5 default value.

- *Backup buffer (backbufsz)*

If the migrating database configuration file has this parameter at a value which is less than the Version 5 minimum, the parameter is reset to its Version 5 minimum value.

- *Restore buffer (restbufsz)*

If the migrating database configuration file has this parameter at a value which is less than the Version 5 minimum, the parameter is reset to its Version 5 minimum value.

- *Maximum number of concurrent active databases (numdb)*

If the migrating database configuration file has this parameter at a value which is greater than the Version 5 maximum, the parameter is reset to its Version 5 maximum value.

- *TM-database (tm_database)*

If the migrating database configuration file has this parameter set to NULL, the parameter is reset to 1ST_CONNECT.

- *Query heap size (query_heap_sz)*

If the migrating database configuration file has this parameter at a value less than the Application Support Layer heap size (aslheapsz) of the same file, the parameter is reset to aslheapsz + 1.

- *Agent pool size (num_poolagents)*

If the migrating database configuration file has the value of this parameter greater than the value of the Maximum Number of agents (maxagents) of the same file, the parameter is reset to the value of maxagents - 1.

The following database configuration parameters are changed to Version 5 defaults:

- *Application Control heap size (app_ctl_heap_sz)*

If the migrating database configuration file has this parameter at a value which is less than the Version 5 default, the parameter is reset to its Version 5 default value.

- *Lock List (locklist)*

For Version 1 DB2 databases, the lock list parameter will be first adjusted to $\text{locklist} * 32 / 25$. If the migrating database configuration file has this parameter at a value less than the Version 5 default value, the parameter is reset to its Version 5 default value.

- *Database Heap (dbheap)*

For Version 1 DB2 databases, the database heap size will first be adjusted to $\text{dbheap} * 16$. If the migrating database configuration file has this parameter at a value that is less than the Version 5 default, the parameter is reset to its Version 5 default value.

- *Default log space (logfilsiz)*

The database migration process will attempt to increase the *logfilsiz* value, if the log file related parameters have a total *logfilsiz* that is less than the default *logfilsiz* value.

- *Application Heap size (applheapsz)*

For Version 1 DB2 databases, application heap size will be first adjusted to $\text{applheapsz} * 16$. If the migrating database configuration file has this parameter at a value which is less than the Version 5 default, the parameter is reset to its Version 5 default value.

- *Sort heap (sortheap)*

For Version 1 DB2 databases, the value of the sort heap parameter will be adjusted to $\text{sortheap} * 16$.

For better performance, you may want to tune your database and database manager configuration parameters to take advantage of Version 5 enhancements. Refer to the *Command Reference* for the syntax of updating database and database manager configuration.

- **Migrate Explain Tables**



If you are not using explain tables in the version of DB2 that you are migrating, skip this task.

Version 5 of DB2 has added several new columns to the explain tables. These columns provide for the capture of:

- Data for new SQL features added in Version 5
- More detailed access plan information.

While the explain function in Version 5 will continue to work with explain tables created for Version 2, the new Version 5 data will not be captured in them.

For better performance of SQL statements, we recommend that the Version 2 explain tables be dropped and new explain tables be created; refer to the *SQL Reference* and the *Administration Guide* for details on creating new explain tables. If, however, there are Version 2 explain tables that you need for ongoing comparison, you can use the EXPLMIG.DDL script to migrate them.

To migrate the explain tables in a database that has been migrated to Version 5, connect to the database and run the following command from the `x:\sql1lib\misc\` directory:

```
db2 -tf EXPLMIG.DDL
```

The explain tables belonging to the user ID that is used to connect to the database will be migrated. To migrate explain tables belonging to another user, connect to the database with that user ID and run the command.

Updating Catalog Table Data



You only need to run the **db2upd52** command if you are upgrading from Version 5 to Version 5.2 of DB2. If you upgrade from Version 2 to Version 5.2 of DB2, database migration will update the data types and the SYSFUN schema.

For DB2 Version 5.2, the SYSIBM.SYSDATATYPES catalog table includes a new data type, BIGINT, and other associated functions in the SYSFUN schema. To support these new features, we recommend that you run the **db2upd52** command for each database owned by a DB2 instance being upgraded to Version 5.2. The command adds the new SYSFUN functions signatures that support the BIGINT data type as follows:

- Add SYSIBM.BIGINT to the SYSIBM.SYSDATATYPES catalog table.
- Add the following function signatures to the SYSIBM.SYSFUNCTIONS table:

```
create function sysfun.abs(bigint) returns bigint ...
create function sysfun.absval(bigint) returns bigint ...
create function sysfun.ceil(bigint) returns bigint ...
create function sysfun.ceiling(bigint) returns bigint ...
create function sysfun.floor(bigint) returns bigint ...
create function sysfun.mod(bigint,bigint) returns bigint ...
create function sysfun.power(bigint,bigint) returns bigint ...
create function sysfun.round(bigint,integer) returns bigint ...
create function sysfun.sign(bigint) returns bigint ...
create function sysfun.trunc(bigint,integer) returns bigint ...
create function sysfun.truncate(bigint,integer) returns bigint ...
```

The **db2upd52** command also adds the REAL data type to the SYSIBM.SYSDATATYPES catalog table and the function signature (SYSFUN.UCASE) to the SYSIBM.SYSFUNCTIONS table. For information on the syntax of the **db2upd52** command, refer to the *Command Reference*.



To install clients, see Chapter 5, "Installing DB2 Clients" on page 55.

Chapter 18. Using the Client Configuration Assistant



The information in this section describes how to use the Client Configuration Assistant (CCA) to perform administration tasks on OS/2 and Windows 32-bit operating systems. Before you can use the Client Configuration Assistant, be sure that you have completed the instructions in "Before You Begin" on page 43. You should also refer to "Setting Up the Administration Server to Use the CCA and the Control Center" on page 97.

When you install the DB2 server, if your protocols are set up and configured, DB2 server communications are automatically configured to allow DB2 to accept requests from remote clients.



The CCA can be installed on OS/2 and Windows 32-bit operating systems. If it is not installed on your system, you can rerun the setup program and select to install the CCA.

With the CCA, you can:

- Configure database connections that applications can use.
- Update or delete existing configured database connections.
- Display the information for existing configured connections.
- Test a connection to a database.
- Enable or disable databases to be configured as CLI or ODBC data sources.
- Import or export client profiles which contain information for the setup of a client.
- Update client configuration settings.
- Discover remote databases (if enabled).
- Bind user applications and utilities to databases.
- Change your password.

Configuring Database Connections



To complete the steps in this section, you must be logged on to the local system as a user with System Administrative (SYSADM) authority on the instance.



You can configure database connections through the Add Database SmartGuide, by:

- Using a profile as a source of information to add database connections, see “Using an Access Profile” on page 160.
 - Searching the network for databases, see “Searching the Network for Databases” on page 163.
 - Adding database connections manually, see “Manually Configure a Connection to a DB2 Database” on page 168.
-

Using an Access Profile

You can use either a Server profile or a Client profile to configure database connections on a client.

Server Profiles

Server profiles can be generated for a DB2 server. They contain information about instances on the server system, and databases within each instance. The information for each instance includes the protocol information required to set up a client to connect to databases in that instance.

To generate server profiles, use the Generate Access Profile function provided in the Control Center. When a profile is generated for a DB2 server system, it includes server instances that have the *discover_inst* configuration parameter and databases with the *discover_db* configuration parameter set to *ENABLE*. The *discover* parameter in the Administration Server's configuration file must be set to either *SEARCH* or *KNOWN* to generate a profile for a server system.

For information on setting the *discover_inst*, *discover_db* and *discover* configuration parameters, see “Setting Discovery Parameters” on page 167.

Generating a Server Profile: To generate an access profile, perform the following steps:

- 1 Click on **Start** and select **Programs->DB2 for Windows NT->Administration Tools->Control Center**.
- 2 Click on the **[+]** sign beside the **Systems** icon to get a list of systems.
- 3 Select the system to be profiled and click on the right mouse button.
- 4 Select the **Generate access profile** option.
- 5 Select the path and type in a file name for this profile, then click on the **Generate** push button.

To process a server profile and add its databases to the client's connection configuration list, use the Client Configuration Assistant's Import or Add functions. Using the Add function is the preferred method.

Using a Server Profile on the Client: To add a database using the Add function:

- 1** Click on **Start** and select **Programs->DB2 for Windows NT->Client Configuration Assistant**.
- 2** Click on the **Add** push button.
- 3** Select the **Use an access profile** radio button, and click on the **Next** push button.
- 4** Click on the **Browse** push button and select the profile for the server that you want to access.
- 5** Select a database to be added and click on the **Next** push button.
- 6** If desired, you can select the **Alias** and **ODBC** tabs at the top of the panel to specify a database alias name for the database, or to select CLI/ODBC options for the database. This step is optional.
- 7** Click on the **Done** push button.
- 8** You can test the connection that has been added by clicking on the **Test Connection** push button.

Client Profiles

Information in client profiles can be added to another client using the:

- Add function, which allows you to selectively process the database connection data in the client profile and add it to a new client. (This function invokes the Add Database SmartGuide.)
- Import function, which allows you to selectively process database connection data, client configuration settings, and CLI or ODBC common parameters.

Note: This scenario assumes that the database connections configured on one client will be exported and used to set up one or more additional clients.

Exporting a Client Profile: Client profiles are generated from clients using the Export function of the CCA. Use the Export function to copy the database information used by one client to other clients.

The information contained in a client profile is determined during the export process. Depending on the settings chosen, it can contain the existing client's:

- Database connection information (including CLI or ODBC settings).
- Client settings (including database manager configuration parameters).
- CLI or ODBC common parameters.

- Configuration data for the local APPC communications subsystem.

The Export function can be used to generate a customized profile that can be imported on another client to set it up, or update it.

To export a profile from the client, configure the client for communications and do the following:

- 1 Enter the **db2cca admin** command at a command prompt to start the Client Configuration Assistant in administrator mode. (You must belong to the local Administrator group to perform this command.)



The Export function of the CCA is only available when the CCA is started in administrator mode. The CCA can be started permanently in administrator mode by modifying the **Client Configuration Assistant** icon to add the parameter *admin* to the startup. To add the parameter *admin* to the CCA startup:

- 1 Select the **CCA** icon and click on the right mouse button.
- 2 Select the **Properties** option.
- 3 Add the *admin* parameter to the:
 - **Parameters** field on OS/2
 - **Target** field (located on the **Security** tab) on Windows 32-bit operating systems.

- 2 Click on the **Export** push button.
- 3 Select the databases to be exported from the **Available DB2 Databases** window, and add them to the **Databases to be exported** window.
- 4 Select the check boxes that correspond to the options that you want to set up for the target client.

To customize settings, click on the appropriate **Customize** push button. The settings that you customize will only affect the profile to be exported, no changes will be made to your workstation. For more information, click on the **Help** push button.
- 5 Click on **OK**.
- 6 Enter a path and file name for the Client profile.

Importing a Client Profile:: Perform these steps at the client that you want to set up.

- 1 Click on **Start** and select **Programs->DB2 for Windows NT->Client Configuration Assistant**.
- 2 Click on the **Import** push button.
- 3 Select the path and filename of the client profile you want to import and click on **OK**.

- 4 The Import Client Profile window opens. Select the items you want to import. For more information, click on the **Help** push button.
- 5 Click on **OK**.



If databases are contained in the client profile that you are importing, and you select to import them, the Add Database SmartGuide starts to allow you to selectively import the databases you want to connect to.

Searching the Network for Databases

Instead of entering protocol information to make a connection to remote database servers, you can use the CCA to find all the databases on your local network by following these steps:



The following scenario assumes that messages used by the *Search* method of discovery are not filtered by your network, and that the installation defaults on the client and the server have not been changed. To run the following scenario, you must ensure that the *discover* parameter in the database manager configuration file on the client, and the *discover* parameter in the Administration Server configuration file on the server are set to *SEARCH*. Also, the *discover_comm* parameter must be set to the search protocols used on the client and Administration Server.

- 1 Start the Client Configuration Assistant (CCA).
- 2 Click on the **Add** push button on the CCA's main panel to start the Add Database SmartGuide.
- 3 Select the **Search the network** radio button, and click on the **Next** push button.
- 4 Click on the **[+]** sign beside the **Known Systems** icon to list all the systems known to your client.
 - a Click on the **[+]** sign beside a system to get a list of the instances and databases on it. Select the database that you want to add and proceed to Step 5.
 - b If the system that contains the database that you want is not listed, click on the **[+]** sign beside the **Other Systems (Search the network)** icon to search the network for additional systems. Click on the **[+]** sign beside a system to get a list of the instances and databases on it. Select the database that you want to add and proceed to Step 5.
 - c If the system you want is still not listed, it can be added to the list of systems by clicking on the **Add Systems** push button.

- 5 If desired, you can select the **Alias** and **ODBC** tabs at the top of the panel to specify a database alias name for the database, or to select CLI/ODBC options for the database. This step is optional.
- 6 Click on the **Done** push button.
- 7 You can test the connection that has been added by clicking on the **Test Connection** push button.

Searching the network can be customized to meet the needs of individual organizations. The material that follows provides details on this customization. Refer to the *Administration Guide* for more information on individual configuration parameters and profile registry values.

Choosing a Discovery Method

Network searching uses a DB2 facility called *Discovery* to obtain information from DB2 servers. This information is used to configure clients for database connections.

Note: Discovery cannot retrieve information about pre-Version 5 DB2 systems.

Two discovery methods are available for searching the network:

- **Known**

Known discovery allows you to discover instances and databases on systems that are known to your client, and add new systems so that their instances and databases can be discovered.

Initially, the list of systems will be blank; however, if you are running the CCA on the server, an entry for the local server will be shown. Add systems to the list by clicking on the **Add System** push button. To use this option you must know a few details about the Administration Server on the DB2 system to be searched:

- A protocol configured and running on the Administration Server.
- The protocol's configuration information.

The Administration Server will listen for Known discovery requests, from clients, on the protocols specified by the *db2comm* registry value in the Administration Server.

- **Search**

This mode provides all of the facilities of Known discovery, and adds the option to allow your local network to be searched for DB2 servers.



The TCP/IP host name returned to a client by Search discovery is the same host name that is returned by your DB2 server system when you enter the **hostname** command. On the client, the IP address that this host name maps to is determined by either the TCP/IP domain name server (DNS) configured on your client machine or, if a DNS is not configured, a mapping entry in the client's hosts file. If you have multiple adapter cards configured on your DB2 server system, you must ensure that TCP/IP is configured on the server to return the correct hostname, and that the DNS or local client's hosts file, maps the hostname to the IP address desired.

Search may appear to be a simpler discovery method. However, in larger networks, network routers and bridges can filter the messages that *Search* uses to find DB2 servers on the network, resulting in an incomplete or even empty list. In this case, use the *Add System* method; its messages are not filtered by routers and bridges. If in doubt, contact your network administrator for assistance.

To have the server support Known discovery, set the *discover* parameter in the Administration Server configuration file to *KNOWN*. To have it support Search discovery, set this parameter to *SEARCH* (which also supports all the facilities of Known discovery). To prevent discovery of a server, and all of its instances and databases, set *discover* to *DISABLE*.

On the client, enabling discovery is also done using the *discover* parameter; however, in this case, the *discover* parameter is set in the client instance (or a server acting as a client) as follows:

- Known** Allows the CCA to refresh systems in the known list, and to add new systems to the list by using the **Add Systems** push button. When the *discover* parameter is set to *KNOWN*, the CCA will not be able to search the network.
- Search** Enables all the facilities of Known discovery, and enables network searching.
- Disable** Disables discovery. In this case, the **Search the network** option is not available in the Add Database SmartGuide.



The *discover* parameter defaults to *SEARCH* on all client and server instances. The *discover* parameter defaults to *SEARCH* on all Administration Servers except Administration Servers installed in a UNIX Extended Enterprise Edition environment, where *discover* defaults to *KNOWN*.

Additional Settings for Search Discovery

Search discovery requires that the *discover_comm* parameter be set on both the server (in the Administration Server's configuration file) and the client (in the database manager configuration file).

The *discover_comm* parameter is used to control the communication protocols that the server will listen to for search requests from clients, and that clients will use to send out search requests. The *discover_comm* parameter can be any combination of TCP/IP and NetBIOS; the protocols supported by Search discovery.

