IBM DB2 Universal Database
for UNIX**

Quick Beginnings

Version 5.2

S10J-8148-01

IBM DB2 Universal Database
for UNIX**

Quick Beginnings

Version 5.2

S10J-8148-01

Before using this information and the product it supports, be sure to read the general information under Appendix E, “Notices” on page 321.

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.

Contents

Welcome to DB2 Universal Database

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

Part 1 gives an overview of DB2 Universal Database.

Part 2 provides basic planning, installing, and configuration information for DB2 Universal Database.

Part 3 describes how to install a DB2 Client Application Enabler on any supported operating system.

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 installation and configuration tasks, such as installing DB2 products on UNIX operating systems using SMIT, swinstall, or db2_install; performing administration tasks using the Client Configuration Assistant; and using the command line processor for setting up server communications.

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 book titles and to emphasize words.
- `Monospace` indicates file names, directory paths, and 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 2, “Planning for Installation” on page 21.
Confirm that your system meets the DB2 disk and software prerequisites,	“Disk Requirements” on page 24 and “Software Requirements” on page 27.
Migrate from a previous release of DB2,	Chapter 6, “Migrating from Previous Versions and Releases” on page 57.
Install DB2 Universal Database from a CD-ROM,	“Preparing to Install the DB2 Products” on page 35.
Install the client products,	Chapter 7, “Installing DB2 Clients” on page 79.
Set up communications on the client to access databases,	Chapter 13, “Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 111 or Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.
Use the DB2 components,	Chapter 15, “Getting Started with DB2 Universal Database” on page 139.
Create the sample database,	“Creating the Sample Database” on page 140.
Connect to a database,	“Connecting to a Database” on page 141.
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 163.
Use the DB2 Command Center or the DB2 command line processor to enter SQL statements or DB2 commands interactively,	Chapter 19, “Entering DB2 Commands and SQL Statements” on page 181.
Print or view online documentation,	“Printing the PostScript Books” on page 305 or “Viewing Online Books” on page 304.

If you want to...	Refer to...
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 297 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 Enterprise Edition*, *DB2 Workgroup Edition, DB2 Connect Enterprise Edition*, and *DB2 Client Application Enabler*.

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 Universal Database Enterprise Edition
A relational database management system that enables local and remote clients and applications to create, update, control, and manage relational databases using Structured Query Language (SQL), ODBC, or CLI. Also included in DB2 Universal Database Enterprise Edition is the DB2 Connect support feature. This provides users with access to DB2 databases that reside on host systems such as MVS/ESA, OS/390, AS/400, VM, and VSE.

The *DB2 Net.Data* CD-ROM contains all supported DB2 Net.Data (formerly known as DB2 World Wide Web Connection) products. DB2 Net.Data enables application developers to create Internet applications that access data from DB2 databases.

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 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 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 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 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 8.

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 7, "Installing DB2 Clients" on page 79 for information on how to download the DOS client.

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

Figure 1 on page 6 shows a DB2 server with local applications and applications that run on remote clients. The server has either DB2 Workgroup Edition or DB2 Enterprise Edition installed. 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.