On the Administration Server, the values specified for *discover_comm* must be equal to, or a subset of, the values set for *db2comm*.



To avoid problems with the Control Center and the Client Configuration Assistant, ensure that the *DB2COMM* parameter is set in the DB2 registry using the *db2set* command. It is not recommended that you use any other mechanism to set the *DB2COMM* value.

On the server, the *discover_comm* parameter is set in the Administration Server's configuration file. On the client (or a server acting as a client), *discover_comm* is set in the database manager configuration file.

Note: When using Search discovery, at least one protocol specified by the *discover_comm* parameter on the client must match those specified by the *discover_comm* parameter on the Administration Server. If there is no match, the server will not respond to the client's requests.



Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, see Chapter 15, "Controlling Your DB2 Environment" on page 119.

In addition, there are two DB2 profile registry values that can be used to tune Search discovery via NetBIOS on the client: *db2discoverytime* and *db2nbdiscoverrcvbufs*. The default values should be suitable in most cases. For more information, refer to the *Administration Guide*.

Hiding Server Instances and Databases from Discovery

You may have multiple instances, and multiple databases within these instances, on a server. You may want to hide some of these from the discovery process.

To allow clients to discover server instances on a system, set the *discover_inst* database manager configuration parameter in each server instance on the system to *ENABLE* (this is the default value). Set this parameter to *DISABLE* to hide this instance and its databases from discovery.

To allow a database to be discovered from a client, set the *discover_db* database configuration parameter to *ENABLE* (this is the default value). Set this parameter to *DISABLE* to hide the database from discovery.

Setting Discovery Parameters

discover and discover_comm

The *discover* and *discover_comm* parameters are set in the Administration Server configuration file on the server system, and in the database manager configuration file on the client. Set these parameters as follows:

- On the Administration Server:

Update the Administration Server's configuration file, in the command processor, as follows:

```
update admin cfg using discover [ DISABLE ] KNOWN ] SEARCH ]
update admin cfg using discover_comm [ NETBIOS,TCPIP ]
```

Stop and restart the Administration Server by entering the following commands:

```
db2admin stop
db2admin start
```

Note: Search Discovery will only operate on TCP/IP and NetBIOS.

- On the client:

- 1 Start the Client Configuration Assistant (CCA).
- 2 Click on the **Client Settings** push button.
- 3 Select the **Communications** tab.
- 4 Select the parameters that you want to modify from the **Parameters** window.
- 5 Select a value for the parameter that you want to modify from the **Value** box.
- 6 Click on the **OK** push button to close the **Client Settings** window. A DB2 message window opens.
- 7 Click on the **OK** push button and restart your applications so that your changes can take affect.



If the *discover_comm* includes *NETBIOS*, you must ensure that the Workstation name (*nname*) configuration parameter is set for the both the client and the Administration Server. Also, you must ensure that the *db2nbadapters* registry value is set to the Adapter number that you want to use. For more information, refer to the *Administration Guide*.

discover_inst and discover_db

Use the Control Center to set the *discover_inst* and *discover_db* parameters. To set these parameters, follow these steps:

- 1 Start the Control Center.
- 2 Click on the **[+]** sign beside the **Systems** icon to get a list of systems.

- 3 Click on the **[+]** sign beside a system icon to get a list of instances on it.
- 4 Select the instance that you want to configure and click on the right mouse button.
- 5 Select the **Configure** option from the pop-up menu. The Configuration Instance window opens.
- 6 Select the **Environment** tab and select the *discover_inst* parameter from the **Parameters** box.
- 7 Select the desired value from the **Value** box and click on **OK**.
- 8 Click on the **[+]** beside an instance icon to get a list of databases in it.
- 9 Select the database that you want to configure and click on the right mouse button.
- 10 Select the **Configure** option from the pop-up menu. The Configuration Database window opens.
- 11 Select the **Environment** tab and select the *discover_db* parameter.
- 12 Select the desired value from the **Value** box and click on **OK**.

db2discovertime and db2nbdiscoverrcvbufs

The *db2discovertime* and *db2nbdiscoverrcvbufs* profile registry values are set in the client instance (or a server acting as a client). Set these parameters as follows:

- To set the *db2discovertime* registry value to 60 seconds, enter the following command:


```
db2set db2discovertime=60
```

This specifies that Search discovery should wait 60 seconds for a response from servers.
- To set the *db2nbdiscoverrcvbufs* registry value to 20, enter the following command:


```
db2set db2nbdiscoverrcvbufs=20
```

This specifies the number of NetBIOS buffers that will be allocated for concurrent response messages from discovered servers.

Manually Configure a Connection to a DB2 Database

Manually configuring a database connection requires you to know:

- One of the protocols supported by the server instance containing the database.
- The protocol connection information required to configure the connection to the server instance.
- The name of the database on the server system.

With this information, the SmartGuide will guide you through the steps necessary to add the database connection.

- 1 Start the Client Configuration Assistant (CCA).
- 2 Click on the **Add** push button on the CCA's main panel to start the Add Database SmartGuide.
- 3 Select the **Manually configure a connection to a DB2 database** radio button, and click on the **Next** push button.
- 4 Select the protocol that you will use to connect to the database and click on the **Next** push button.
- 5 Enter the required protocol parameters and click on the **Next** push button.
- 6 In the **Target database** panel, enter the name of the remote database alias in the **Target Database** field.



When a database is created on the remote server, if a database alias is not specified during database creation, the database is created with a database alias=*database_name*; otherwise, the database alias is the name specified.

- 7 If desired, you can select the **Alias** and **ODBC** tabs at the top of the panel to specify a local database alias name or select **CLI/ODBC** options for the remote database. This step is optional.
- 8 Click on the **Done** push button.
- 9 You can test the connection that has been added by clicking on the **Test Connection** push button.



When you add a database using this configuration method, the Client Configuration Assistant will generate a default node name for the database.

For example, if you used TCP/IP to add the database to your system, the instance name could be *TCP0000*.

Chapter 19. Setting up Communications on the Server Using the Command Line Processor

This section describes how to configure your DB2 server to communicate with remote client workstations. Also described in this section are considerations when configuring the DB2 Administration Server for communications. The Control Center and the Discovery function of the Client Configuration Assistant are dependent on the Administration Server's protocol configuration.

Follow the instructions in this section if you deselected a detected communication protocol during the installation or, you have added a communication protocol to your network since running the DB2 setup program.



If you have installed the Control Center, you can use the Setup Communications function to configure an instance for communications; however, it cannot be used to set up communications for an Administration Server. The Control Center can be installed on OS/2 and Windows NT servers, or on OS/2 and Windows 32-bit operating systems clients.

The Setup Communications function cannot update the APPC communications subsystem; however, it will update the required DB2 instance parameters on the server.

For instructions on using the command line processor, see Chapter 14, "Entering DB2 Commands and SQL Statements" on page 113. For a description of database naming rules, see Appendix D, "Naming Rules" on page 233.

Setting the DB2COMM Registry Value

Your DB2 server can support multiple communication protocols concurrently; however, you only need to enable the protocols that you want to use.

If you deselected a detected protocol during installation, did not install the Control Center, or have added a communication protocol to your network since installation, you must update the *db2comm* registry value with the protocol that you now want to support.

The *db2comm* registry value determines which protocol's connection managers will be enabled when the database manager is started. You can set this value for multiple communication protocols by separating the keywords with commas.

For your DB2 server, *DB2COMM* can be any combination of the following keywords:

appc starts APPC support
ipxspx starts IPX/SPX support
netbios starts NetBIOS support
npipe starts NAMED PIPE support
tcPIP starts TCP/IP support



To avoid problems with the Control Center and the Client Configuration Assistant, ensure that the *DB2COMM* parameter is set in the DB2 registry using the `db2set` command. It is not recommended that you use any other mechanism to set the *DB2COMM* value.

To set the *db2comm* registry value for the current instance, enter the **db2set DB2COMM=protocol_names** command. Change the *protocol_names* parameter to reflect those protocols that you want to start when the database manager is started.

For example, to set the database manager to start connection managers for the APPC and TCP/IP communication protocols, enter:

```
db2set DB2COMM=appc,tcPIP
db2stop
db2start
```



If you are setting up communications for the Administration Server, use the **db2set** command as follows:

```
db2set DB2COMM=appc,tcPIP -i DB2DAS00
db2admin stop
db2admin start
```

where:

DB2DAS00 The name of the Administration Server's instance. If you are not sure of the Administration Server name, enter the following command to view it:

```
db2set DB2ADMINSERVER
```

When the **db2start** command is reissued, connection managers for the protocols specified by the *db2comm* registry value are started.



If there are problems starting a protocol's connection managers, a warning message is displayed and the error messages are logged in the `db2diag.log` file.

Additionally, any failure encountered while executing the **db2start** command is logged to a time-stamped file in the log subdirectory within the instance directory. This file contains the **db2start** results from each database partition server in the instance.

Refer to the *Troubleshooting Guide* for information on the `db2diag.log` file.

If *db2comm* is undefined or set to null, no protocol connection managers are started when the database manager is started.

For more information on setting DB2 parameters, see “Controlling the DB2 Profile Registry” on page 120.



You are now ready to configure the server workstation to use any of the following communication protocols:

- Named Pipes - see “Configuring Named Pipes on the Server”
 - TCP/IP - see “Configuring TCP/IP on the Server”
 - NetBIOS - see “Configuring NetBIOS on the Server” on page 177
 - IPX/SPX - see “Configuring IPX/SPX on the Server” on page 181
 - APPC - see “Configuring APPC on the Server” on page 184
-

Configuring Named Pipes on the Server

To access a remote server through Named Pipes, you must first have installed and configured communication software for both the client and DB2 server workstations. See “Software Requirements” on page 37 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 40 for the supported communication protocols for your particular client and server.

If you have read and completed the instructions in “Setting the DB2COMM Registry Value” on page 171, there are no further steps required to set up Named Pipe communications on the server or to support the Administration Server.



Now that you have configured the server, you are ready to install a DB2 client. Go to Chapter 5, “Installing DB2 Clients” on page 55 for more information.

Configuring TCP/IP on the Server

To access a remote server through TCP/IP, you must first have installed and configured communication software for both the client and DB2 server workstations. See “Software Requirements” on page 37 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 40 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in "Setting the DB2COMM Registry Value" on page 171.



If you are setting up communications for the Administration Server, it was configured for TCP/IP (using the registered Port number 523) when it was created. There are no further steps required to enable the Administration Server to use TCP/IP.

The following steps are required to set up TCP/IP communications:

- 1 Identify and record parameter values.
- 2 At the server:
 - a Update the services file.
 - b Update the database manager configuration file.



Due to the characteristics of the TCP/IP protocol, the TCP/IP subsystem may not be immediately notified of the failure of a partner on another host. As a result, a client application accessing a remote DB2 server using TCP/IP, or the corresponding agent at the server, may sometimes appear to be hung. DB2 uses the TCP/IP SO_KEEPALIVE socket option to detect when there has been a failure and the TCP/IP connection has been broken.

If you are experiencing problems with your TCP/IP connection, refer to the *Troubleshooting Guide* for information on this parameter and other common TCP/IP problems.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

<i>Table 7. TCP/IP Values Required at the Server</i>			
Parameter	Explanation	Sample Value	Your Value
Connection Port <ul style="list-style-type: none"> • Connection Service name (<i>svcename</i>) • Port number/Protocol (<i>port_number/tcp</i>) Interrupt Port <ul style="list-style-type: none"> • Interrupt Service name • Port number/Protocol (<i>port_number + 1/tcp</i>) 	Values required in the <i>services</i> file. The Connection and Interrupt Service names are arbitrary, but each must be unique within the <i>services</i> file; ensure that this number does not conflict with the port numbers used by the Fast Communication Manager (FCM). The Connection and Interrupt Ports must be unique within the <i>services</i> file, with the Interrupt Port number equal to the Connection Service Port number plus one.	server1 3700/tcp server1i 3701/tcp	
Service name (<i>svcename</i>)	The name used to update the Service name (<i>svcename</i>) parameter in the database manager configuration file at the server. This value must be the same as the Connection Service name specified in the <i>services</i> file.	server1	

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample values with your values and record them on your worksheet.

A. Update the Services File

The TCP/IP *services* file specifies ports that the DB2 server will listen on for client requests. The *services* file must contain two entries for each database manager instance.

You need to update the *services* file and specify the ports that you want the DB2 server to listen on for incoming client requests. The first entry describes the Connection Port, and the second entry describes the Interrupt Port.

The *services* file is located in the `\winnt\system32\drivers\etc` directory.



The location of the *services* file can depend on the products you have installed on your system. See your TCP/IP documentation for more information.

Using a local text editor, add the Interrupt Port and Connection Port entries to the services file for TCP/IP support. For example:

```
server1    3700/tcp    # DB2 connection service port
server1i   3701/tcp    # DB2 interrupt connection service port
```

where:

server1 is the Connection Service name
server1i is the Interrupt Service name
3700 and 3701 are the port numbers for the Connection and Interrupt Ports
tcp is the communication protocol that you are using

B. Update the Database Manager Configuration File

You must update the database manager configuration file with the Service name (*svcename*) parameter.

To update the database manager configuration file, perform the following steps:

- 1 Log on to the system with a user account that belongs to the local Administrators group on each machine in your partitioned database system.



If you have trouble logging on to the system, see “Logging on to the System” on page 77.

- 2 Update the database manager configuration file with the Service name (*svcename*) parameter using the following commands in the command line processor:

```
update database manager configuration using svcename svcename
db2stop
db2start
```

For example, if the Connection Service name in the services file was entered as *server1*, use:

```
update database manager configuration using svcename server1
db2stop
db2start
```



The *svcename* used must match the Connection Service name specified in the services file.

After the database manager is stopped and started again, view the database manager configuration file to ensure that these changes have taken effect. Enter the following command in the command line processor:

```
get database manager configuration
```



Now that you have configured the server, you are ready to install a DB2 client. Go to Chapter 5, “Installing DB2 Clients” on page 55 for more information.

Configuring NetBIOS on the Server

To access a remote server through NetBIOS, you must first have installed and configured communication software for both the client and DB2 server workstations. See “Software Requirements” on page 37 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 40 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in “Setting the DB2COMM Registry Value” on page 171.

The following steps are required to set up NetBIOS communications:

- 1** Identify and record parameter values.
- 2** Configure the server:
 - a** Configure the NetBIOS Interface.
 - b** Update the database manager configuration file.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Parameter	Description	Sample Value	Your Value
Adapter number (<i>adapter_number</i>)	The local logical adapters that will be used for the NetBIOS connection. The server uses adapter 0 if this parameter is not configured.	0	
Workstation name (<i>nname</i>)	The NetBIOS name of the server workstation. <i>nname</i> is a name, chosen by the user, where the last 4 characters must be unique among all NetBIOS server nodes in the network. For more information on workstation names, see "Workstation Names (nname)" on page 236	server1	

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample values with your values and record them on your worksheet.

A. Configure the NetBIOS Interface

For a client to connect to a Windows NT server, the network route of the adapter number used must be **Nbf** for native NetBIOS operation. If the *db2nbadapters* parameter is not set, the default local adapter will be 0. If you want to use additional server adapters, or you need to specify a server adapter other than 0, set the *db2nbadapters* parameter to the number that you want to use.



DB2 uses registry values to control its use of the NetBIOS resources on the server. Use the *db2nbadapters* registry parameter when you want to specify a value other than the default Logical adapter number 0.

For the DB2 server, set the *db2nbadapters* parameter by entering the **db2set db2nbadapters=adapter_number** command.

For the Administration Server, set the *db2nbadapters* parameter by entering the **db2set db2nbadapters=adapter_number -i DB2DAS00** command.

For more information, see Chapter 15, "Controlling Your DB2 Environment" on page 119.

To view or modify the NetBIOS interface configuration, which maps network routes to adapter numbers, do the following:

- 1 Click on **Start** and select **Settings->Control Panel**.
- 2 Double-click on the **Network** icon and select the **Services** tab.
- 3 Select the **NetBIOS Interface** icon from the Network Services window and click on the **Properties** push button.
- 4 Scroll through the network routes until you find the Logical adapter number associated with **Nbf** and record it on your worksheet. If this adapter number is associated with **Nbf**, and you do not want to change it, go to Step 7.
- 5 To change the logical adapter number associated with **Nbf**, select the associated **Lana Number**, and click on the **Edit** push button. Enter the new adapter number 0 (or the value that you set for *db2nbadapters*).
- 6 Record the new adapter number associated with **Nbf** on your worksheet.
- 7 Click on **OK**.
- 8 Click on the **Close** push button.
- 9 The Network Settings Change pop-up box opens. You must shutdown and reboot your system for these changes to take effect. Select the **Yes** push button to shutdown and reboot your system or select the **No** push button to shutdown and reboot your system at a later time.



Each adapter number must be uniquely associated with a network route. Windows NT has a built in checking facility that will not allow you to specify the same adapter number for different network routes. If a Network route already exists that is using the adapter number 0, assign a different number to that route. (The valid range for adapter numbers is 0 to 255.) This will allow you to select 0 as the adapter number that corresponds to **Nbf**. Approve the changes by clicking on **OK**.

B. Update the Database Manager Configuration File

You must update the database manager configuration file with the server's Workstation name (*nname*) parameter.

To update the database manager configuration file, perform the following steps:

- 1 Log on to the system with a user account that belongs to the local Administrators group on each machine in your partitioned database system.



If you have trouble logging on to the system, see "Logging on to the System" on page 77.

- 2 Update the database manager configuration file with the server's Workstation name (*nname*) parameter using the following commands in the command line processor:

```
update database manager configuration using nname nname
db2stop
db2start
```

For example, if the server's workstation name (*nname*) is *server1*, use:

```
update database manager configuration using nname server1
db2stop
db2start
```

If you are configuring the Administration Server for NetBIOS, you must update the admin server configuration file. Use the following command:

```
update admin configuration using nname server1
db2admin stop
db2admin start
```



Now that you have configured the server, you are ready to install a DB2 client. Go to Chapter 5, "Installing DB2 Clients" on page 55 for more information.

Autostarting DB2 with NetBIOS

If your NetBIOS protocol was configured when you installed on the instance-owning machine, the setup program automatically created a dependency for the Administration Server. If you created the Administration Server on anything other than the instance-owning machine, you will need to manually create a dependency on NetBIOS for each new instance.

Note: You can only autostart the Administration Server; you cannot autostart the instance.

You can create this dependency with the **db2depnb** command as follows:

- 1 Go to the `x:\sql11ib\misc` directory, where *x*: is the drive on which DB2 was installed
- 2 Enter the **db2depnb** command as follows:

```
db2depnb instance_name
```

This records a dependency on the startup order which causes NetBIOS to start before any DB2 instances start.



If you remove the NetBIOS protocol from your network, you **must** remove the dependencies that were created during installation, and any dependencies that you created for additional instances. Failure to remove these dependencies may cause problems when running DB2 after the NetBIOS protocol has been removed from the network.

To remove a dependency, enter the **db2depnb** command as follows:

```
db2depnb instance_name /r
```

where *instance_name* is the name of the instance for which you removed a dependency.



Now that you have configured the server, you are ready to install a DB2 client. Go to Chapter 5, “Installing DB2 Clients” on page 55 for more information.

Configuring IPX/SPX on the Server

To access a remote database server through IPX/SPX, you must first have installed and configured communication software for both the client and DB2 server workstations. See “Software Requirements” on page 37 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 40 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in “Setting the DB2COMM Registry Value” on page 171.



If you are setting up communications for the Administration Server, it was configured for IPX/SPX (using the registered Socket number 87A2) when it was created. There are no further steps required to enable the Administration Server to use IPX/SPX.

Your DB2 server can be set up to support IPX/SPX client communications via Direct Addressing.

In Direct Addressing, the client connects to the DB2 server by directly specifying the IPX/SPX internetwork address of the server (bypassing the NetWare file server). Using this method, a NetWare file server is not required on the network. Configure DB2 servers using this addressing method if they will be accessed solely by clients using Direct Addressing.

The following steps are required to set up IPX/SPX communications:

- 1 Identify and record parameter values.
- 2 Configure the server:
 - a Update the database manager configuration file.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

<i>Table 9. IPX/SPX Values Required at the Server</i>			
Parameter	Explanation	Our Example	Your Value
File server name (<i>FILESERVER</i>)	A * value indicates that you are using Direct Addressing.	*	
DB2 server object name (<i>OBJECTNAME</i>)	A * value indicates that you are using Direct Addressing.	*	
Socket number (<i>IPX_SOCKET</i>)	Represents the connection end point in a DB2 server's internetwork address. It must be unique for all server instances, and all IPX/SPX applications running on the workstation. It is entered in hexadecimal format. DB2 has registered well known sockets with Novell in the range 0x879E to 0x87A1. If you run more than 4 instances on the server machine, you must prevent socket collisions for instances 5 and up by choosing a socket number that is not 0x0000, in the dynamic socket range 0x4000 to 0x7FFF, or in the range 0x8000 to 0x9100 (these are well known sockets that are registered to various applications). The maximum value for this parameter is 0xFFFF.	879E (default)	
IPX/SPX Internetwork Address	The internetwork address required when configuring a client to communicate with a server using Direct Addressing.	09212700.400011527745.879E	

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample values with your values and record them on your worksheet.

A. Update the Database Manager Configuration File

You must update the database manager configuration file with the File server (*FILESERVER*), DB2 server object name (*OBJECTNAME*), and Socket number (*IPX_SOCKET*) parameters.

To update the database manager configuration file, perform the following steps:

- 1 Log on to the system with a user account that belongs to the local Administrators group on each machine in your partitioned database system.



If you have trouble logging on to the system, see "Logging on to the System" on page 77.

- 2 In the database manager configuration file, update the File server (*FILESERVER*), DB2 server object name (*OBJECTNAME*), and Socket number (*IPX_SOCKET*) parameters using the following commands in the command line processor:

```
update dbm cfg using fileserver FILESERVER objectname OBJECTNAME ipx_socket IPX_SOCKET
db2stop
db2start
```

For example, if you have chosen a value of 879E for the Socket number (*IPX_SOCKET*) parameter, use:

```
update dbm cfg using fileserver * objectname * ipx_socket 879E
db2stop
db2start
```

After the database manager is stopped and started again, view the database manager configuration file to ensure that these changes have taken effect. Enter the following command in the command line processor:

```
get database manager configuration
```



If you are planning to only support clients that use Direct Addressing, you will be required to enter the server's IPX/SPX internetwork address as the DB2 server object name when cataloging the node on the client.

Determine the value for the *OBJECTNAME* parameter by entering the **db2ipxad** command on the server. This command is located in the `sqllib\misc\` directory.

Configuring APPC on the Server

To access a remote server through APPC, you must first have installed and configured communication software for both the client and DB2 server workstations. See “Software Requirements” on page 37 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 40 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in “Setting the DB2COMM Registry Value” on page 171.

The following steps are required to set up APPC communications:

- 1 Identify and record parameter values.
- 2 Configure the server:
 - a Update the database manager configuration file.
 - b Configure the APPC communications subsystem.

Step 1. Identify and Record Parameter Values

Before you proceed through the configuration steps, enter your value for the Transaction program name in the following table.

Parameter	Description	Sample Value	Your Value
Transaction program name (<i>tpname</i>)	The Transaction program that the server listens on for client requests.	nyserver	

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample value with your value as recorded on your worksheet.

A. Update the Database Manager Configuration File

You must update the database manager configuration file with the transaction program name (*tpname*).

To update the database manager configuration file, perform the following steps:

- 1 Log on to the system with a user account that belongs to the local Administrators group on each machine in your partitioned database system.



If you have trouble logging on to the system, see “Logging on to the System” on page 77.

- 2 Update the database manager configuration file with the server's transaction program name (*tpname*) using the following commands in the command line processor:

```
update dbm cfg using tpname tpname
db2stop
db2start
```

For example, if the server's transaction program name (*tpname*) is *nyserver*, use:

```
update dbm cfg using tpname nyserver
db2stop
db2start
```

If you are configuring the Administration Server to use APPC, you must update the admin server configuration file. Use the following command:

```
update admin configuration using tpname nyserver
db2admin stop
db2admin start
```

B. Configure the APPC Communications Subsystem

To configure your DB2 server to accept remote clients using APPC, you need to update the APPC communications subsystem to support the Transaction program name (*tpname*) that the server will use.



Refer to the *DB2 Connect Enterprise Edition Quick Beginnings* manual for detailed instructions on how to configure your APPC communications subsystem to support communications from the Transaction program that you specified for this server.

Chapter 20. Unattended Installation

During a regular installation, you interact with the setup program, filling in a series of panels to set up and configure DB2.

In an unattended installation, you supply the setup and configuration data in a response file that you create before running the installation.

You can use this feature to install a DB2 product or component on Windows 32-bit operating systems:

- Locally, from a CD-ROM (see “Installing DB2 Products from a Remote Server Hard Disk or CD-ROM”).
- Remotely, across a network connection from a remote server hard disk or CD-ROM (see “Installing DB2 Products from a Remote Server Hard Disk or CD-ROM”).
- Remotely, across a network connection to target workstations, using Microsoft Systems Management Server (SMS) if it is installed on your Windows NT system (see “Installing DB2 Remotely Using SMS” on page 198).

Note: If you need to install the same DB2 product on several machines, you can use a response file to create a customized configuration, save the customized response file, then reuse it as a template for each installation.

Installing DB2 Products from a Remote Server Hard Disk or CD-ROM

To install a DB2 product for Windows 32-bit operating systems, from a remote server hard disk or CD-ROM to local or remote workstations, perform the following steps:

- “Step 1. Make DB2 Files Available for Installation” on page 188
- “Step 2. Create a Response File for Unattended Installation” on page 189
- “Step 3. Run the Setup Program from the Workstation” on page 197.

Install a DB2 product only on a drive that is local to the target workstation. Installing on a non-local drive can cause performance and availability problems.

Before You Begin

Read and perform the steps in “Before You Begin” on page 43 before you start the installation to ensure that you have the required items and information you will need.

Step 1. Make DB2 Files Available for Installation

To make the DB2 install files accessible to the machine where DB2 will be installed:

- If performing *remote* installs directly from the CD-ROM:
 - 1 Insert the appropriate CD-ROM into the drive.
 - 2 Ensure that the CD-ROM remains in the drive for as long as the installation requires it. If the drive is often used for other tasks as well, consider copying the install code to the server hard disk instead, and installing from there.
- If installing from a remote server, you must copy the required files from the CD-ROM directories to the remote server's hard disk:
 - 1 Insert the appropriate CD-ROM into the drive.
 - 2 Create a directory on the remote server, for example, q:\db2prods.
 - 3 Use the **cpyssetup.bat** command to copy the files necessary for a particular language install to a remote server.

The syntax of the **cpyssetup.bat** command is:

```
x:\db2\common\cpyssetup.bat q:\db2prods language (for DB2 Server)
```

or

```
x:\db2\winnt95\common\cpyssetup.bat q:\db2prods language (for DB2 CAE)
```

where:

- x: designates your CD-ROM drive
- q: designates the remote server's disk.
- language is the two-character country code that represents your language (for example, EN for English). See Appendix C, “National Language Support (NLS)” on page 231 for further information.

If performing a remote installation, you must enable your network clients to access the DB2 install files.

Grant *shared* access to the directory that you just created, or to the CD-ROM drive.

To set up a share:

- 1 Click on **Start** and select **Programs->Windows Explorer**.

- 2 Select the directory that you want to share, for example `q:\db2prods`.
- 3 Select **File->Properties** from the menu bar. The properties window for the directory will open. Select the **Sharing** tab.
- 4 Select the **Shared As** radio button.
- 5 In the **Share Name** field, enter a share name. For example, enter `db2nt`.
- 6 To specify read access for everyone on Windows NT:
 - a Click on the **Permissions** push button. The Access Through Share Permissions window opens.
 - b In the **Name** window, make sure that the **Everyone** option is selected.
 - c Click on the **Type of Access** drop down box and select the **Read** option.
 - d Click on **OK** twice.

In our example scenario, `q:\db2prods` uses a share name of `db2nt`. This value will be used in the examples that follow.

Step 2. Create a Response File for Unattended Installation

During an unattended installation, you supply the setup and configuration data in a response file that you create before running the installation. DB2's install package includes a number of ready-to-use sample response files, with default entries already in place. You can use these sample response files or create your own (using the samples provided as a basis for your customized installation).

The sample response files for DB2 Universal Database Extended Enterprise Edition or the DB2 Client Application Enabler are located in:

- `x:\db2\common` (for DB2 Server CD-ROM)
- `x:\db2\winnt95\common` (for DB2 Client Pack CD)

where `x:` designates the CD-ROM drive.

There are three different server response files that you can choose from, depending on how you want to use the machine on which you are installing DB2. Each of these response files is mutually exclusive so you should not use one response file and then try to run another response file on the same machine.



If you are planning to add another database partition server in order to run multiple logical nodes, go to "Adding a Database Partition Server to an Instance" on page 88.

The following are the sample response files provided on your CD-ROM.

db2eeeio.rsp

Use this response file if you want to install DB2 on the machine that will be the instance-owning database partition server. If there is an existing version of DB2 on this machine, you can choose to migrate the default instance to the Version 5 multipartition format (the default instance retains its name), or create a new default instance. All other existing DB2 instances will be migrated to the Version 5 single-partition format.

db2eeenn.rsp

Use this response file if you want to add a new node to an existing database partition system. This machine will work collectively with the instance-owning database partition server and other nodes to handle requests for data. All existing DB2 instances will be migrated to the Version 5 single-partition format.

db2eeesp.rsp

Use this response file if you want to migrate existing DB2 instances to the Version 5 single-partition format. This option does not allow you to exploit the scalability and power of a multipartition database system.

Note: This response file can only be used if there is a previous version of DB2 on the machine on which you are installing.

db2cae.rsp

Use this response file if you want to install DB2 clients. Refer to the *Installing and Configuring Clients* online book for more information.



This response file is also located on the DB2 Client Application Enabler CD-ROM.

Edit the sample response file for the installation that you want to perform as follows.

- 1 Open the appropriate sample response file. For example to perform an unattended installation of an instance-owning machine, you would edit the `db2eeeio.rsp` sample response file. The response file contains:

- Keywords unique to installation
- Registry value/environment variable settings
- Database manager configuration parameter settings.

For more information on:

- DB2 environment variables and registry values, refer to “DB2 Registry Values and Environment Variables” on page 124 or the *Administration Guide*.
- Database manager configuration parameters, refer to the *Administration Guide*.

- 2 To activate an item in the response file, remove the asterisk (*) to the left of the keyword. The possible settings are listed to the right of the current setting. Erase these settings and type in a new one.

An example section of a response file appears below:

```

...
FILE             = c:\sql11b
TYPE             = 0,1,2 (0=compact, 1=typical, 2=custom)
*COMP           = Control Center
...

```

Keywords that are unique to installation are only specified in a response file during an unattended installation. The following is a list of keywords unique to installation:

- **PROD**
Specifies the DB2 product being installed.
- **FILE**
Specifies the destination directory for DB2.
Note: Check the value for the FILE keyword. FILE specifies where to install DB2. If you do not change this value, it defaults to x:\sql11b, where x: is the drive on which Windows NT or Windows 95 is installed.
- **REBOOT**
Specifies whether to reboot the system following installation.
- **INSTALL_OPTION**
There are three options. For each option, there is a different sample response file. It is important to keep the default value of this option.

INSTANCE_OWNING

Specifies that this machine will become the the instance-owning database partition server.

If there is an existing version of DB2 on the machine, you must fill in the **MIGRATE_DEFAULT_INSTANCE** keyword to specify if you want to use the existing default instance as the default instance in the new partitioned database system. If you do not use the existing default instance as the default instance in your new partitioned database system, the setup program will create a new one called DB2MPP.

If there is no existing version of DB2 on the machine, this option creates the default instance on what will become the instance-owning database partition server. The default instance is called DB2MPP.

NEW_NODE

The setup program adds this machine as a node to an existing partitioned database system. All existing DB2 instances are migrated to the Version 5 single-partition format.

SINGLE_PARTITIONED

All existing DB2 instances are migrated to the Version 5 single-partition format. This option does not allow you to exploit the scalability and power of a multipartition database system. This keyword is only valid if there is a previous version of DB2 on the machine.