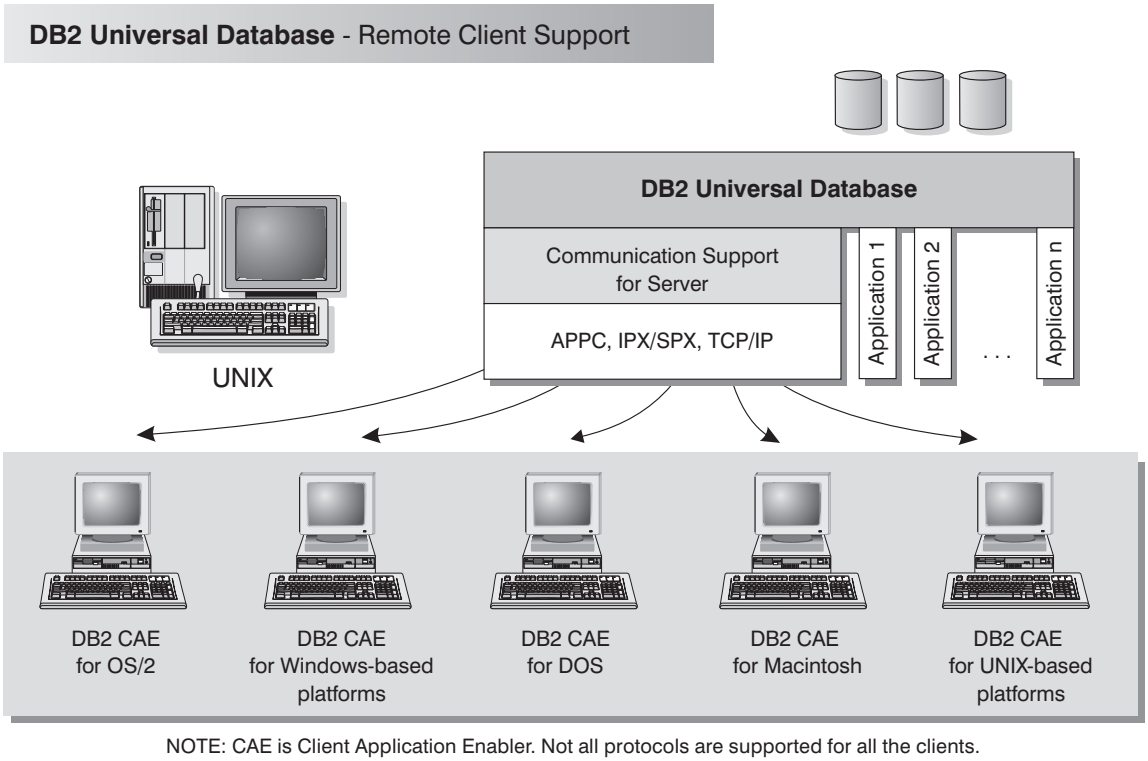


Figure 1. DB2 with Local Applications and Remote Clients

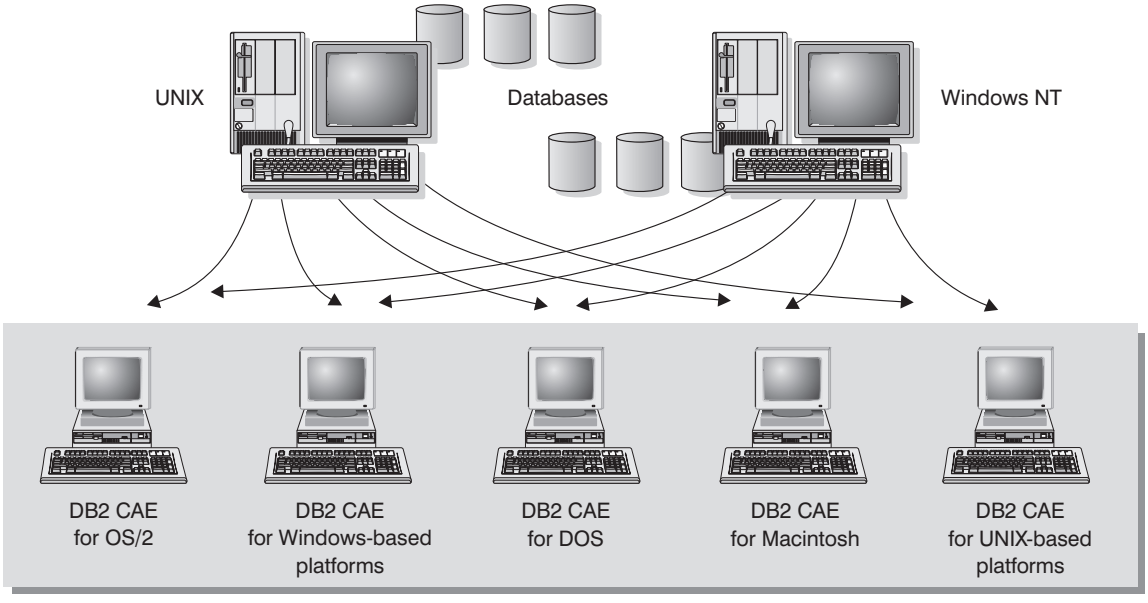
In addition to accepting requests from the clients that are listed above, DB2 Universal Database also has the DRDA Application Server function built in. It accepts requests from MVS, OS/400, VM, and other clients which are DRDA Application Requesters. See Chapter 28, "Accessing DB2 Universal Database Servers from Host and AS/400 Applications" on page 277, 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 on a Windows NT workstation that is connected to a LAN, and another DB2 server on a UNIX system 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.

Communicating Between DB2 Universal Database Servers



NOTE: CAE is Client Application Enabler.

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

You can administer local databases on DB2 servers by using the command line processor. You can also perform database administration tasks remotely from an OS/2 or Windows 32-bit operating systems by using the DB2 administration 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.

You can find additional information in the *Administration Getting Started* guide.

Managing Databases Using the Control Center

The Control Center displays database objects (such as databases, tables, 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 on page 9 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.

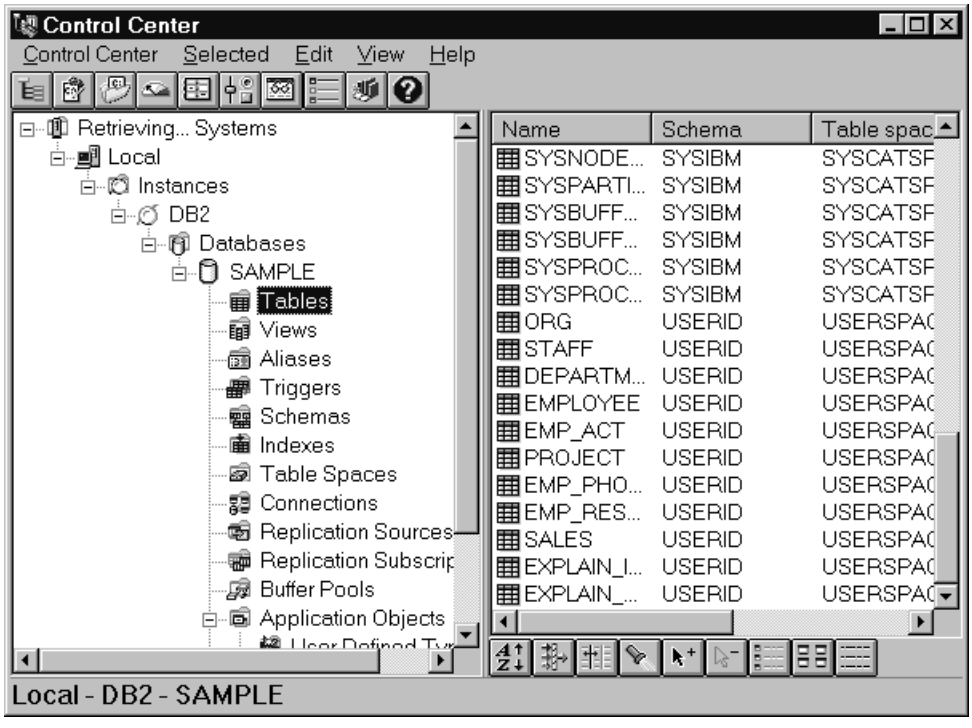


Figure 3. Control Center Main Window

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
- Backup and recover a database or table space
- Define the replication sources and subscriptions to replicate data between systems

You can also control database manager environments (which are known as DB2 instances) by:

- Maintaining the communications protocols
- Setting database manager configuration values that affect performance

SmartGuides are provided to help you perform complex tasks. For example, a SmartGuide is available to tune the performance of your system. See “SmartGuides” on page 297 for descriptions of the various SmartGuides and how to start them.

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, and Replication.

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 11 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 configure instances, and catalog databases on the DB2 server. 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. For more information, see Chapter 17, “Using the Administration Server” on page 157.

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.

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 Communications on the Server

The Control Center allows you to view, update, and reset server protocol settings. These functions are accessed by clicking with the right mouse button on an instance and selecting the **Setup communications** option from the pop-up menu. This tool helps database administrators to:

- Configure communication parameters in the database manager.
- Generate database information in a profile that can be used to configure clients.

You can find additional information in the online help for the Control Center.

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. Use the command line processor to set up DB2 clients on UNIX platforms.

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

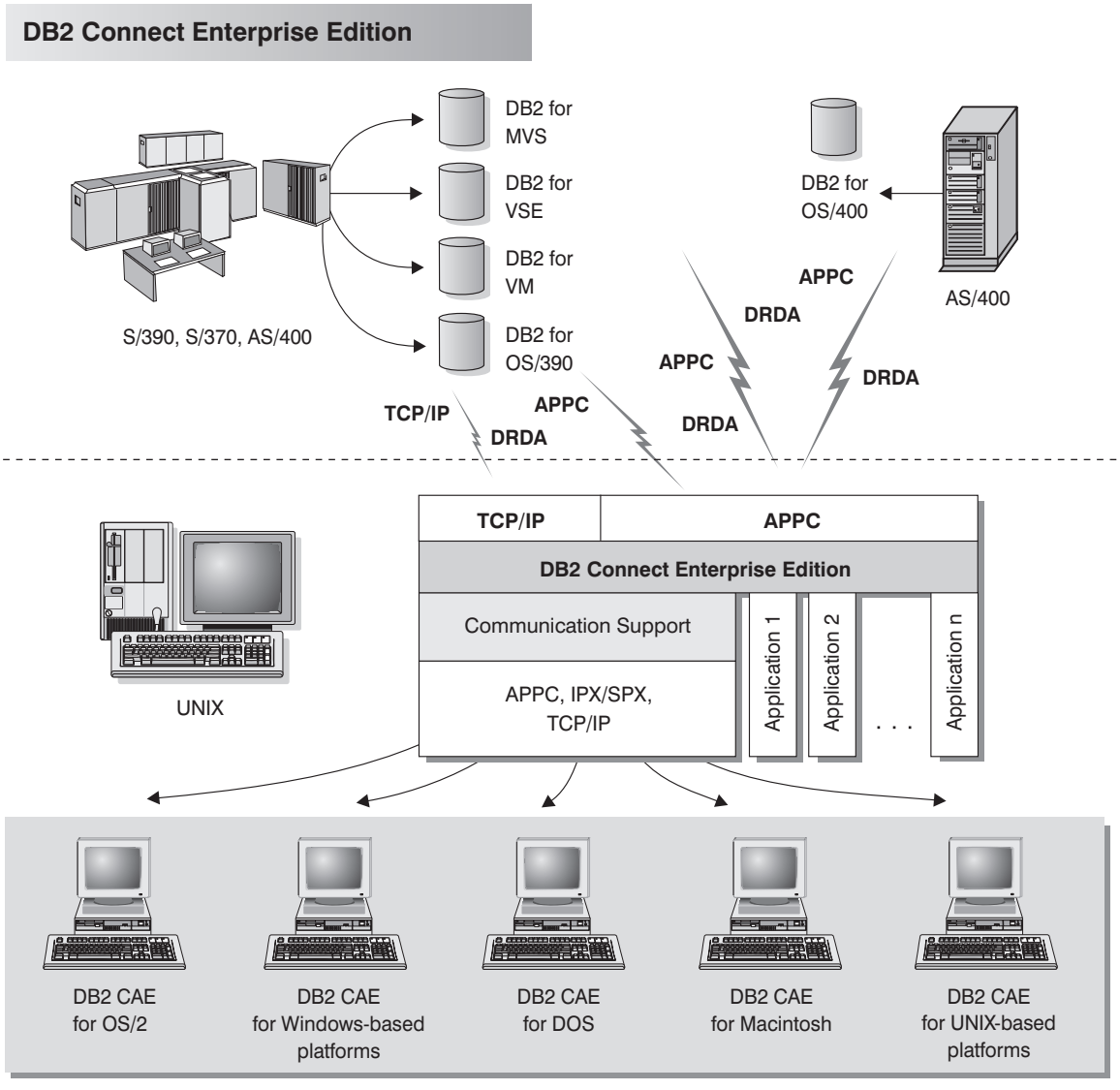
DB2 Connect Enterprise Edition gives clients on your LAN access to data that is stored on host systems. It provides applications with transparent access to host data through a standard architecture for managing distributed data. This standard is known as Distributed Relational Database Architecture (DRDA). 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 15 for an example of clients connecting to host databases through 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

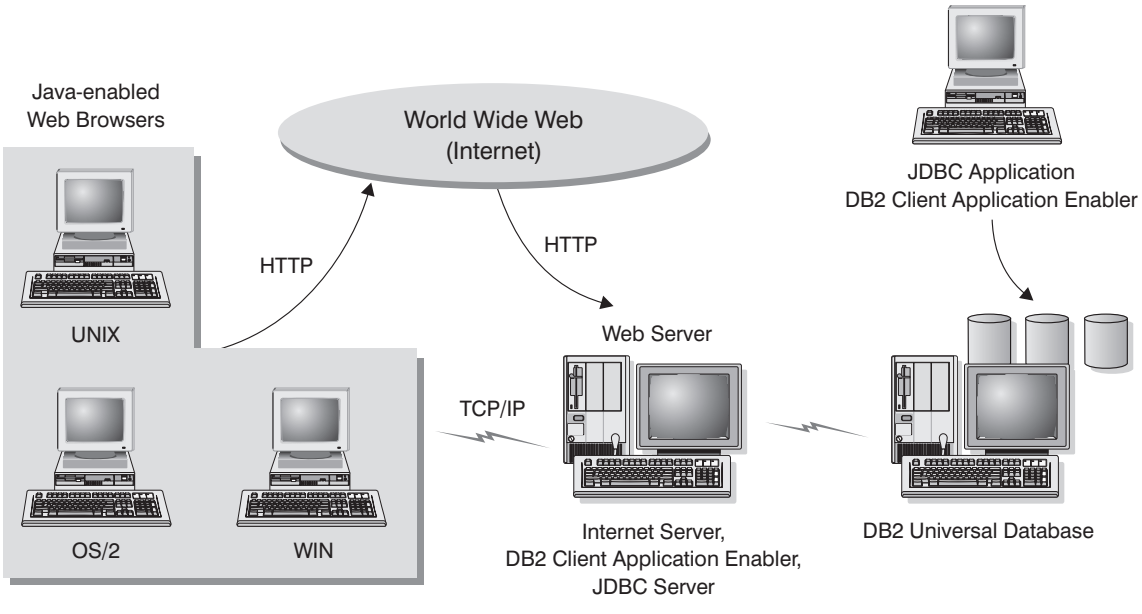


Figure 5. Accessing Internet Data Stored on DB2 Using JDBC

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 18 for an example of Net.Data that is being used to access data from remote DB2 databases.