- TYPE

Specifies the type of install. The options are:

- 0=Compact
- 1=Typical (default)
- 2=Custom

For 0 and 1 (compact and typical), any Component Keyword (COMP) value will be ignored.

For all types, default communication values will be used where applicable, as follows:

- a** DB2.DB2COMM and ADMIN.DB2COMM will be set for the protocols detected on your system.
- b** If a protocol is detected on the system, or if you have specified a protocol for the DB2.DB2COMM or ADMIN.DB2COMM parameters, values will be generated so that the protocol will be enabled for communications when the setup program completes the installation.

Default values will be generated for the following:

- For NetBIOS:

DB2.DB2NBADAPTERS	ADMIN.DB2NBADAPTERS
DB2.NNAME	ADMIN.NNAME

- For TCP/IP:

DB2.PORT_NUMBER	ADMIN.SVCENAME
DB2.SVCENAME	

- For IPX/SPX:

DB2.FILESERVER	ADMIN.FILESERVER
DB2.OBJECTNAME	ADMIN.OBJECTNAME
DB2.IPX_SOCKET	

- For APPC:

DB2.TPNAME	ADMIN.TPNAME
------------	--------------

Note: You may override these values; however, if the associated protocol has not been detected or specified, the values will not be used.

- COMP

Specifies the components you want installed. The setup program automatically installs components that are required for DB2, and ignores requested components that are not available.

Note: Component selections have no effect unless you specify a custom (2) installation.

The following table lists COMP keywords and the components they represent.

Component Keywords for Windows NT Unattended Installation

Keyword Value	Component

COMPONENTS	
=====	
GRAPHICAL_TOOLS	Graphical Tools
CLIENT_CONFIGURATION_ASSISTANT	Client Configuration Assistant
CONTROL_CENTER	Control Center
PERFORMANCE_MONITOR	Performance Monitor
VISUAL_EXPLAIN	Visual Explain
ODBC_DRIVER	DB2 ODBC Driver
SAMPLE_APPLICATIONS	Sample Applications
CONNECT_SUPPORT	Connect Support
SNA_SUPPORT	Integrated SNA Support
JAPANESE_CONVERSION	Japanese Conversion Support
KOREAN_CONVERSION	Korean Conversion Support
T_CHINESE_CONVERSION	Traditional Chinese Conversion Support
S_CHINESE_CONVERSION	Simplified Chinese Conversion Support
BOOKS	
=====	
API_REFERENCE	API Reference
ADMINISTRATION_GUIDE	Administration Guide
ADMINISTRATION_GETTING_STARTED	Administration Getting Started
DB2_PROGRAMMING_GETTING_STARTED	Road Map to DB2 Programming
BUILDING_OS2_WIN_APPLICATIONS	Building Applications for Windows and OS/2 Environments
CLI_GUIDE_REFERENCE	CLI Guide and Reference
COMMAND_REFERENCE	Command Reference
CONNECT_PERSONAL_QUICK	DB2 Connect Personal Edition Quick Beginnings
CONNECT_USERS_GUIDE	DB2 Connect User's Guide
UNIVERSAL_DB_QUICK	DB2 Universal Database Quick Beginnings
SQL_PROGRAMMING_GUIDE	Embedded SQL Programming Guide
GLOSSARY	Glossary
MASTER_INDEX	Master Index
MESSAGES_REFERENCE	Messages Reference
SQL_GETTING_STARTED	SQL Getting Started
SQL_REFERENCE	SQL Reference
SYSTEM_MONITOR_GUIDE	System Monitor Guide and Reference
CONNECT_ENTERPRISE_QUICK	DB2 Connect Enterprise Edition Quick Beginnings
TROUBLE_SHOOTING_GUIDE	Troubleshooting Guide
CONNECTIVITY_SUPPLEMENT	DB2 Connectivity Supplement
UNIVERSAL_DB_PERSONAL_QUICK	DB2 Universal Database Personal Edition Quick Beginnings
REPLICATION-GUIDE	DB2 Replication Guide and Reference
INSTALLING_CONFIGURING_CLIENTS	Installing and Configuring DB2 Clients

- **AUTO_CTRL_CENTER**

Specifies whether or not to automatically start the Control Center each time the system is rebooted.

Note: By default, the Control Center starts automatically unless this parameter is set to *NO*.

Keywords for an Instance-Owning Database Partition Server Installation

The following section contains information about each of the keywords used in the response files. If a keyword is prefaced with DB2, then it will apply to the default instance of the machine you are installing on. If a keyword is prefaced with ADMIN, then it will apply to the Administration Server instance of the machine you are installing on. Keywords that are not prefaced with either DB2 or ADMIN are not instance specific.

- **MIGRATE_DEFAULT_INSTANCE**

Specifies if you want to use the existing default instance as the default instance in the partitioned database system. This keyword is only selectable if there is an existing version of DB2 on the machine, and if you have chosen `INSTALL_OPTION=INSTANCE_OWNING`.

The options are:

- YES** Specifies that you want to use the existing default instance. The setup program migrates the existing instance to the DB2 Version 5 multipartition format (the default instance retains its name). This machine becomes the the instance-owning database partition server. All other existing DB2 instances will be migrated to the DB2 Version 5 single-partition format.
- NO** Specifies that you want to create a new default instance. The setup program creates a default instance, called DB2MPP, on what will become the instance-owning database partition server. All existing DB2 instances will be migrated to the DB2 Version 5 single-partition format.

- **DB2.DB2PORTRANGE**

Specifies the ports that you want to allocate for communication between database partition servers.

The port range you specify must be greater than or equal to the number of logical nodes running on the machine with the most multiple logical nodes in the partitioned database system.

The setup program verifies that this port range is available on this machine; however, you must ensure that this range is available on all of the machines that will make up your partitioned database system. To verify this port range, check the services file on each of the machines that will participate in the partitioned database system. The services file is located in the

x:/winnt/system32/drivers/etc directory, where x: is the drive on which you installed Windows NT.

Port numbers are separated by a colon, for example:

```
DB2.DB2PORTRANGE=20000:20007
```

- **DB2.DB2ACCOUNTNAME**

Specifies the domain username that DB2 will use when it logs on as a Windows NT service. For information on a valid domain username, see “Before You Begin” on page 43.

This keyword is required to perform an instance-owning machine installation option.

- **DB2.PASSWORD**

Specifies the domain user account's password that DB2 will use when it logs on as a Windows NT service.

This keyword is required to perform an instance-owning machine installation option.

- **DB2.DB2CCMSRV**

Specifies a TCP/IP port to be used by the DB2 Performance Monitor component of the Control Center. This same port must be reserved on all machines that will make up the partitioned database system.

The setup program verifies that this port number is available on this machine; however, you must ensure that this range is available on all of the machines that will make up your partitioned database system. To verify that this port is available, check the `services` file on each of the machines that will participate in the partitioned database system. The `services` file is located in the x:/winnt/system32/drivers/etc directory, where x: is the drive on which you installed Windows NT.

- **AUTOSTART_FIRST_STEPS**

Specifies whether or not to automatically start First Steps the first time the server system is rebooted.

Note: By default, First Steps will not be automatically started. When installing on remote systems, you can set the parameter to YES so that First Steps will start automatically. We suggest that you do not start First Steps until you have completed installing all the machines that will make up your partitioned database system.

- **ADMIN.USERID**

Specifies the username that the DB2 Administration Server will use to log on to the system. For more information on this username, see “Before You Begin” on page 43.

- **ADMIN.PASSWORD**

Specifies the password for the DB2 Administration Server.

Keywords for a New Node Installation

- NEW_NODE.INSTANCE_OWNING_MACHINE_NAME
Specifies the computer name of the instance-owning machine.
This keyword is required to perform a new node installation.
- NEW_NODE.INSTANCE_NAME
Specifies the name of the instance that you want the new node to participate in.
This keyword is required to perform a new node installation.
- NEW_NODE.NODE_NUMBER
Specifies the unique node number for the database partition server. The node number can be any unique number in the range 1-999.
- NEW_NODE.DB2ACCOUNTNAME
Specifies the domain user account that DB2 will use when it logs on as a Windows NT service.
This keyword is required to perform a new node installation.
- NEW_NODE.PASSWORD
Specifies the domain user account's password that DB2 will use when it logs on as a Windows NT service.
This keyword is required to perform a new node installation.

Keywords for a Single-Partition Installation

- DB2.AUTOSTART
Specifies whether or not to automatically start the DB2 instance each time the system is rebooted.
Note: By default, the DB2 instance starts automatically unless this parameter is set to *NO*.
- ADMIN.USERID
Specifies the user account that the DB2 Administration Server will use to log on to the system. For more information on this user account, see “Before You Begin” on page 43.
- ADMIN.PASSWORD
Specifies the password for the DB2 Administration Server.

Keywords for a Client Installation

- AUTOSTART_CCA

Specifies whether or not to automatically start the Client Configuration Assistant each time the system is rebooted.

Note: By default, the Client Configuration Assistant starts automatically unless this parameter is set to *NO*.

Exit the file. If you have made any changes, save the file under a new file name to preserve the original sample response file.

If you are editing a response file directly from the CD-ROM, you must store the renamed response file on another drive.

Step 3. Run the Setup Program from the Workstation

To perform the installation:

- 1** At the workstation where DB2 will be installed, log on with a user account that meets the requirements for installing DB2. For more information on this username, see “Before You Begin” on page 43.
- 2** If performing a remote install, connect to the shared directory of the remote server hard disk, or to the CD-ROM drive, using the following command:

```
net use e: \\machine\directory
```

where:

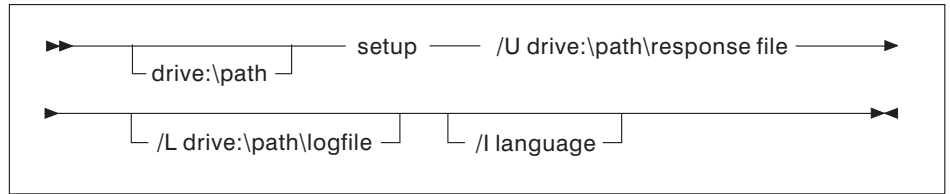
- e: represents the shared directory on the server's hard disk or CD-ROM drive.
- machine is the machine name of the remote server.
- directory is the share name of the remote server or CD-ROM drive.

For example, enter:

```
net use e: \\codeserv\db2nt
```

Based on the example set up in previous steps, this points to q:\db2prods on the remote server.

- 3** Run the setup program:
 - a** Click on **Start** and select the **Run** option.
 - b** In the Open field, enter the path to the setup program. The syntax of the setup command is as follows:



where:

- /U** Specifies the fully qualified response file name. If you changed and renamed the sample response file provided, make sure that this parameter matches the new name.
- /L** Specifies the fully qualified log file name, where setup information and any errors occurring during setup are logged.
If you do not specify the log file's name, DB2 names it `db2.log`, and stores it in a directory called `db2log`, on the drive on which your operating system is installed.
- /I** Specifies the two-character country code that represents your language (for example, EN for English). See Appendix C, "National Language Support (NLS)" on page 231 for more information.
If you do not specify the language, setup will determine the system language, and launch the appropriate DB2 install for that language.

For example, to install an instance-owning database partition server, using a custom response file, enter:

```
x:\setup /U d:\temp\eeentio.rsp
```

- c** Click on **OK** to start the setup program.
 - d** The installation proceeds without further action on your part.
- 4** Check the messages in the log file when the installation finishes.
 - 5** Following installation, you must reboot the workstation before using DB2.
- Note:** You must reboot the workstation before installing additional nodes.

Installing DB2 Remotely Using SMS

With Microsoft Systems Management Server (SMS), you can install DB2 across a network onto remote Windows NT systems, setting up the installation from a central location.

You can also use SMS to *remove* all DB2 products from a target workstation. Simply follow the instructions below, but when creating the SMS package, select **Uninstall DB2 Products** instead of selecting the packages to be installed.

To install DB2 using SMS, perform the steps in the following sections:

- “Step 1. Make DB2 Files Available for Installation”
- “Step 2. Create a Response File for Unattended Installation”
- “Step 3. Import the DB2 Install Files into SMS” on page 200
- “Step 4. Distribute the DB2 Installation Package Using SMS” on page 200
- “Step 5. Run the Installation on the SMS Client” on page 201.

Before You Begin

Read and perform the steps in the “Before You Begin” on page 43 before you begin the installation to ensure that you have the required items and information you will need.

SMS Requirements

You must already have SMS installed and configured on your network. Refer to *Microsoft's Systems Management Server for Windows NT Administrator's Guide* for information on how to:

- Set up SMS (including setting up primary and secondary sites).
- Add clients to the SMS system.
- Set up inventory collection for clients.

To create an SMS package that copies the required files directly from the CD-ROM, insert the appropriate CD-ROM into the drive and go to “Step 2. Create a Response File for Unattended Installation.”

Step 1. Make DB2 Files Available for Installation

To complete this Step, follow the instructions in “Step 1. Make DB2 Files Available for Installation” on page 188.

Step 2. Create a Response File for Unattended Installation

To complete this Step, follow the instructions in “Step 2. Create a Response File for Unattended Installation” on page 189.

Step 3. Import the DB2 Install Files into SMS

To import the DB2 install files into SMS:

- 1 Insert the appropriate CD-ROM into the drive.
- 2 Click on **Start** and select **Programs->Systems Management Server-> SMS Administrator**. The SMS Administrator window opens.
- 3 Open the Package Properties window. The first time you start the SMS Administrator program you will see a dialog box prompting you to open an SMS Window. You can select the Packages option from there. Otherwise, select **File->New** from the menu bar.
- 4 Click on the **Import** push button. The **File Browser** opens.
- 5 In the **File Name** field, select the db2.pdf file.

The server db2.pdf file for DB2 Universal Database is located in the

x:\db2\common directory (for DB2 Server)

x:\db2\winnt95\common directory (for DB2 CAE or DB2 SDK)

where x: designates your CD-ROM drive, if you did not set up a remote server as described in on page 188.

Step 4. Distribute the DB2 Installation Package Using SMS

In this step, you create an SMS package and send it to the SMS client where you want to install DB2.

To create the SMS package:

- 1 From the Package Properties window, click on the **Workstations** push button. The Setup Package For Workstations window opens, showing a list of predefined DB2 installation packages that you can use.
- 2 In the **Source Directory** field, type the name of the parent directory where you put the copied DB2 files.
- 3 Select the name of the product to install from the **Workstation Command Lines** window.
- 4 If you changed and renamed the sample response file, click on the **Properties** push button and change the value of the **Command line** parameter to match the new response file name.

If you do not specify the language, the setup program will determine the system language, and launch the appropriate DB2 install for that language. If you want to install a different language, use the /l parameter to specify the two-character code

that represents your language (for example, EN for English). See Appendix C, “National Language Support (NLS)” on page 231 for more information.

- 5** Click on the **Close** push button.
- 6** Click on **OK** to close the opened windows.
The Packages window shows the name of the new SMS package.
- 7** Specify where to run the package:
 - a** Open the Sites window and the Packages window.
 - b** In the Packages window, select the appropriate package and drag it onto the target client in the Sites window. The Job Details window opens.
 - c** Select the **Run Workstation Command** check box and select the installation package that you want to use for the Run Command on Workstation job. For more information about completing the Job Details window, refer to *Microsoft's Systems Management Server for Windows NT Administrator's Guide*.
 - d** When the job specifications are complete, click on **OK** to close the opened windows.

The job is created and sent to the SMS client.

Step 5. Run the Installation on the SMS Client

Now that the remote installation package has been prepared and sent, the SMS client on the target machine can run the installation.

- 1** On the SMS client where DB2 will be installed, logon as a username that meets the requirements for installing DB2. See “Before You Begin” on page 43 for more information.
- 2** Click on **Start** and select **Systems Management Client-> Package Command Manager**. The Package Command Manager window opens.
- 3** When packages are sent from the SMS server, they are listed in the window. Select the DB2 package, then click on the **Execute** push button. The installation runs automatically.
- 4** Following installation, you must reboot the SMS client before using DB2.
Note: If you specified REBOOT=**YES** in your response file the SMS client will reboot automatically.

Chapter 21. Accessing DB2 Universal Database Servers from Host and AS/400 Applications

Host and AS/400 applications can access DB2 Universal Database data using DB2 Workgroup Edition, DB2 Enterprise Edition and DB2 Extended Enterprise Edition. The following are examples of the use of this access:

Host or AS/400 applications

If you are migrating data from your host or AS/400 database to a DB2 Universal Database server, you can continue to use your existing host or AS/400 applications by having them access the data from DB2 Universal Database. This can allow for staged migration from the host or the AS/400 system.