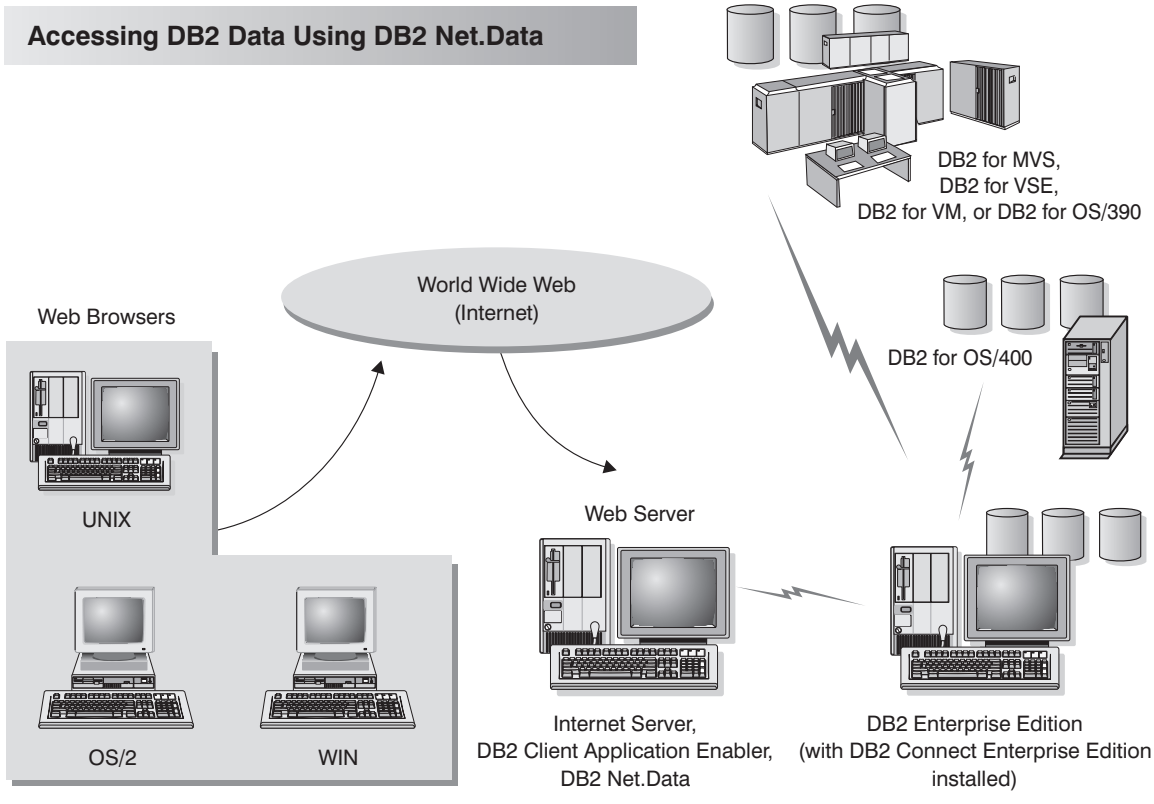


Figure 6. Accessing Internet Data Stored on DB2 Using Net.Data

Part 2. Installation and Initial Configuration

Chapter 2. 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 “Preparing to Install the DB2 Products” on page 35.

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, 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 7 on page 22. This diagram shows a DB2 Universal Database server 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 server using one of the supported protocols: APPC, IPX/SPX, or TCP/IP.

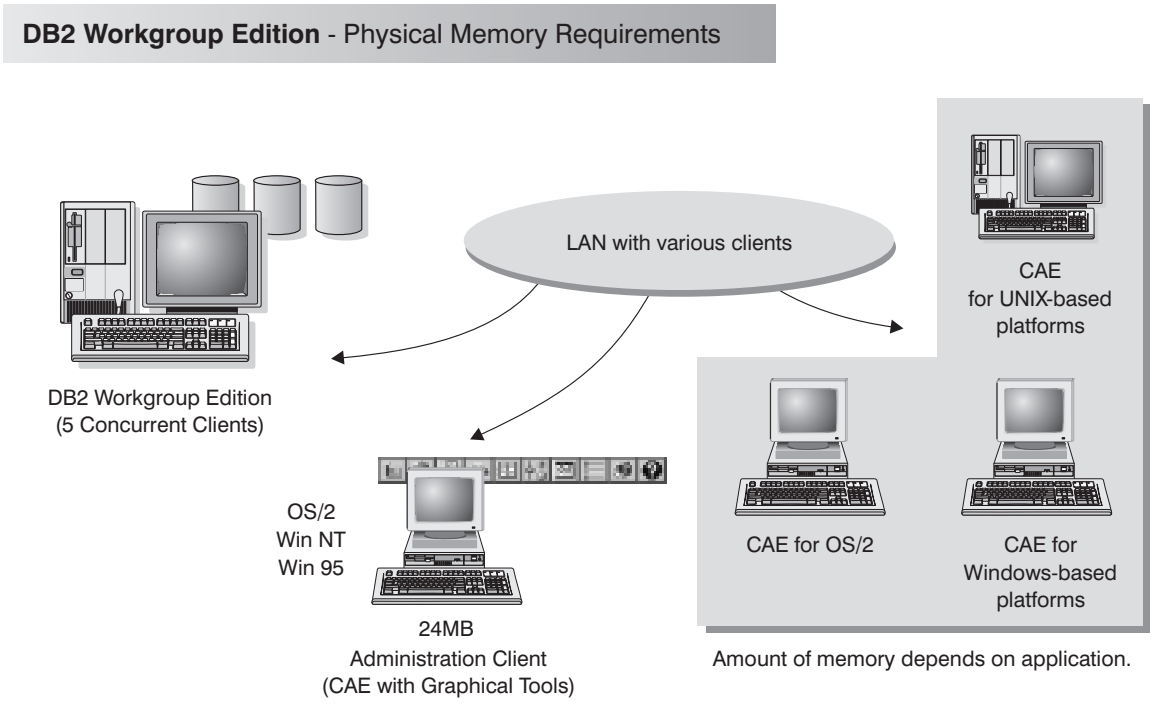


Figure 7. Configuration for Accessing DB2 Servers on the LAN

The amount of memory you need on your server depends on the number of concurrent users you will have, and the size of the databases. The diagram shows a scenario of five concurrent clients connected to the server with a simple application using static SQL. For the server, it is recommended that a minimum of 64 MB of memory be installed to accommodate five concurrent clients. We suggest that you have a minimum of 96 MB of memory to accommodate 25 concurrent clients and 128 MB of memory for 50 concurrent clients. (If you have a more complex application environment, your server may require significantly 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 8. You will need to install the DB2 Connect Support option of DB2 Enterprise Edition.

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

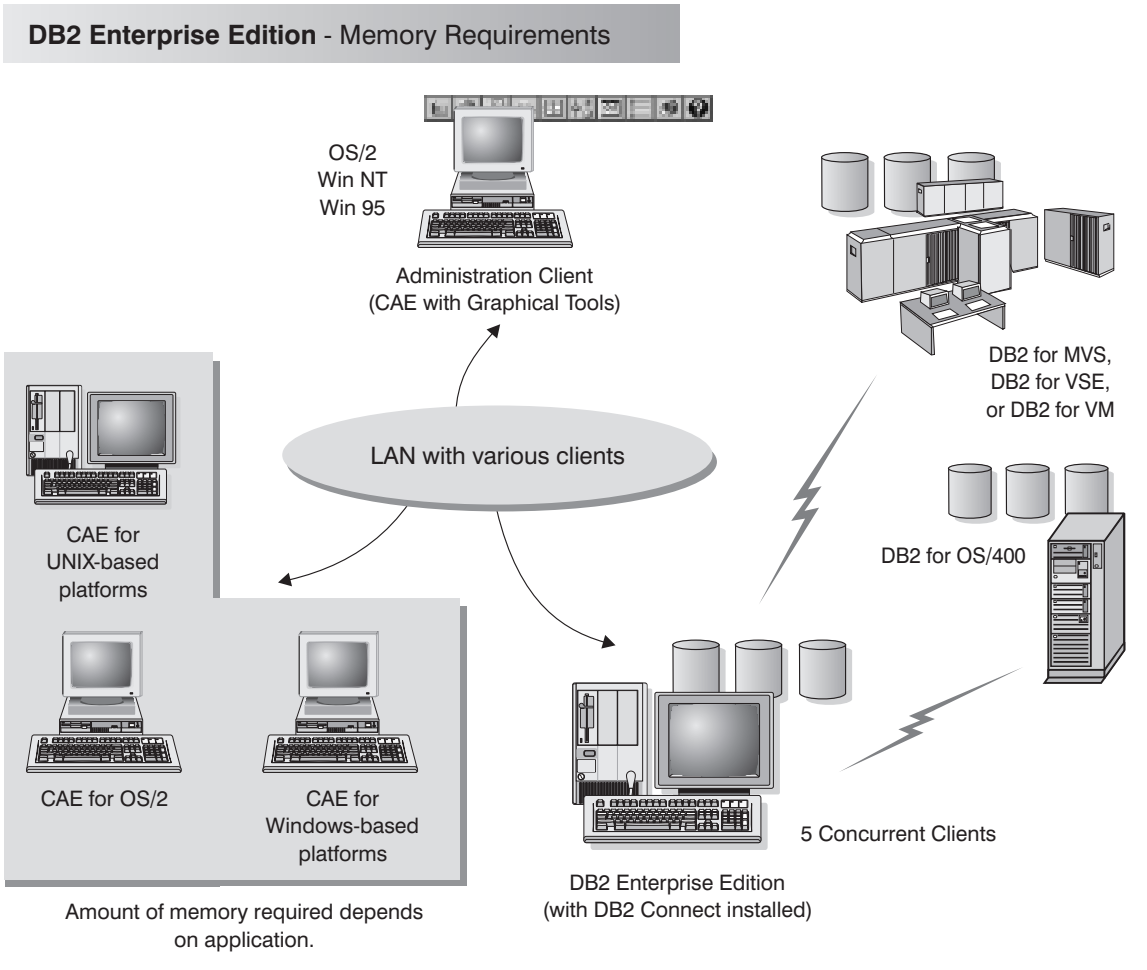


Figure 8. Configuration for Accessing LAN and Host Data

The amount of memory you need on your server depends on the number of concurrent users you will have, and the size of the databases. The diagram shows a scenario of five concurrent clients connected to the server with a simple application using static SQL. For the server, it is recommended that a minimum of 64 MB of memory be installed to accommodate five concurrent clients. We suggest that you have a minimum of 96 MB of memory to accommodate 25 concurrent clients and 128 MB of memory for 50 concurrent clients. (If you have a more complex application environment, your server may require significantly 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 16 or Figure 6 on page 18.

Follow the guidelines in “Configuration for Accessing DB2 Servers on the LAN” on page 21 and “Configuration for Accessing LAN and Host Data” on page 23 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 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 for AIX	
DB2 Enterprise Edition	55 MB
• Online documentation in HTML format (English)	72 MB
• Far-East Code Page Conversion Support	5 MB
DB2 Universal Database for HP-UX	
DB2 Enterprise Edition	77 MB
• Online documentation in HTML format (English)	72 MB
• Far-East Code Page Conversion Support	5 MB
DB2 Universal Database for SCO UnixWare 7	
DB2 Universal Database Workgroup Edition	70 MB
• Online documentation in HTML format (English)	72 MB
• Far-East Code Page Conversion Support	5 MB
DB2 Enterprise Edition	70 MB
• Online documentation in HTML format (English)	72 MB
• Far-East Code Page Conversion Support	5 MB
DB2 Universal Database for Solaris	
DB2 Enterprise Edition	54 MB
• Online documentation in HTML format (English)	72 MB
• Far-East Code Page Conversion Support	5 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 AIX	
DB2 Client Application Enabler for AIX	22 MB
• Online documentation in HTML format	25 MB
DB2 Client Application Enabler for HP-UX	
DB2 Client Application Enabler for HP-UX	18 MB
• Online documentation in HTML format	25 MB
DB2 Client Application Enabler for Macintosh	
DB2 Client Application Enabler for Macintosh	8 MB
• Online documentation in HTML format	7 MB
DB2 Client Application Enabler for OS/2	
DB2 Client Application Enabler for OS/2 — this amount includes the following components: • Client Configuration Assistant • Graphical Tools • Control Center • Performance Monitor • Visual Explain	70 MB 2 MB 10 MB 24 MB 8 MB 4 MB
• Online documentation in HTML format	25 MB
DB2 Client Application Enabler for SCO OpenServer Version 2	
DB2 Client Application Enabler for SCO OpenServer Version 2	16 MB
• Online documentation in HTML format	25 MB
DB2 Client Application Enabler for SCO UnixWare 7	
DB2 Client Application Enabler for SCO UnixWare 7	16 MB
• Online documentation in HTML format	25 MB
DB2 Client Application Enabler for Silicon Graphics IRIX	
DB2 Client Application Enabler for Silicon Graphics IRIX	14 MB
• Online documentation in HTML format	19 MB
DB2 Client Application Enabler for SINIX Version 2	
DB2 Client Application Enabler for SINIX	16 MB
• Online documentation in HTML format	21 MB

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

	Recommended Minimum Disk (MB)
DB2 Client Application Enabler for Solaris	
DB2 Client Application Enabler for Solaris	16 MB
• 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
• 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
• Client Configuration Assistant	2 MB
• Control Center	18 MB
• Performance Monitor	6 MB
• Visual Explain	3 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. These associated products include the DB2 Client Application Enabler, DB2 Software Developer's Kit, Net.Data, and DB2 Connect.

Server Product Requirements

Table 3 on page 28 lists the operating system and communications software required for DB2 Universal Database.