Host or AS/400 Applications leveraging DB2 Extended Enterprise Edition

Your host or AS/400 application can use the parallel processing power of DB2 Universal Database for CPU intensive queries.

Access Distributed Data

Your host or AS/400 application can access distributed data stored in departmental DB2 Universal Database servers.

Supported Clients

The following database products can access DB2 Universal Database servers:

- DB2 for MVS/ESA Version 3.1 (or higher)

For information on setting up the connection from DB2 for MVS/ESA to a DB2 Universal Database server, see the instructions in "Configuration Steps for DB2 Universal Database Server" on page 204.

- DB2 for OS/390 Version 5 (or higher)

For information on setting up the connection from DB2 for OS/390 to a DB2 Universal Database server, see the instructions in "Configuration Steps for DB2 Universal Database Server" on page 204.

- DB2 for AS/400 Version 3.1 (or higher)

For information on setting up the connection from DB2 for AS/400 to a DB2 Universal Database server, refer to the *DB2 Connectivity Supplement*.

- DB2 for VM & VSE (SQL/DS) Version 3.3 and Version 3.4 (or higher)

For information on setting up the connection from DB2 for VM & VSE to a DB2 Universal Database server, refer to the *DB2 Connectivity Supplement*.

For information about other IBM and non-IBM products accessing DB2 Universal Database servers, contact the software support for those products.

PTFs Required

The following PTFs are required:

DB2 for MVS/ESA Version 3: UN73393

DB2 for MVS/ESA Version 4: UN75959

DB2 for OS/390 Version 5: PQ07537

DB2 for VM/ESA Version 5: VM60922; VM61072

OS/400: See Table 11.

Table 11. OS/400 PTFs Required

OS/400 PTF	OS/400 V2R3 PTF	OS/400 V3R0.5 PTF	OS/400 V3R1 PTF
SA42461	SF23100	SF23950	SF23270
SA43024	SF23205	SF23950	SF23277
3989448	SF23101	SF23994	SF23271
3993315	SF23722	SF23988	SF23721
3994088	SF23987	SF23986	SF23985
3994297	SF23990	SF23989	SF23960

Configuration Steps for DB2 Universal Database Server

This section describes the steps and conditions required to configure a DB2 Universal Database server to accept connection from host and AS/400 database clients.

Before you begin, you must determine whether your connection will use the APPC communication protocol, or the TCP/IP communication protocol, or both.

Platform	Supported Protocols
AIX	TCP/IP, APPC, APPC Two Phase Commit

SCO	TCP/IP
SINIX	TCP/IP
Solaris	TCP/IP, APPC
OS/2	TCP/IP, APPC, APPC Two Phase Commit
Windows NT	TCP/IP, APPC, APPC Two Phase Commit

Notes:

1. The protocol you choose may depend on the host or AS/400 database client version:

- APPC is supported by all host or AS/400 database client versions
- TCP/IP is supported on the following host or AS/400 database client versions
 - DB2 for OS/390 Version 5 or higher
 - DB2 for AS/400 Version 4 Release 2 or higher
 - DB2 for VM Version 6 or higher.

2. Considerations for Multisite Update (Two Phase Commit)

If your host or AS/400 application required Multisite Update support (two-phase commit), you should note the following:

APPC Connections (SNA)

DB2 Universal Database Enterprise Edition for OS/2, AIX, and Windows NT and DB2 Extended Enterprise Edition for AIX, and Windows NT provide support for SNA two phase commit with host and AS/400 database clients. The exception is AS/400 clients which **cannot** have Multisite Update support with DB2 for AIX Enterprise Edition or Extended Enterprise Edition.

TCP/IP Connections

Multisite Update is not supported with any host or AS/400 database clients. Remote Unit of Work (RUW) is supported (one phase commit)

3. To use APPC on AIX, the optional communication support component for SNA (db2_05_00.cs.sna) must also be installed.
4. To use APPC on Solaris, the optional communication support component for SNA (db2cssna) must also be installed.

Configuring DB2 Universal Database Servers for Host or AS/400 Client Access

This section gives an overview of the steps necessary to configure DB2 Universal Database to allow connectivity from host or AS/400 database clients. The example shows how to configure a connection from a DB2 for MVS/ESA or DB2 for OS/390 client to a DB2 Universal Database server:

- 1 If you use APPC, ensure VTAM is installed and operational on the host.

Refer to the *DB2 Connect Enterprise Edition Quick Beginnings* and the *DB2 for OS/390 V5 Installation Guide* manual for information on configuring the DB2 for MVS/ESA or DB2 for OS/390 AR.

- 2 Set up the DB2 server communications, if required. (Typically, DB2 server communications are set up as part of the DB2 installation. However, complete details are provided in Chapter 19, "Setting up Communications on the Server Using the Command Line Processor" on page 171. If you want to use Multisite Update, refer to the chapter entitled *Enabling Multisite Update* in the *DB2 Connect Enterprise Edition Quick Beginnings* manual.) DB2 Connect is not required for host or AS/400 database client connections to DB2 Universal Database; however, the steps to enable Multisite Update are common with DB2 Connect.
- 3 Update the tables at the DB2 for MVS/ESA or DB2 for OS/390 host.
Refer to the *DB2 Connectivity Supplement* for more information.
- 4 Stop and restart the DB2 Universal Database instance named in DB2SERVICETPINSTANCE (that is, the instance in which you updated the database configuration parameter(s) *tpname* and/or *svcename*. Enter the following commands:

```
db2stop  
db2start
```
- 5 Test the host-server DRDA connection by logging onto TSO and using DB2I/SPUFI.

Using the DB2 Universal Database Server from Host or AS/400 Clients

Connections from host or AS/400 database clients are treated like any other connection to the DB2 server to ensure consistency in the way the maximum number of concurrent connections to a server is measured, from host, AS/400, and Universal Database clients.

The *Administration Guide* summarizes the CCSIDs that a host or AS/400 database client must use in order to connect to a DB2 Universal Database server.

When using APPC, a host or AS/400 database client connects to the DB2 Universal Database server by specifying the appropriate Transaction Program Name (TPN) defined at the DB2 Universal Database server. The TPN can correspond to the *tpname* parameter value in the instance's database manager configuration file. The TPN used by the host or AS/400 database client can also be the service transaction program **x'07'6DB**. When using this TPN and if multiple DB2 Universal Database instances exist at the server, the instance which will process the **x'07'6DB** program is specified by the

different local LU name. The instance that handles the TPN **x'07'6DB** program is specified by the DB2 registry value DB2SERVICETPINSTANCE.

Authentication

If you choose APPC as your communication protocol, the communications subsystem may restrict the types of database manager configuration authentication settings you can use at the DB2 Universal Database server. Not all communications subsystems expose the application requester's password to the application server when security (program) is used. When this is the case, the database manager configuration authentication must not be set to SERVER. If you choose APPC as your communications protocol, you may be restricted in the types of database manager configuration authentication setting you can use at the DB2 Universal Database Server. If you have the SNA Syncpoint Manager configured, you can use any available authentication (SERVER, CLIENT, DCS). If you do not have the SNA Syncpoint Manager configured then you are limited to authentication CLIENT or DCS.

Connections via TCP/IP can use any of the available database manager authentication options (SERVER, CLIENT, DCS).

To overcome the restriction that prevents you from using authentication SERVER, you can set the database manager authentication to DCS. This allows host or AS/400 database client connections that have been authenticated by the communication subsystem to succeed. However, with this setting the DB2 Universal Database will act as if authentication SERVER is being used for remote DB2 Universal Database client connections.

Troubleshooting

The DB2 DRDA Trace utility (**db2drdat**) is provided for tracing the data flow between a host or AS/400 database client and the DB2 Universal Database Server. For more information on setting up this trace, refer to the *Troubleshooting Guide*.

DRDA Functions Supported

DRDA functions are categorized as required or optional. Table 12 on page 208 identifies which functions are implemented in the AS on the DB2 Universal Database server. The subsequent table lists the supported bind options.

Table 12. Supported DRDA Functions

Description	Required (R) Optional (O)	Supported	Not Supported
DRDA level 1 required function	R	X*	
Rebind	O	X	
Describe user privileges	O		X
Describe RDB table	O		X
Interrupt RDB request	O		X

Note: * Certain required functions are not supported.

Bind Options Supported by the DB2 DRDA Application Server

Table 13 (Page 1 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Supported	Not Supported	DB2 for IMS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
Package Version Name	<u>Null</u> Any other value	X	X	VERSION			VERSION
Bind Existence Checking	<u>Object existence optional</u> Object existence required	X	X	VALIDATE (RUN) ^b VALIDATE (BIND) ^b	<u>NOEXIST</u> EXIST	GENLVL(10, 11-40) GENLVL(00-09)	VALIDATE <u>RUN</u> VALIDATE BIND
Package Replacement Option	<u>Replacement allowed</u> Replacement not allowed	X	X	ACTION (REPLACE) ACTION(ADD)	<u>REPLACE</u> NEW	REPLACE(*YES) REPLACE(*NO)	ACTION <u>REPLACE</u> ACTION ADD
Package Authorization Option	<u>Keep authorizations</u> Revoke authorizations	X	X		<u>KEEP</u> REVOKE		RETAIN <u>YES</u> RETAIN NO
Statement String Delimiter (Note 2)	Apostrophe Double quote	X	X	<u>APOSTSQL</u> QUOTESQL	<u>SQLPOST</u> SQLQUOTE	OPTION([...] *APOSTSQL) (Note 3) OPTION([...] *QUOTESQL) (Note 4)	STRDEL APOSTROPHE STRDEL QUOTE
Statement Decimal Delimiter (Note 5)	Period Comma	X	X	<u>PERIOD</u> COMMA	<u>PERIOD</u> COMMA	OPTION([...] *PERIOD) or OPTION([...] *SYSVAL) (Note 6)	DECDEL PERIOD DECDEL COMMA
Date Format (Note 7)	<u>ISO</u> USA EUR JIS	X	X	DATE(ISO) (Note 8) DATE(USA) DATE(EUR) DATE(JIS)	DATE(ISO) DATE(USA) DATE(EUR) DATE(JIS)	DATFMT(*ISO) (Note 8) DATFMT(*USA) DATFMT(*EUR) DATFMT(*JIS)	DATETIME <u>ISO</u> (Note 9) DATETIME USA DATETIME EUR DATETIME JIS
Time Format (Note 7)	<u>ISO</u> USA EUR JIS	X	X	TIME(ISO) (Note 8) TIME(USA) TIME(EUR) TIME(JIS)	TIME(ISO) TIME(USA) TIME(EUR) TIME(JIS)	TIMFMT(*ISO) (Note 8) TIMFMT(*USA) TIMFMT(*EUR) TIMFMT(*JIS)	DATETIME <u>ISO</u> (Note 9) DATETIME USA DATETIME EUR DATETIME JIS
Package Isolation Level (Note 10)	Repeatable read Read Stability (All)	X	X	ISOLATION(RR) ^b ISOLATION(RS)	ISOLATION(RR) ISOLATION(RS)	COMMIT(*ALL)	ISOLATION RR ISOLATION RS

Table 13 (Page 2 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Supported	Not Supported (Note 11)	DB2 for iMVS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
	Cursor stability	X		ISOLATION(CS) ^b	ISOLATION(CS)	COMMIT(*CS)	ISOLATION CS
	Uncommitted Read (Change)	X			ISOLATION(UR)	COMMIT(*CHG)	ISOLATION UR
	No commit		X (Note 11)			COMMIT(*NONE)	ISOLATION NC
Bind Creation Control	No errors allowed	X		SQLERROR (NOPACKAGE) ^b	NOCHECK	OPTION([...] *GEN) GENLVL(00-09, 10, 11-20)	SQLERROR NOPACKAGE
	Check only	X			CHECK	OPTION([...] *NOGEN)	SQLERROR CHECK
	Errors allowed		X	SQLERROR (CONTINUE) ^b	ERROR	OPTION([...] *GEN) GENLVL(21-40)	SQLERROR CONTINUE
Bind Explain Option	No SQL statements	X		EXPLAIN(NO) ^b	EXPLAIN(NO)		EXPLAIN NO
	All explainable SQL statements		X	EXPLAIN(YES) ^b	EXPLAIN(YES)		EXPLAIN YES
Package Owner Identifier	<Authorization ID>	X		OWNER ^b	OWNER		OWNER
	Any other value		X				
RDB Release Option	Release at commit	X		RELEASE (COMMIT) ^b	RELEASE (COMMIT)		RELEASE COMMIT
	Release at conversation deallocation		X	RELEASE (DEALLOCATE) ^b	RELEASE (DEALLOCATE)		RELEASE DEALLOCATE
Default RDB Collection ID	<Authorization ID>	X		QUALIFIER ^b	QUALIFIER	DFTDRBCOL	QUALIFIER
	Any other value		X				
Title (Package Description)	Any value (ignored by DB2)	X			LABEL	TEXT	TEXT
Query Block Protocol Control	Fixed row	X		CURRENTDATA (YES) ^b	SBLOCK	ALWBLK(*READ)	BLOCKING UNAMBIG
	Limited block	X		CURRENTDATA (NO) ^b	BLOCK	ALWBLK (*ALLREAD)	BLOCKING ALL
	Forced fixed row	X			NOBLOCK	ALWBLK(*NONE)	BLOCKING NO
Package Default Char. Subtype	Use system default	X					CHARSUB DEFAULT
If Default CCSID is SBCS	BIT		X		CHARSUB(BIT)		CHARSUB BIT
If Default CCSID is MBCS	SBCS	X			CHARSUB(SBCS)		CHARSUB SBCS
If Default CCSID is UCS	MBCS	X			CHARSUB(MBCS)		CHARSUB MBCS
If Default CCSID is EBCS	BIT	X			CHARSUB(BIT)		CHARSUB BIT

Table 13 (Page 3 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Sup-ported	Not Sup-ported	DB2 for IMS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
If Default CCSID is MBCS	SBCS		X		CHARSUB(SBCS)		CHARSUB SBCS
If Default CCSID is MBCS	MBCS	X			CHARSUB(MBCS)		CHARSUB MBCS
	Any other value		X				
Package Default CCSID	<u>Value specified when DB2 database was created</u>	X			CCSID(SBCS) CCSID(GRAPHIC) CCSID(MIXED)		CCSID(SBCS) CCSID(MIXED)
	Any other value		X				
Decimal Precision (Note 12)	31	X		DEC(31)			DEC 31
	Any other value		X	DEC(15)			DEC 15
Replaced Package Version Name	<u>Null</u>	X		REPLVERD			REPLVER
	Any other value		X				
Generic Bind Option	<u>Null</u>		X				GENERIC
	Any other value		X				
Package Authorization Rule	<u>Requester</u>	X					DYNAMICRULES RUN
	Owner		X				DYNAMICRULES BIND
	Creator of the user-defined function and stored procedure		X				DYNAMICRULES DEFINE
	Invoker of the user-defined function and stored procedure		X				DYNAMICRULES INVOKE
Degree of Parallelism	<u>1</u>		X				DEGREE 1
	n		X				DEGREE n
	ANY		X				DEGREE ANY

Table 13 (Page 4 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Supported	Not Supported	DB2 for MVS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
<p>Note:</p> <p>(*) Default values are in bold.</p> <p>(1) Most are precompile options. Bind options are indicated by b.</p> <p>(2) Defaults to what the target database supports. For DB2 the default is apostrophe.</p> <p>(3) Default for non-COBOL applications.</p> <p>(4) Default for COBOL applications.</p> <p>(5) Defaults to what the target database supports. For DB2 the default is period.</p> <p>(6) Depending on the installation, "SYSVAL" is equivalent to "PERIOD" or "COMMA".</p> <p>(7) Date and time formats must be the same for the DB2 DRDA AS.</p> <p>(8) Default is dependent on the installation.</p> <p>(9) Format applies to both date and time. If not specified, it defaults based on the country code. This default is mapped to ISO in DRDA flow.</p> <p>(10) Package isolation level has no default because an explicit value is always present in the DRDA datastream.</p> <p>(11) The isolation level will be escalated to Uncommitted Read (Change).</p> <p>(12) Defaults to what the target database supports. For DB2 the default is 31.</p> <p>(13) All variables will default to 1.</p>							

Special Considerations for DB2 for VM (SQL/DS)

Additional steps are needed to ensure the following DB2 for VM utilities work properly against a DB2 database through the Application Server.

- SQLDBSU
 - 1 Make sure a PTF for DB2 for VM APAR PN69073 is installed on your DB2 for VM system, either PTF UN91171 or PTF UN91172. (At the time of publication of this manual, there were no PTFs for DB2 for VM V4 or V5.)
 - 2 Set up dummy tables in your DB2 database by executing the **sqldbsu** utility provided with DB2 as `sqldbsu database_name`.
 - 3 Bind SQLDBSU from DB2 for VM. Refer to the “Using a DRDA Environment” section in the *SQL/DS System Administration for IBM VM Systems* manual for details. (You can skip the step on creating and populating the SQLDBA.DBSOPTIONS table because this is done by the **sqldbsu** utility in the previous step.)
- ISQL
 - 1 Complete the steps described above for SQLDBSU.
 - 2 Set up dummy tables in your DB2 database by executing the **isql** utility provided with DB2 as `isql database_name`.
 - 3 Bind ISQL from DB2 for VM. Refer to the “Using a DRDA Environment” in the *SQL/DS System Administration for IBM VM Systems* manual for details.

Note: The **sqldbsu** and **isql** utilities are in DB2PATH\misc, for example:

```
c:\SQLLIB\misc\
```

If you installed DB2 on drive C, using the default directory `sqllib`, then no special setup is required for RXSQL; refer to the *SQL/DS Procedures Language Interface Installation* manual for details.

Security and Auditability

Under APPC, DB2 system security (authentication CLIENT, SERVER, or DCS) must be used with a APPC security SAME or PROGRAM. When these combinations are used, the user ID and password sent by the host or AS/400 is used to CONNECT to the requested database. APPC security level NONE is allowed only with DCE authentication. In this case, the encrypted DCE ticket is flown as part of the CONNECT attempt.

Under TCP/IP, all security information is flown in the CONNECT attempt.

User ID translation is not supported by DB2 Universal Database.

Configuration Considerations

Data access by host and AS/400 applications are tuned at the DB2 Universal Database server using the DB2 database manager configuration parameters. One parameter, DRDA Heap Size, is specifically for host and AS/400 database client connections. You may need to change the setting for some parameters because of the additional resources required for the Application Server.

DRDA Heap Size (drda_heap_sz)

DRDA heap size specifies the amount of memory, in segments, that is allocated for use by the DB2 Universal Database server for host and AS/400 connections.

Refer to the *Administration Guide* for more information about database manager configuration.

Part 7. Appendixes

Appendix A. How the DB2 Library Is Structured

The DB2 Universal Database library consists of SmartGuides, online help, and books. This section describes the information that is provided, and how to access it.

To access product information online, you can use the Information Center. You can view task information, DB2 books, troubleshooting information, sample programs, and DB2 information on the Web. See "Information Center" on page 227 for details.

SmartGuides

SmartGuides help you complete some administration tasks by taking you through each task one step at a time. SmartGuides are available through the Control Center. The following table lists the SmartGuides.

Note: Not all SmartGuides are available for the partitioned database environment.

SmartGuide	Helps you to...	How to Access...
<i>Add Database</i>	Catalog a database on a client workstation.	From the Client Configuration Assistant, click on Add .
<i>Create Database</i>	Create a database, and perform some basic configuration tasks.	From the Control Center, click with the right mouse button on the Databases icon and select Create->New .
<i>Performance Configuration</i>	Tune the performance of a database by updating configuration parameters to match your business requirements.	From the Control Center, click with the right mouse button on the database you want to tune and select Configure performance .
<i>Backup Database</i>	Determine, create, and schedule a backup plan.	From the Control Center, click with the right mouse button on the database you want to backup and select Backup->Database using SmartGuide .
<i>Restore Database</i>	Recover a database after a failure. It helps you understand which backup to use, and which logs to replay.	From the Control Center, click with the right mouse button on the database you want to restore and select Restore->Database using SmartGuide .

SmartGuide	Helps you to...	How to Access...
<i>Create Table</i>	Select basic data types, and create a primary key for the table.	From the Control Center, click with the right mouse button on the Tables icon and select Create->Table using SmartGuide .
<i>Create Table Space</i>	Create a new table space.	From the Control Center, click with the right mouse button on the Table spaces icon and select Create->Table space using SmartGuide .

Online Help

Online help is available with all DB2 components. The following table describes the various types of help. You can also access DB2 information through the Information Center. For information see "Information Center" on page 227.

Type of Help	Contents	How to Access...
<i>Command Help</i>	Explains the syntax of commands in the command line processor.	From the command line processor in interactive mode, enter: <i>? command</i> where <i>command</i> is a keyword or the entire command. For example, ? catalog displays help for all the CATALOG commands, while ? catalog database displays help for the CATALOG DATABASE command.
<i>Control Center Help</i>	Explains the tasks you can perform in a window or notebook. The help includes prerequisite information you need to know, and describes how to use the window or notebook controls.	From a window or notebook, click on the Help push button or press the F1 key.

Type of Help	Contents	How to Access...
<i>Message Help</i>	Describes the cause of a message, and any action you should take.	<p>From the command line processor in interactive mode, enter:</p> <p>? XXXnnnnn</p> <p>where <i>XXXnnnnn</i> is a valid message identifier.</p> <p>For example, ? SQL30081 displays help about the SQL30081 message.</p> <p>To view message help one screen at a time, enter:</p> <p>? XXXnnnnn more</p> <p>To save message help in a file, enter:</p> <p>? XXXnnnnn > filename.ext</p> <p>where <i>filename.ext</i> is the file where you want to save the message help.</p>
<i>SQL Help</i>	Explains the syntax of SQL statements.	<p>From the command line processor in interactive mode, enter:</p> <p>help statement</p> <p>where <i>statement</i> is an SQL statement.</p> <p>For example, help SELECT displays help about the SELECT statement.</p>
<i>SQLSTATE Help</i>	Explains SQL states and class codes.	<p>From the command line processor in interactive mode, enter:</p> <p>? sqlstate or ? class-code</p> <p>where <i>sqlstate</i> is a valid five-digit SQL state and the <i>class-code</i> is first two digits of the SQL state.</p> <p>For example, ? 08003 displays help for the 08003 SQL state, while ? 08 displays help for the 08 class code.</p>

DB2 Books

The table in this section lists the DB2 books. They are divided into two groups:

Cross-platform books These books contain the common DB2 information for UNIX-based and Intel-based platforms.

Platform-specific books These books are for DB2 on a specific platform. For example, for DB2 on OS/2, on Windows NT, and on the UNIX-based platforms, there are separate *Quick Beginnings* books.

Most books are available in HTML and PostScript format, and in hardcopy that you can order from IBM. The exceptions are noted in the table.

If you want to read the English version of the books, they are always provided in the directory that contains the English documentation.

You can obtain DB2 books and access information in a variety of different ways:

View	See "Viewing Online Books" on page 224.
Search	See "Searching Online Books" on page 225.
Print	See "Printing the PostScript Books" on page 225.
Order	See "Ordering the Printed DB2 Books" on page 226.

Book Name	Book Description	Form Number File Name
Cross-Platform Books		
<i>Administration Getting Started</i>	Introduces basic DB2 database administration concepts and tasks, and walks you through the primary administrative tasks.	S10J-8154 db2k0x50
<i>Administration Guide</i>	Contains information required to design, implement, and maintain a database to be accessed either locally or in a client/server environment.	S10J-8157 db2d0x51
<i>API Reference</i>	Describes the DB2 application programming interfaces (APIs) and data structures you can use to manage your databases. Explains how to call APIs from your applications.	S10J-8167 db2b0x51
<i>CLI Guide and Reference</i>	Explains how to develop applications that access DB2 databases using the DB2 Call Level Interface, a callable SQL interface that is compatible with the Microsoft ODBC specification.	S10J-8159 db2l0x50
<i>Command Reference</i>	Explains how to use the command line processor, and describes the DB2 commands you can use to manage your database.	S10J-8166 db2n0x51

Book Name	Book Description	Form Number File Name
<i>DB2 Connect Enterprise Edition Quick Beginnings</i>	Provides planning, migrating, installing, configuring, and using information for DB2 Connect Enterprise Edition. Also contains installation and setup information for all supported clients.	S10J-7888 db2cyx51
<i>DB2 Connect Personal Edition Quick Beginnings</i>	Provides planning, installing, configuring, and using information for DB2 Connect Personal Edition.	S10J-8162 db2c1x51
<i>DB2 Connect User's Guide</i>	Provides concepts, programming and general using information about the DB2 Connect products.	S10J-8163 db2c0x51
<i>DB2 Connectivity Supplement</i>	Provides setup and reference information for customers who want to use DB2 for AS/400, DB2 for OS/390, DB2 for MVS, or DB2 for VM as DRDA Application Requesters with DB2 Universal Database servers, and customers who want to use DRDA Application Servers with DB2 Connect (formerly DDCS) application requesters. Note: Available in HTML and PostScript formats only.	No form number db2h1x51
<i>Embedded SQL Programming Guide</i>	Explains how to develop applications that access DB2 databases using embedded SQL, and includes discussions about programming techniques and performance considerations.	S10J-8158 db2a0x50
<i>Glossary</i>	Provides a comprehensive list of all DB2 terms and definitions. Note: Available in HTML format only.	No form number db2t0x50
<i>Installing and Configuring DB2 Clients</i>	Provides installation and setup information for all DB2 Client Application Enablers and DB2 Software Developer's Kits. Note: Available in HTML and PostScript formats only.	No form number db2iyx51
<i>Master Index</i>	Contains a cross reference to the major topics covered in the DB2 library. Note: Available in PostScript format and hardcopy only.	S10J-8170 db2w0x50
<i>Messages Reference</i>	Lists messages and codes issued by DB2, and describes the actions you should take.	S10J-8168 db2m0x51
<i>DB2 Replication Guide and Reference</i>	Provides planning, configuring, administering, and using information for the IBM Replication tools supplied with DB2.	S95H-0999 db2e0x52
<i>Road Map to DB2 Programming</i>	Introduces the different ways your applications can access DB2, describes key DB2 features you can use in your applications, and points to detailed sources of information for DB2 programming.	S10J-8155 db2u0x50

Book Name	Book Description	Form Number File Name
<i>SQL Getting Started</i>	Introduces SQL concepts, and provides examples for many constructs and tasks.	S10J-8156 db2y0x50
<i>SQL Reference</i>	Describes SQL syntax, semantics, and the rules of the language. Also includes information about release-to-release incompatibilities, product limits, and catalog views.	S10J-8165 db2s0x51
<i>System Monitor Guide and Reference</i>	Describes how to collect different kinds of information about your database and the database manager. Explains how you can use the information to understand database activity, improve performance, and determine the cause of problems.	S10J-8164 db2f0x50
<i>Troubleshooting Guide</i>	Helps you determine the source of errors, recover from problems, and use diagnostic tools in consultation with DB2 Customer Service.	S10J-8169 db2p0x50
<i>What's New</i>	Describes the new features, functions, and enhancements in DB2 Universal Database, Version 5.2, including information about Java-based tools.	S04L-6230 db2q0x51
Platform-Specific Books		
<i>Building Applications for UNIX Environments</i>	Provides environment setup information and step-by-step instructions to compile, link, and run DB2 applications on a UNIX system.	S10J-8161 db2axx51
<i>Building Applications for Windows and OS/2 Environments</i>	Provides environment setup information and step-by-step instructions to compile, link, and run DB2 applications on a Windows or OS/2 system.	S10J-8160 db2a1x50
<i>DB2 Personal Edition Quick Beginnings</i>	Provides planning, installing, migrating, configuring, and using information for DB2 Universal Database Personal Edition on OS/2, Windows 95, and the Windows NT operating systems.	S10J-8150 db2i1x50
<i>DB2 SDK for Macintosh Building Your Applications</i>	Provides environment setup information and step-by-step instructions to compile, link, and run DB2 applications on a Macintosh system. Note: Available in PostScript format and hardcopy for DB2 Version 2.1.2 only.	S50H-0528 sqla7x02
<i>DB2 SDK for SCO OpenServer Building Your Applications</i>	Provides environment setup information and step-by-step instructions to compile, link, and run DB2 applications on a SCO OpenServer system. Note: Available for DB2 Version 2.1.2 only.	S89H-3242 sqla9x02
<i>DB2 SDK for SINIX Building Your Applications</i>	Provides environment setup information and step-by-step instructions to compile, link, and run DB2 applications on a SINIX system. Note: Available in PostScript format and hardcopy for DB2 Version 2.1.2 only.	S50H-0530 sqla8x00

Book Name	Book Description	Form Number File Name
<i>Quick Beginnings for OS/2</i>	Provides planning, installing, migrating, configuring, and using information for DB2 Universal Database on OS/2. Also contains installing and setup information for all supported clients.	S10J-8147 db2i2x50
<i>Quick Beginnings for UNIX</i>	Provides planning, installing, configuring, migrating, and using information for DB2 Universal Database on UNIX-based platforms. Also contains installing and setup information for all supported clients.	S10J-8148 db2ixx51
<i>Quick Beginnings for Windows NT</i>	Provides planning, installing, configuring, migrating, and using information for DB2 Universal Database on the Windows NT operating system. Also contains installing and setup information for all supported clients.	S10J-8149 db2i6x50
<i>DB2 Extended Enterprise Edition for UNIX Quick Beginnings</i>	Provides planning, installing, configuring, and using information for DB2 Universal Database Extended Enterprise Edition for UNIX. This book supercedes the <i>DB2 Extended Enterprise Edition Quick Beginnings for AIX</i> book, and is suitable for use with all versions of DB2 Extended Enterprise Edition that run on UNIX-based platforms.	S99H-8314 db2v3x51
<i>DB2 Extended Enterprise Edition for Windows NT Quick Beginnings</i>	Provides planning, installing, configuring, and using information for DB2 Universal Database Extended Enterprise Edition for Windows NT.	S09L-6713 db2v6x51

Notes:

1. The character in the sixth position of the file name indicates the language of a book. For example, the file name db2d0e50 indicates that the *Administration Guide* is in English. The following letters are used in the file names to indicate the language of a book:

Language	Identifier	Language	Identifier
Brazilian Portuguese	B	Japanese	J
Bulgarian	U	Korean	K
Czech	X	Norwegian	N
Danish	D	Polish	P
English	E	Russian	R
Finnish	Y	Simp. Chinese	C
French	F	Slovenia	L
German	G	Spanish	Z
Greek	A	Swedish	S
Hungarian	H	Trad. Chinese	T
Italian	I	Turkish	M

2. For late breaking information that could not be included in the DB2 books:
 - On UNIX-based platforms, see the `Release.Notes` file. This file is located in the `DB2DIR/Readme/%L` directory, where `%L` is the locale name and `DB2DIR` is:

- /usr/lpp/db2_05_00 on AIX
- /opt/IBMDB2/V5.0 on HP-UX, Solaris, SCO UnixWare 7, and SGI.
- On other platforms, see the RELEASE.TXT file. This file is located in the directory where the product is installed.

Viewing Online Books

The manuals included with this product are in Hypertext Markup Language (HTML) softcopy format. Softcopy format enables you to search or browse the information, and provides hypertext links to related information. It also makes it easier to share the library across your site.

You can use any HTML Version 3.2-compliant browser to view the online books.

To view online books:

- If you are running DB2 administration tools, use the Information Center. See “Information Center” on page 227 for details.
- Use the open file function of your Web browser. The page you open contains descriptions of and links to DB2 books:
 - On UNIX-based platforms, open the following page:


```
file:/INSTHOME/sqllib/doc/%L/html/index.htm
```

 where %L is the locale name.
 - On other platforms, open the following page:


```
sqllib\doc\html\index.htm
```

The path is located on the drive where DB2 is installed.

You can also open the page by double-clicking on the **DB2 Online Books** icon. Depending on the system you are using, the icon is in the main product folder or the Windows Start menu.

Note: The **DB2 Online Books** icon is only available if you do not install the Information Center.