Table 3 (Page 1 of 3). Software Requirements for DB2 Servers

Product	Hardware/Operating System	Communications
• DB2 Enterprise Edition (including DB2 Connect Enterprise Edition)	RISC System/6000 and the following: • AIX Version 4.1.4 or later	APPC, IPX/SPX, or TCP/IP • For APPC connectivity with two-phase commit, you require IBM Communications Server Version 4.1 or later for AIX in order to use the LU6.2 Syncpoint Manager function. If you do not need to use two-phase commit, IBM SNA Server/6000 Version 1.x or later is sufficient. • The AIX base operating system provides TCP/IP connectivity. For two-phase commit, TCP/IP connectivity is supported only on the AIX Version 4.1.4 or later server. • IPX/SPX connectivity is provided by: – IBM NetBIOS and IPX/SPX for AIX Version 2.1 – IBM NetWare for AIX Version 3.11B – AIX Connections feature of AIX 4.1.4 – AIX base operating system 4.1.4 or later **Notes:** 1. If you want to install the DRDA Application Server function of the DB2, you must install the Syncpoint Manager (SPM) and its prerequisites in order to use two-phase commit. 2. If you want to use DCE (Distributed Computing Environment) with Version 5.2 of the DB2 Universal Database products, you require a DCE product that is at OSF DCE level 1.1, which is provided by the AIX Version 4.1.4 operating system with its latest DCE PTF. For DB2 Connect support, you require DB2/MVS Version 5.1 plus its prerequisite, OS/390 DCE Base Services Version 3 for DCE support. 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. 3. 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. 4. If you want to use the Simple Network Management Protocol (SNMP) subagent, you require DPI 2.0 provided by IBM Netfinity Agent.

Table 3 (Page 2 of 3). Software Requirements for DB2 Servers

Product	Hardware/Operating System	Communications
• DB2 Universal Database Enterprise Edition (including DB2 Connect Enterprise Edition)	HP 9000 series 700 or 800 system and the following: • HP-UX Version 10.10 or later The following patches are required: • For HP-UX Version 10.20 – PHSS_10556 – PHSS_10436 – PHSS_10053 – PHSS_10113 • For HP-UX Version 10.10 – PHSS_10437 – PHSS_10053 – PHSS_10113 • For systems with the ANSI C or C++ compilers: – PHSS_10261 – PHSS_7505 – PHSS_9096 for C++	APPC, IPX/SPX, or TCP/IP • For APPC connectivity, HP-UX Version 10.10 and 10.20 require one of the following: – SNAplus and the following components: - SNAplusLink Version A.10.10 - SNAplusAPI Version A.10.10 OR – SNAplus2 and the following components: - SNAplus2Link Version A.10.10 - SNAplus2API Version A.10.10 **Note:** HP-UX does not provide support for inbound client APPC requests. • TCP/IP is provided with the HP-UX base operating system. • For IPX/SPX connectivity, you require Netware Version 3.12 for the HP 9000. **Notes:** 1. If you want to use DCE (Distributed Computing Environment) with Version 5.2 of the DB2 Universal Database products, you require a DCE product that is at OSF DCE level 1.1, which is provided by the HP-UX Version 10 and later operating system. For DB2 Connect support, you require DB2/MVS Version 5.1 plus its prerequisite, OS/390 DCE Base Services Version 3 for DCE support. 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. 2. 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.8.
• DB2 Universal Database Enterprise Edition (including DB2 Connect Enterprise Edition)	HP 9000 series 700 or 800 system and the following: • HP-UX Version 11.00 or later	APPC, IPX/SPX, or TCP/IP • For APPC connectivity, HP-UX Version 11.00 requires the following: – SNAplus2 Link R6.11.00.00 – SNAplus2 API R.6.11.00.00 **Note:** HP-UX does not provide support for inbound client APPC requests. • TCP/IP is provided with the HP-UX base operating system. • For IPX/SPX connectivity, you require Netware for HP-UX 11. **Notes:** 1. If you want to use DCE (Distributed Computing Environment) with Version 5.2 of the DB2 Universal Database products, you require a DCE product that is at OSF DCE level 1.1, which is provided by the HP-UX Version 11 and later operating system. 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.

Table 3 (Page 3 of 3). Software Requirements for DB2 Servers

Product	Hardware/Operating System	Communications
• DB2 Universal Database Enterprise Edition (including DB2 Connect Enterprise Edition)	SCO UnixWare 7	APPC, IPX/SPX, or TCP/IP • For APPC connectivity, you require SCO UnixWare 7 and IBM Communication Server. • The SCO UnixWare 7 base operating system provides TCP/IP and IPX/SPX connectivity **Note:** DB2 Connect for SCO requires SCO UnixWare 7 and IBM eNetwork Communications Server Version 5.
• DB2 Universal Database Enterprise Edition (including DB2 Connect Enterprise Edition)	Solaris SPARC-based computer and the following: • Solaris Version 2.5.1 or later The following patches are required for Solaris version 2.5.1: • 101242 Rev.11 or higher • 103566 Rev.08 or higher • 103600 Rev.13 or higher • 103640 Rev.20 or higher	APPC, IPX/SPX, or TCP/IP • For APPC connectivity, you require SunLink SNA 9.0 and the following communication products: – SunLink P2P LU6.2 9.0 – SunLink PU2.1 9.0 – SunLink P2P CPI–C 9.0 • TCP/IP is provided with the Solaris base operating system. • IPX/SPX connectivity is provided with SolarNet PC Protocol Services 1.1 with IPX/SPX for the Solaris 2.x Operating Environment. **Notes:** 1. DB2 Connect for Solaris requires Solaris Version 2.5.1 or later and the following communication products: • SunLink P2P LU6.2 9.0 • SunLink SNA PU2.1 9.0 Server • SunLink P2P CPI-C 9.0 2. If you want to use DCE (Distributed Computing Environment) with Version 5.2 of the DB2 Universal Database products, you require Transarc DCE Version 1.1 for Solaris 2.5 and 2.5.1, patch level 18 or higher. 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. 3. 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.8.

Client Product Requirements

Table 4 on page 31 lists the software requirements needed for the DB2 client products.