Setting up a Document Server

By default the DB2 information is installed on your local system. This means that each person who needs access to the DB2 information must install the same files. To have the DB2 information stored in a single location, use the following instructions:

1. Copy all files and sub-directories from \sqllib\doc\html on your local system to a web server. Each book has its own sub-directory containing all the necessary HTML and GIF files that make up the book. Ensure that the directory structure remains the same.

2. Configure the web server to look for the files in the new location. For information, see *Setting up DB2 Online Documentation on a Web Server* at:
`http://www.software.ibm.com/data/pubs/papers/db2html.html`
3. If you are using the Java version of the Information Center, you can specify a base URL for all HTML files. You should use the URL for the list of books.
4. Once you are able to view the book files, you should bookmark commonly viewed topics such as:
 - List of books
 - Tables of contents of frequently used books
 - Frequently referenced articles like the *ALTER TABLE* topic
 - Search form.

For information about setting up a search, see the *What's New* book.

Searching Online Books

To search for information in the HTML books, you can do the following:

- Click on **Search the DB2 Books** at the bottom of any page in the HTML books. Use the search form to find a specific topic.
- Click on **Index** at the bottom of any page in an HTML book. Use the Index to find a specific topic in the book.
- Display the Table of Contents or Index of the HTML book, and then use the find function of the Web browser to find a specific topic in the book.
- Use the bookmark function of the Web browser to quickly return to a specific topic.
- Use the search function of the Information Center to find specific topics. See "Information Center" on page 227 for details.

Printing the PostScript Books

If you prefer to have printed copies of the manuals, you can decompress and print PostScript versions. For the file name of each book in the library, see the table in "DB2 Books" on page 220.

Note: Specify the full path name for the file you intend to print.

On OS/2 and Windows platforms:

1. Copy the compressed PostScript files to a hard drive on your system. The files have a file extension of .exe and are located in the `x:\doc\language\books\ps` directory, where `x`: is the letter representing the CD-ROM drive and *language* is the two-character country code that represents your language (for example, EN for English).

2. Decompress the file that corresponds to the book that you want. The result from this step is a printable PostScript file with a file extension of .psz.
3. Ensure that your default printer is a PostScript printer capable of printing Level 1 (or equivalent) files.
4. Enter the following command from a command line:

```
print filename.psz
```

On UNIX-based platforms:

1. Mount the CD-ROM. Refer to your *Quick Beginnings* manual for the procedures to mount the CD-ROM.
2. Change to /cdrom/doc/%L/ps directory on the CD-ROM, where /cdrom is the mount point of the CD-ROM and %L is the name of the desired locale. The manuals will be installed in the previously-mentioned directory with file names ending with .ps.Z.
3. Decompress and print the manual you require using the following command:

- For AIX:

```
zcat filename ] qprt -P PSprinter_queue
```

- For HP-UX, Solaris, or SCO UnixWare 7:

```
zcat filename ] lp -d PSprinter_queue
```

- For Silicon Graphics IRIX and SINIX:

```
zcat < filename ] lp -d PSprinter_queue
```

where *filename* is the name of the full path name and extension of the compressed PostScript file and *PSprinter_queue* is the name of the PostScript printer queue.

For example, to print the English version of *Quick Beginnings for UNIX* on AIX, you can use the following command:

```
zcat /cdrom/doc/en/ps/db2ixe50.ps.Z ] qprt -P ps1
```

Ordering the Printed DB2 Books

You can order the printed DB2 manuals either as a set, or individually. There are three sets of books available. The form number for the entire set of DB2 books is SB0F-8915-00. The form number for the set of books updated for Version 5.2 is SB0F-8921-00. The form number for the books listed under the heading "Cross-Platform Books" is SB0F-8914-00.

Note: These form numbers only apply if you are ordering books that are printed in the English language.

You can also order books individually by the form number listed in "DB2 Books" on page 220. To order printed versions, contact your IBM authorized dealer or marketing

representative, or phone 1-800-879-2755 in the United States or 1-800-IBM-4YOU in Canada.

Information Center

The Information Center provides quick access to DB2 product information. You must install the DB2 administration tools to obtain the Information Center.

Depending on your system, you can access the Information Center from the:

- Main product folder
- Toolbar in the Control Center
- Windows Start menu
- Help menu of the Control Center
- **db2ic** command.

The Information Center provides the following kinds of information. Click on the appropriate tab to look at the information:

Tasks	Lists tasks you can perform using DB2.
Reference	Lists DB2 reference information, such as keywords, commands, and APIs.
Books	Lists DB2 books.
Troubleshooting	Lists categories of error messages and their recovery actions.
Sample Programs	Lists sample programs that come with the DB2 Software Developer's Kit. If the Software Developer's Kit is not installed, this tab is not displayed.
Web	Lists DB2 information on the World Wide Web. To access this information, you must have a connection to the Web from your system.

When you select an item in one of the lists, the Information Center launches a viewer to display the information. The viewer might be the system help viewer, an editor, or a Web browser, depending on the kind of information you select.

The Information Center provides some search capabilities so you can look for specific topics, and filter capabilities to limit the scope of your searches.

For a full text search, follow the *Search DB2 Books* link in each HTML file, or use the search feature of the help viewer.

The HTML search server is usually started automatically. If a search in the HTML information does not work, you may have to start the search server via its icon on the Windows or OS/2 desktop.

Refer to the release notes if you experience any other problems when searching the HTML information.

Appendix B. Installation Troubleshooting

Use this section to correct errors that occur during installation.

Error: Invalid TCP/IP port range

Solution: Ensure that the port range designated for the instance-owning machine is available on this machine. To correct this error, do one of the following:

- Free up the port range that you specified for the instance-owning machine of this machine by editing the `x:\winnt\system32\drivers\etc\services` file, where `x:` represents the drive where you installed Windows NT.
- Enter the **db2set** command to use a different port range for the partitioned database system as follows:

```
db2set DB2PORTRANGE=begin_port,ending_port
```

Note: You must enter the **db2set** command on all machines that make up your partitioned database system.

Error: Invalid TCP/IP port for the Performance Monitor

Solution: Ensure that the port number designated for the Performance Monitor is available on this machine. To correct this error, do one of the following:

- Free up the port that you specified during the installation of the instance-owning machine on this machine by editing the `x:\winnt\system32\drivers\etc\services` file, where `x:` represents the drive where you installed Windows NT.
- Enter the **db2set** command to globally set a different port for the Performance Monitor as follows:

```
db2set -g DB2CCMSRV=port_number
```

Note: You must enter the **db2set** command on all machines that make up your partitioned database system.

Error: Name conflict

Solution: An instance with the same name already exists on this machine. To correct this error, do one of the following:

- Add this machine to a different instance
- Drop the local instance that is causing the name conflict.

Error: Cannot validate username during configuration

Solution: Ensure that the specified user account exists in the domain.

Error: Cannot validate password during configuration

Solution: Ensure that the user account that you are using to run the setup program belongs to the local Administrators group and has the "Act as part of the operating system" advanced user right. If it doesn't, assign the required permissions, then log off and log back on as this user; alternatively, specify a different user account that has the required permissions.

Appendix C. National Language Support (NLS)

This appendix contains information about the National Language Support (NLS) provided by DB2, including information about supported languages and code pages. For more detailed information on developing applications, refer to the *Embedded SQL Programming Guide*.

Code Page and Language Support

During installation of DB2, the country, codepage and regional settings are established. You can, however, change these settings after installing DB2. This includes changing regional settings such as code page, country language (for monetary, date, and numeric formatting) and time zone. When a new connection to a database is made, the database manager uses these new values.



You must ensure that your regional settings are set correctly. DB2 may not produce the expected results if the country, code page, or regional settings are incorrect for the intended language.

Table 14 on page 232 shows the languages into which the DB2 messages are translated.

Note: The code page values in the table that follows are also used as directory names on DB2 CD-ROMs. For example, a reference to `x:\language\win32\install` would be `x:\en\win32\install` for English. For more detailed information on the languages and code pages support, refer to the *Administration Guide*.

<i>Table 14. Languages and Code Pages</i>			
Country Code	Language	Country Code	Language
BG	Bulgarian	HU	Hungarian
BR	Brazilian Portuguese	IL	Hebrew
CN	Simplified Chinese (PRC)	IT	Italian
CZ	Czech	JP	Japanese
DE	German	KR	Korean
DK	Danish	NO	Norwegian
EN	English	PL	Polish
ES	Spanish	RU	Russian
FI	Finnish	SE	Swedish
FR	French	SI	Slovenian
GR	Greek	TR	Turkish
		TW	Traditional Chinese (Taiwan)

Appendix D. Naming Rules



Go to the section that describes the naming rules that you require information on:

- “General Naming Rules”
 - “Database, Database Alias, and Catalog Node Names” on page 234
 - “Object Names” on page 234
 - “Usernames, Group Names, and Instance Names” on page 235
 - “DB2SYSTEM Naming Rules” on page 237
 - “Workstation Names (nname)” on page 236
 - “Password Rules” on page 237
-

General Naming Rules

Unless otherwise specified, all names can include the following characters:

- A through Z

Note: When used in most names, characters A through Z are converted from lowercase to uppercase.

- 0 through 9
- @
- #
- \$
- _ (underscore)

Unless otherwise specified, all names must begin with one of the following characters:

- A through Z
- @
- #
- \$

Do not use SQL reserved words to name tables, views, columns, indexes, or authorization IDs. A list of SQL reserved words is included in the *SQL Reference*.

Database, Database Alias, and Catalog Node Names

Database names are the identifying names assigned to databases in the database manager.

Database alias names are synonyms given to remote databases. Database aliases must be unique within the System Database Directory in which all aliases are stored.

When naming a database or database alias, see “General Naming Rules” on page 233.

In addition, the name you specify can contain 1 to 8 characters.

Notes:

1. To avoid potential problems, do not use the special characters @, #, and \$ in a database name if you intend to have a client remotely connect to a host database. Also, because these characters are not common to all keyboards, do not use them if you plan to use the database in another country.
2. On Windows NT systems, ensure that no instance name is the same as a service name.

Object Names

Database objects include:

- Tables
- Views
- Columns
- Indexes
- User-defined functions (UDFs)
- User-defined types (UDTs)
- Triggers
- Aliases
- Table spaces
- Schemas

When naming database objects, see “General Naming Rules” on page 233.

In addition, the name you specify:

- Can contain 1 to 18 characters

- Cannot be any of the SQL reserved words that are listed in the *SQL Reference*.

Note: Using delimited identifiers, it is possible to create an object that violates these naming rules; however, subsequent use of the object could result in errors.

For example, if you create a column with a + or – sign included in the name and you subsequently use that column in an index, you will experience problems when you attempt to reorganize the table. To avoid potential problems with the use and operation of your database, *do not* violate these rules.

Username, Group Name, and Instance Names

Username, group, or instance names are the identifiers assigned to individual users, groups, or instances, see “General Naming Rules” on page 233.

In addition, the name you specify:

- Can contain 1 to 8 characters
- Cannot be any of the following:
 - USERS
 - ADMINS
 - GUESTS
 - PUBLIC
 - LOCAL
- Cannot begin with:
 - IBM
 - SQL
 - SYS
- Cannot include accented characters
- In general, when naming users, groups, or instances:
 - On OS/2, use uppercase names.
 - On Windows 32-bit operating systems, use any case.
 - On UNIX, use lowercase names.

Workstation Names (*nname*)

A workstation name is used to specify the NetBIOS name for a database server or database client that resides on the local workstation. This name is stored in the database manager configuration file. The workstation name is known as the *workstation nname*. When naming workstations, see "General Naming Rules" on page 233.

In addition, the name you specify:

- Can contain 1 to 8 characters;
- Cannot include &, #, and @;
- Must be unique within the network.

In an partitioned database system, there is still only one *workstation nname* that represents the entire partitioned database system, but each node has its own derived unique NetBIOS *nname*.

The *workstation nname* that represents the partitioned database system is stored in the instance-owning database partition server's database manager configuration file.

Each node's unique *nname* is a derived combination of the *workstation nname* and the node number.

For a node that does not own an instance, its NetBIOS *nname* is derived as follows:

- 1** The first character of the instance-owning machine's *workstation nname* is used as the first character of the node's NetBIOS *nname*.
- 2** The next 1 to 3 characters represent the node number. The range is from 1 to 999.
- 3** The remaining characters are taken from instance-owning machine's *workstation nname*. The number of remaining characters depend on the length of the instance-owning machine's *workstation nname*. This number can be from 0 to 4.

For example:

Instance-Owning Machine's <i>Workstation nname</i>	Node Number	Derived Node NetBIOS <i>nname</i>
GEORGE	3	G3ORGE
A	7	A7
B2	94	B942
N0076543	21	N216543
GEORGE5	1	G1RGE5

Note: If you have changed the default workstation *nname* during the installation, the workstation *nname*'s last 4 characters should be unique across the NetBIOS network to minimize the chance of deriving a conflicting NetBIOS *nname*.

DB2SYSTEM Naming Rules

DB2SYSTEM names are used by DB2 to identify a physical DB2 machine, system, or workstation within a network. On UNIX, the *DB2SYSTEM* name defaults to the TCP/IP hostname. On OS/2, you must specify the *DB2SYSTEM* name during install. On Windows 32-bit operating systems, you do not need to specify a *DB2SYSTEM* name; the DB2 setup program detects the NT Computer name and assigns it to *DB2SYSTEM*.

When creating a *DB2SYSTEM* name, see "General Naming Rules" on page 233.

In addition, the name you specify:

- Must be unique within a network;
- Can contain a maximum of 21 characters.

Password Rules

When determining passwords on OS/2 and Windows 3.x, the password you specify can consist of a maximum of 8 characters. On all other operating systems, the word or name you specify can consist of a maximum of 18 characters.

Appendix E. Notices

Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent product, program or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter 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 the

IBM Director of Licensing,
IBM Corporation,
500 Columbus Avenue,
Thornwood, NY, 10594
USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Canada Limited
Department 071
1150 Eglinton Ave. East
North York, Ontario
M3C 1H7
CANADA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

This publication 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.

Trademarks

The following terms are trademarks or registered trademarks of the IBM Corporation in the United States and/or other countries:

ACF/VTAM	MVS/ESA
ADSTAR	MVS/XA
AISPO	NetView
AIX	OS/400
AIXwindows	OS/390
AnyNet	OS/2
APPN	PowerPC
AS/400	QMF
CICS	RACF
C Set++	RISC System/6000
C/370	SAA
DATABASE 2	SP
DatagLANce	SQL/DS
DataHub	SQL/400
DataJoiner	S/370
DataPropagator	System/370
DataRefresher	System/390
DB2	SystemView
Distributed Relational Database Architecture	VisualAge
DRDA	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WIN-OS/2
IBM	
IMS	
Lan Distance	

Trademarks of Other Companies

The following terms are trademarks or registered trademarks of the companies listed:

C-bus is a trademark of Corollary, Inc.

HP-UX is a trademark of Hewlett-Packard.

Java, HotJava, Solaris, Solstice, and Sun are trademarks of Sun Microsystems, Inc.

Microsoft, Windows, Windows NT, Visual Basic, and the Windows logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

SCO is a trademark of The Santa Cruz Operation.

SINIX is a trademark of Siemens Nixdorf.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, or service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Appendix F. Contacting IBM

This section lists ways you can get more information from IBM.

If you have a technical problem, please take the time to review and carry out the actions suggested by the *Troubleshooting Guide* before contacting DB2 Customer Support. Depending on the nature of your problem or concern, this guide will suggest information you can gather to help us to serve you better.

For information or to order any of the DB2 Universal Database products contact an IBM representative at a local branch office or contact any authorized IBM software remarketer.

Telephone

If you live in the U.S.A., call one of the following numbers:

- 1-800-237-5511 to learn about available service options.
- 1-800-IBM-CALL (1-800-426-2255) or 1-800-3IBM-OS2 (1-800-342-6672) to order products or get general information.
- 1-800-879-2755 to order publications.

For information on how to contact IBM outside of the United States, see Appendix A of the IBM Software Support Handbook. You can access this document by accessing the following page:

<http://www.ibm.com/support/>

then performing a search using the keyword "handbook."

Note that in some countries, IBM-authorized dealers should contact their dealer support structure instead of the IBM Support Center.