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

Component	Hardware/Operating System	Communications
• DB2 Client Application Enabler for AIX • DB2 Software Developer's Kit for AIX • Net.Data for AIX	RISC System/6000 and the following: • AIX Version 4.1.4 or later	APPC or TCP/IP • For APPC connectivity, you require IBM Communications Server Version 4.1.4 or later for AIX • The AIX base operating system provides TCP/IP connectivity, if selected during install. **Notes:** 1. If you want to use DCE (Distributed Computing Environment), you require a DCE product that is provided by the AIX Version 4.1.4 operating system with its latest DCE PTF. 2. DB2 for AIX also has a server IPX communications manager.
• DB2 Client Application Enabler for HP-UX Version 10.10 or later • DB2 Software Developer's Kit for HP-UX	HP 9000 Series 700 or 800 system and the following: • HP-UX Version 10.10 or later The following patches are required: • For HP-UX Version 10.20 – PHSS_10556 – PHSS_10436 – PHSS_10053 – PHSS_10113 • For HP-UX Version 10.10 – PHSS_10437 – PHSS_10053 – PHSS_10113 • For systems with the ANSI C or C++ compilers: – PHSS_10261 – PHSS_7505 – PHSS_9096 for C++	APPC or TCP/IP • For APPC connectivity, HP-UX Version 10.10 and 10.20 require one of the following: – SNAplus and the following components: - SNAplusLink Version A.10.10 - SNAplusAPI Version A.10.10 OR – SNAplus2 and the following components: - SNAplus2Link Version A.10.10 - SNAplus2API Version A.10.10 • The HP-UX base operating system provides TCP/IP connectivity. **Notes:** 1. DB2 for HP-UX has a server IPX communications manager 2. If you want to use DCE (Distributed Computing Environment), you require a DCE product that is provided by the HP-UX Version 10 or later base operating system.
• DB2 Client Application Enabler for HP-UX Version 11 • DB2 Software Developer's Kit for HP-UX	HP 9000 Series 700 or 800 system and the following: • HP-UX Version 11.00 or later	APPC or TCP/IP • For APPC connectivity, you require either of the following: – SNAplus2 Link R6.11.00.00 – SNAplus2 API R6.11.00.00 • The HP-UX base operating system provides TCP/IP connectivity. **Notes:** 1. DB2 for HP-UX has a server IPX communications manager 2. If you want to use DCE (Distributed Computing Environment), you require a DCE product that is provided by the HP-UX Version 11 or later base operating system.
• DB2 Client Application Enabler for Macintosh • DB2 Software Developer's Kit for Macintosh	• System 7.5 or later	TCP/IP • For TCP/IP connectivity, you require MacTCP Version 2.0.6. **Note:** If you are running in a 68K environment, you need to install Apple Shared Library Manager (ASLM) Version 2.0 or later.

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

Component	Hardware/Operating System	Communications
• DB2 Client Application Enabler for OS/2 • DB2 Software Developer's Kit for OS/2 • Net.Data for OS/2	• 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	APPC, IPX/SPX, Named Pipes (Local), NetBIOS, or TCP/IP • 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. **Notes:** 1. OS/2 Warp Server provides the Netfinity LAN client that Netfinity LAN needs for installing clients and servers. 2. Net.Data requires OS/2 Warp Version 3 or later and a Web server such as IBM's Internet Connection Server. 3. 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.
• DB2 Client Application Enabler for SCO OpenServer • DB2 Software Developer's Kit for SCO OpenServer • Net.Data for SCO OpenServer	• SCO OpenServer Version 5.0.2 or later	APPC, IPX/SPX, or TCP/IP • For APPC connectivity, you require Interface Systems CLEO SNA Version 4.1.1. • The SCO OpenServer base operating system provides IPX/SPX and TCP/IP connectivity.
• DB2 Client Application Enabler for SCO UnixWare 7 • DB2 Software Developer's Kit for SCO UnixWare 7 • Net.Data for SCO UnixWare 7	• SCO UnixWare 7	APPC, IPX/SPX, or TCP/IP • For APPC connectivity, you require: • The SCO UnixWare 7 base operating system provides IPX/SPX and TCP/IP connectivity.
• DB2 Client Application Enabler for Silicon Graphics IRIX • DB2 Software Developer's Kit for Silicon Graphics IRIX	• Silicon Graphics IRIX, Version 6.x, and the following filesets: – eoe.sw.oampkg – eoe.sw.svr4net	TCP/IP **Note:** The Silicon Graphics IRIX base operating system provides TCP/IP connectivity.
• DB2 Client Application Enabler for SINIX • DB2 Software Developer's Kit for SINIX	Siemens Nixdorf RM200 or RM400 or RM600 system and the following: • SINIX-N or SINIX-Y Version 5.42 or later, • Reliant UNIX	APPC or TCP/IP • For APPC connectivity, you require the following: – SINIX TRANSIT-SERVER V3.3, SNA Communication Server – SINIX TRANSIT-CLIENT V3.3 SNA Communication Server / Local Function – SINIX TRANSIT-CPIC V3.3 SNA LU6.2 Communication and CPI-C. • The SINIX base operating system provides TCP/IP connectivity.

Table 4 (Page 3 of 4). Software Requirements for Clients

Component	Hardware/Operating System	Communications
• DB2 Client Application Enabler for Solaris • DB2 Software Developer's Kit for Solaris • Net.Data for Solaris	Solaris SPARC-based computer and the following: • Solaris Version 2.5.1 or later The following patches are required with Version 2.5.1: • 101242 Rev.11 or higher • 103566 Rev.08 or higher • 103600 Rev.13 or higher • 103640 Rev.20 or higher	APPC or TCP/IP • For APPC connectivity, you require SunLink SNA 9.0 and the following: – SunLink P2P LU6.2 9.0 – SunLink PU2.1 9.0 – SunLink P2P CPI–C 9.0 • The Solaris base operating system provides TCP/IP connectivity. **Notes:** 1. If you want to use DCE (Distributed Computing Environment), you require a DCE product with Transarc DCE Version 1.1 for Solaris 2.5 and 2.5.1, patch level 18 or higher. 2. DB2 for Solaris also has a server IPX communications manager.
• DB2 Client Application Enabler for Windows 3.x • DB2 Software Developer's Kit for Windows 3.x	• Microsoft Windows 3.x	IPX/SPX, NetBIOS, or TCP/IP • 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: – 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.
• 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	• 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	NetBIOS, IPX/SPX, TCP/IP, or Named Pipes (Local) • 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.
• DB2 Client Application Enabler for Windows 95 • DB2 Software Developer's Kit for Windows 95	• Windows 95 4.00.950 or later	IPX/SPX, Named Pipes, NetBIOS, or TCP/IP • The Windows 95 base operating system provides NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. **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.

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

Component	Hardware/Operating System	Communications
• DB2 Client Application Enabler for Windows 98 • DB2 Software Developer's Kit for Windows 98	• Windows 98	IPX/SPX, Named Pipes, NetBIOS, or TCP/IP • The Windows 98 base operating system provides NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. **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.
• DB2 Client Application Enabler for Windows NT • DB2 Software Developer's Kit for Windows NT • Net.Data for Windows NT	• Windows NT Version 3.51 • Windows NT Version 4.0 or later	APPC, IPX/SPX, Named Pipes, NetBIOS, or TCP/IP • 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: – 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 **Note:** If you have the IBM Antivirus program installed on your operating system, it must be Version 3.0 or later.