World Wide Web

<http://www.software.ibm.com/data/> <http://www.software.ibm.com/data/db2/library/>

The DB2 World Wide Web pages provide current DB2 information about news, product descriptions, education schedules, and more. The DB2 Product and Service Technical Library provides access to frequently asked questions, fixes, books, and up-to-date DB2 technical information. (Note that this information may be in English only.)

Anonymous FTP Sites

<ftp.software.ibm.com>

Log on as anonymous. In the directory `/ps/products/db2`, you can find demos, fixes, information, and tools concerning DB2 and many related products.

Internet Newsgroups

comp.databases.ibm-db2, bit.listserv.db2-l

These newsgroups are available for users to discuss their experiences with DB2 products.

CompuServe

GO IBMDB2 to access the IBM DB2 Family forums

All DB2 products are supported through these forums.

To find out about the IBM Professional Certification Program for DB2 Universal Database, go to <http://www.software.ibm.com/data/db2/db2tech/db2cert.html>

Index

A

- Access Profile
 - creating 160
 - using 160
- accessing data
 - through DB2 Connect 13
 - through Net.Data or JDBC 15
- accessing multiple servers 6
- Administration Server
 - CCA 97
 - Control Center 97
 - create manually 96
 - logon account 100
 - overview 10, 95
 - remove 100
 - setup 143
 - start 99
 - stop 99
 - updating instance lists 98
- Administrator name
 - valid names 44
- ADSM
 - software requirements 38
- agents
 - coordinating 28
 - pools 28
 - purpose 28
 - subagent 28
- Alert Center 9
- APPC
 - server 184
 - software requirements 38, 39, 40
 - support on OS/2
 - Communications Manager for OS/2 39
 - Communications Server for OS/2 39
 - support on Windows NT
 - IBM Communications Server for NT 38
 - Microsoft SNA Server 38
 - supported platforms 38
- application development
 - using Net.Data or JDBC 15
 - using ODBC 103
- application programs

- application programs (*continued*)
 - connecting to logical node 117
 - coordinator node 20
 - migrating to DB2 Extended Enterprise Edition 20
 - serviced by agents 28
- authentication
 - client 55
 - server 171
- authority
 - levels 78

B

- binding
 - utilities and applications 102

C

- capacity
 - disk storage 23
- catalog node 24
- cataloging
 - IPX/SPX node 183
 - NetBIOS node 180
 - TCP/IP node 177
- CD-ROM
 - installing DB2 Client Applications Enablers 55
 - installing DB2 Software Developer's Kit 55, 61
- Client Application Enabler
 - installing 55
 - installing on OS/2 61
 - installing on Windows 3.x 63
 - software requirements 38
- Client Configuration Assistant
 - administering clients communications 69
 - and DB2 setup 95
 - changing privileges 80
 - client profiles 69
 - exporting 161
 - using 161
 - discover 165
 - discovering databases 70
 - overview 11
 - searching the network 69, 163

- Client Configuration Assistant (*continued*)
 - setting configuration parameters 130, 167
 - testing the database connection 73
 - using access profiles 160
- client profile
 - exporting 161
- clients
 - installing 55
 - operating systems supported 5, 55
 - overview 5, 55
 - verifying the installation 73
- CLP
 - See using the command line processor (CLP)
- column
 - long-field, partitioning key considerations 25
- Command Center
 - entering DB2 commands 113
 - entering SQL statements 113
 - overview 7
- commands
 - alter table 145
 - connecting to logical node 117
 - db2admin 95, 99
 - db2admin create 96
 - db2admin drop 100
 - db2admin setid 100
 - db2admin start 99
 - db2admin stop 99
 - db2ckmig 143, 146
 - db2icrt 85
 - db2idrop
 - db2ilist 85
 - db2iset 98
 - db2nlist 98
 - db2set 120, 172
 - db2start 87, 98, 153
 - db2stop 87
 - db2uidl 153
 - drop function 145
 - drop trigger 145
 - drop view 145
 - echo 121, 122
 - set 121
- communication protocols
 - APPC 38, 39, 40, 184
 - IPX/SPX 38, 39, 40, 181
 - Named Pipes 38, 39, 40, 173
 - NetBIOS 38, 39, 40, 177
 - TCP/IP 38, 39, 40, 173
- communications
 - configuring the client 69
 - configuring the server 171
 - managing 171
 - Client Configuration Assistant 69
 - fast communication manager 27
 - supported protocols
 - IPX/SPX 181
 - Named Pipes 173
 - NetBIOS 177
 - TCP/IP 173
- concurrency
 - improving with logical nodes 26
- configuration parameters
 - intra-parallel 28
 - max_time_diff 44
 - maxagents 28
 - num_poolagents 28
 - setting DB2 55, 80, 119, 130, 171
 - SYSADM_GROUP 80
- configuring
 - APPC 184
 - Named Pipes 173
 - NetBIOS 177
 - ODBC driver 106, 108
 - TCP/IP 173
- configuring client communications
 - setting configuration parameters 130, 154
 - using the Client Configuration Assistant 69
- configuring communications
 - using Client Configuration Assistant 69
- configuring server communications
 - generating an access profile 52
 - setting DB2COMM 160, 171
 - using the command line processor 171
- configuring servers 171
- connecting to a database 79
- connecting to the Internet using Net.Data 15
- Control Center
 - components 8
 - overview 7
- coordinator agent
 - description 28
- coordinator node
 - description 20
- cost-based query optimizer
 - description 20
- creating the sample database
 - using First Steps 78
 - using the command line processor 78

D

DAS considerations 96

database

- authentication type 147, 150
- definition 19
- migration 151
- naming rules 234
- partition 19

database access

- two-phase commit 7

database administration tools

- Control Center 7
- overview 7

database alias

- naming rules 234

database manager

- definition 19
- starting 87, 92
- stopping 92

database objects

- naming rules 234

database partition server

- definition 19

DB2 Client Application Enabler

- accessing databases 6
- installing
 - Windows 3.x 63
 - Windows 95 57
 - Windows NT 57
- overview 5, 6, 55
- platforms supported 5

DB2 Client Application Enablers CD-ROM

- Control Center 97
- installing clients from 55

DB2 Connect

- overview 3, 4, 13

DB2 Enterprise Edition

- memory requirements 31

DB2 Extended Enterprise Edition

- configuration planning 31
- hardware configuration 22

DB2 Installer

- See installation

DB2 library

- books 220
- Information Center 227
- language identifier for books 223
- late breaking information 223
- online help 218

DB2 library (*continued*)

- ordering printed books 226
- printing PostScript books 225
- searching online books 225
- setting up document server 224
- SmartGuides 217
- structure of 217
- viewing online books 224

DB2 Performance Monitor

- DB2Perf.DLL 134
- db2perfc command 136
- db2perfi command 134
- db2perfr command 134
- registering DB2 for Windows NT Performance Counters DLL 134

DB2 server, definition ix

DB2 Snapshot Monitor

- overview 11

db2 terminate command 87

DB2 Universal Database

- components included
 - Client Configuration Assistant 11
 - Control Center 7, 8
 - DB2 Client Application Enabler 5
 - DB2 Performance Monitor 11, 133
 - DB2 Snapshot Monitor 11, 15
 - Visual Explain 10, 11

DAS considerations 95

getting started 77

overview 8

platforms supported 7

software requirements 37

db2.ini file

- environment keywords 123

db2admin command

- db2admin create 96
- db2admin drop 100
- db2admin setid 100
- db2admin start 99
- db2admin stop 99, 100

db2ckmig command 146

DB2COMM 171

DB2DMNBCKCTLR

- registry value 126

db2fcmdm daemon

- description 27

db2icrt command 85

db2idrop command

db2ilist command 85

- db2imigr command 143
- DB2INSTANCE 92
- DB2NBADAPTERS 178
- DB2NODE
 - connecting to logical node 117
- db2perfc command
 - resetting DB2 performance values 136
- db2perfi command
 - registering performance counters 134
- db2perfr command
 - deregistering an administrative username and password 135
 - registering an administrative username and password 134
- db2set command
 - using 120, 172
- db2start command 87, 97
- db2stop command 87
- db2uidl command 153
- developing applications
 - using Net.Data or JDBC 15
- Discovery
 - administration configuration file 98
 - choosing a discovery method 165
 - client configuration 95
 - searching the network 163
 - setting Discovery parameters 167
- disk requirements
 - client 35
 - server 35
- Distributed Computing Environment
 - software requirements 38, 39

E

- environment variables 27
 - DB2INSTANCE 92, 128
 - db2instprof 128
 - DB2NODE 126
 - connecting to logical node 117
 - db2path 128
- error handling
 - migration 144
- execs
 - db2perfc.exe 136
 - db2perfi.exe 134
 - db2perfr.exe 134
 - db2uexit.exe 146
 - db2uext2.exe 146

- Extended Enterprise Edition
 - capacity 23
 - storage 23
- Extended Enterprise Edition configuration
 - logical nodes 26
 - security for an instance 26
 - shared nothing definition 19

F

- fast communication manager (FCM)
 - db2fcmdm daemon 27
 - description 27
- fixed disks
 - hardware requirements 35

G

- GB (gigabyte) 23
- gigabyte (GB) 23

H

- hardware requirements
 - fixed disk 35

I

- IBMCATGROUP nodegroup 24
- IBMDEFAULTGROUP nodegroup 24
- IBMTMPGROUP 24
- IBMTMPGROUP nodegroup 24
- installation
 - clients
 - OS/2 61
 - Windows 3.x 63
 - Windows 95 57
 - Windows NT 57
 - disk requirements
 - client 35
 - server 35
 - memory requirements
 - client 31
 - server 31
 - verifying the installation 50
- installing Extended Enterprise Edition
 - enabling FCM communications 44
 - preparations 43
- instance owner 27

- instance-owning machine
 - definition 27
- instances
 - creating multiple 83
 - definition 26, 83
 - listing 85
 - multiple on one processor 27
 - naming restrictions 85, 235
 - removing 93
 - running multiple 93
 - selecting 92
 - starting 87, 92
 - stopping 92
- inter-operator parallelism 21
- IPX/SPX
 - server 181
 - software requirements 39
 - supported platforms 182
 - verifying the connection 73

J

- Java Support 15
- Java, running programs 109
- JDBC
 - See Java Support
- JDBC, running programs 109
- Journal 9

L

- logging off the system 78
- logging on to the system 78
- Logical LAN Adapter Number (Lana Number)
 - default value 179
- logical node
 - connecting to 117
 - default 117
 - improving concurrency 26
 - purpose 26
 - supported configurations 26
- long-field columns
 - partitioning key considerations 25

M

- managing connections
 - client
 - overview 11
 - using the Client Configuration Assistant 11, 69

- managing connections (*continued*)
 - server
 - overview 11
 - using the Client Configuration Assistant 69
 - using the Command Line Processor 171
 - managing databases using the Control Center 8
 - memory requirements
 - client 31
 - estimating 31
 - recommended 31
 - server 31
 - Microsoft ODBC Driver Manager 105
 - Microsoft SNA Server
 - version required 40
 - migration
 - database 151
 - DB2 Version 2.x 141, 142
 - db2chkmig 146
 - of applications 20
 - post-installation 150
 - post-migration
 - explain tables 155
 - rebinding packages 153
 - unique index conversion 152
 - updating database and database manager
 - configuration 154
 - updating statistics 153
 - pre-installation 142
 - previous releases 141, 142
 - user exit program 146
 - multipartition nodegroup
 - See nodegroup

N

- Named Pipes
 - server 173
 - software requirements 39
- naming rules
 - database 234
 - database alias 234
 - database objects 234
 - general 233
 - groups 235
 - instance names 235
 - password 237
 - userid 235
 - username 57, 235
 - workstation name (nname) 236

- National Language Support (NLS)
 - code set 231
 - codepage support 231
 - determining codepage 232
- Nbf
 - network route required by DB2 179
- Net.Data 39
 - overview 15
- NetBIOS
 - autostarting NetBIOS 181
 - on the client 39
 - on the server 171, 177
 - supports SEARCH discovery 166
- Netfinity Server 39
- network route
 - setting to Nbf 178, 179
- NLS
 - See National Language Support (NLS)
- node
 - definition 19
 - in the same instance 26
- nodegroup
 - IBMCATGROUP 24
 - IBMDEFAULTGROUP 24
 - multipartition nodegroup 23

O

- ODBC
 - registering the driver manager 105
 - running ODBC-enabled applications 103
 - running programs 103
- odbcad32.exe 105
- operating systems
 - client 5
 - server 55

P

- parameters
 - SYSADM_GROUP 80
- partition
 - definition 19
- partitioning key
 - definition 25
- partitioning map
 - overview 25
- password
 - naming rules 237

- Performance Monitor
 - using 10
 - using with DB2 133
 - Windows NT Performance monitor
 - configuring 133
 - using 133
- planning
 - DB2 configuration 31
 - DB2 Connect configuration 31
- pools
 - agent 26
- privileges
 - required 80
- product
 - descriptions 3
 - overview 3
- profile registry
 - using 119
- programs
 - db2chkmig 143
- protocols
 - APPC 184
 - IPX/SPX 181
 - Named Pipes 173
 - NetBIOS 177
 - TCP/IP 173

Q

- query
 - cost-based optimizer 20

R

- registering
 - ODBC driver manager 105
- registry value
 - DB2DMNBCKCTLR 126
- registry values
 - db2account 124
 - db2codepage 125
 - db2comm 129, 171
 - db2country 125
 - db2dbdft 125
 - db2discoverytime 125
 - db2include 126
 - db2instdef 93, 126
 - db2nbadapters 129, 178
 - db2netreq 129
 - db2options 126

registry values (*continued*)

- db2porange 129
- db2slogon 126
- db2timeout 126
- db2tracename 127
- db2traceon 127
- db2trcflush 127
- db2trcsyserr 127
- db2yield 128
- general 124
- setting 119

removing

- administration server 100

restrictions

- instance name 85, 235

running applications

- database client 101
- ODBC considerations 103

S

sample database

- See creating the sample database

scaling the database system

- definition 25

Script Center 9

SDK Kit

- See Software Developer's Kit

security

- for an instance 26

selecting an instance 92

setting

- environment variables 119

setting configuration parameters 171

setting up client communications

- using the Client Configuration Assistant 69

setting up document server 224

setting up server communications

- using the command line processor 171

shared-nothing configuration

- definition 19
- storage 23

SMP

- See symmetric multiprocessor architecture

snapshot monitoring considerations 11

Software Developer's Kit

- overview 12

software registration 81

software requirements

- communication protocols 37

software requirements (*continued*)

- DB2 Client Application Enabler 37, 38, 39, 40
- DB2 Connect 37
- DB2 SDK 37, 38, 39, 40
- DB2 Universal Database 37
- Net.Data 38, 39, 40

SQL

- access plans
 - viewing using Visual Explain 11

starting

- database manager 87
- DB2 87

stopping

- DB2 87

subagent

- function 28

symmetric multiprocessor architecture 26

SYSADM

- privileges
 - controlling 80

SYSADM_GROUP parameter 80

system configuration

- with DB2 Connect 13
- with DB2 Universal Database 6

system environment variables

- DB2INSTANCE 92

system management 92

T

table

- in multinode nodegroup 25
- in single-node nodegroup 25
- maximum size 23

TB (terabyte) 23

TCP/IP

- preventing socket collisions 174
- resolving host address 177
- server 173
- set with db2set 129
- software requirements 39
- supports only one-phase commit 205
- supports SEARCH discovery 166
- troubleshooting 174

temporary table spaces

- See IBMTEMPGROUP

terabyte (TB) 23

Tools Setting 9

transparent parallelism 20

U

- user privileges
 - See privileges
- username
 - naming rules 235
- using the command line processor (CLP)
 - cataloging a database 151
 - connecting to a database 73, 79
 - updating the database manager configuration 176, 179, 183, 184
 - verifying the connection 73
- utilities
 - binding 101

V

- verifying the connection
 - using the Client Configuration Assistant 73
 - using the command line processor
 - IPX/SPX 181
 - Named Pipes 173
 - NetBIOS 177
 - TCP/IP 173, 184
- verifying the installation
 - client 73
 - server 50
- Visual Explain
 - overview 10, 11

W

- Windows NT
 - applications 136
 - database manager performance object 135
 - databases 135
 - Performance Monitor 133
- Windows NT Performance Monitor
 - using with DB2 133



Part Number: 04L7808



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

S09L-6713-01



04L7808