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

	DB2 Servers							
	AIX	**HP-UX**	**OS/2**	**SCO UnixWare 7**	**Solaris**	**Windows NT**	**SCO OpenServer**	**SINIX**
AIX	APPC TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP	APPC TCP/IP	TCP/IP
HP-UX	APPC TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP	APPC TCP/IP	TCP/IP
Macintosh	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
OS/2	APPC IPX/SPX(1),(2) TCP/IP	IPX/SPX(1),(2) TCP/IP	APPC IPX/SPX(1),(2) NetBIOS TCP/IP	IPX/SPX(1), TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) NetBIOS TCP/IP	APPC IPX/SPX(1) TCP/IP	TCP/IP
SCO OpenServer	APPC IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	TCP/IP
SCO UnixWare 7	APPC IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	TCP/IP

Table 5 (Page 2 of 2). Possible Client-to-Server Connectivity Scenarios

	DB2 Servers							
	AIX	**HP-UX**	**OS/2**	**SCO UnixWare 7**	**Solaris**	**Windows NT**	**SCO OpenServer**	**SINIX**
Silicon Graphics IRIX	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
SINIX	APPC TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP	APPC TCP/IP	TCP/IP
Solaris	APPC TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP	APPC TCP/IP	TCP/IP
Windows 3.x	IPX/SPX(1),(2) TCP/IP	IPX/SPX(1),(2) TCP/IP	IPX/SPX(1),(2) NetBIOS TCP/IP Local Named Pipes	IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	IPX/SPX(1) NetBIOS TCP/IP	IPX/SPX(1) TCP/IP	TCP/IP
Windows 95 or Windows 98	IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	IPX/SPX(1) NetBIOS TCP/IP	IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	IPX/SPX(1) Named Pipes NetBIOS TCP/IP	IPX/SPX(1) TCP/IP	TCP/IP
Windows NT	APPC IPX/SPX(1) TCP/IP	IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) NetBIOS TCP/IP	IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) Named Pipes NetBIOS TCP/IP	APPC IPX/SPX(1) TCP/IP	TCP/IP

1 Direct Addressing
2 File Server Addressing

Note: SCO OpenServer and SINIX are DB2 Version 2 only.

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 primary group of the instance owner 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 142.

Preparing to Install the DB2 Products

Before you install DB2 Universal Database products, it is important to know which products and components you must install and which are optional. “Selecting Products and Components” on page 36 lists these DB2 products and their components.

Selecting Products and Components

Table 6 lists the DB2 product components that you can select to install on your system.

Table 6 (Page 1 of 2). Components for DB2 Products

Product / Component Description	**DB2 Client Application Enabler**	**DB2 Universal Database Workgroup Edition**	**DB2 Universal Database Enterprise Edition**	**DB2 Connect**
DB2 Client Application Enabler	√√	√√	√√	√√
Open Database Connectivity (ODBC) support	√	√	√	√
Java Support (JDBC)	√	√	√	√
DB2 Web Control Center support	√	√	√	√
Administration Server	n/a	√√	√√	√√
DB2 Run-time Environment	n/a	√√	√√	√√
DB2 Sample Database Source	n/a	√√	√√	n/a
DB2 Engine	n/a	√√	√√	n/a
DB2 Connect	n/a	√√	√√	√√
Communication Support for TCP/IP	n/a	√√	√√	√√
Communication Support for IPX/SPX	n/a	√√	√√	√√
Communication Support for SNA	n/a	√√	√√	√√
Communication Support for DRDA Application Server	n/a	√√	√√	√√
Code Page Conversion Tables - Japanese	n/a	√	√	√
Code Page Conversion Tables - Korean	n/a	√	√	√
Code Page Conversion Tables - Simplified Chinese	n/a	√	√	√
Code Page Conversion Tables - Traditional Chinese	n/a	√	√	√
Licensed Support for DB2 Universal Database Workgroup Edition	n/a	√√	n/a	n/a
License Support for DB2 Universal Database Enterprise Edition	n/a	n/a	√√	n/a
Licensed Support for DB2 Connect	n/a	n/a	n/a	√√
DB2 Replication	n/a	√	√	√
DB2 Product Messages (non-English)[1]	o	o	o	o
DB2 Product Library (HTML)[1]	o	o	o	o

Table 6 (Page 2 of 2). Components for DB2 Products

Product / Component Description	DB2 Client Application Enabler	DB2 Universal Database Workgroup Edition	DB2 Universal Database Enterprise Edition	DB2 Connect

√√	This is a required component that must be installed.
√	This is a recommended component that is installed by default. You can choose not to install this component.
o	This is an optional component that is not installed by default. If you want to install it, you must select it.
n/a	This component is not available for installation.

Notes:

1. There is a separate component of the DB2 Product Messages and the DB2 Product Library for each locale.
2. The DB2 SDK is available only with the SDK Pack. Refer to “Developing Applications Using the DB2 Software Developer's Kit” on page 13.

Chapter 3. Installing and Configuring DB2

If you need to migrate databases that were created using previous versions of DB2, you must complete certain procedures after installing Version 5 of DB2. See Chapter 6, “Migrating from Previous Versions and Releases” on page 57 for further information.

We recommend that you install and configure DB2 Universal Database products using the DB2 Installer program. Other methods to install DB2 Universal Database products are also supported.

This section describes how to use the DB2 Installer program. For other advanced installation and configuration methods, refer to the following:

- Chapter 21, “Other Methods to Install DB2 for AIX” on page 203.
- Chapter 22, “Other Methods to Install DB2 for HP-UX” on page 213.
- Chapter 23, “Other Methods to Install DB2 for Solaris” on page 225.
- Chapter 24, “Other Methods to Install SCO UnixWare 7” on page 235.

For information on how to install the DB2 clients, refer to Chapter 7, “Installing DB2 Clients” on page 79.

Before You Begin

Before you begin installing DB2 products using the DB2 Installer program, you need to gather the following information:

Where is the CD-ROM mount point?
You need to mount the CD-ROM before you can install the DB2 product. To mount the CD-ROM, you must know where the mount point is. For example, the CD-ROM mount point could be `/cdrom`.

Note: On the Solaris operating system, the CD-ROM is automatically mounted if the Volume Manager (vold) is running.

Which DB2 products should I install?
Table 6 on page 36 lists the required and optional components for all of the DB2 products available. It also shows the recommended components that are installed by default.

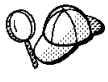
How much disk space does my operating system require for all selected products?
See “Estimating Fixed Disk Requirements” on page 25 and “Server Components” on

page 25 to find out how much disk space is required for selected products and how to estimate disk space requirements.

How do I assign a group name and user name for a DB2 instance?
A separate user name must be used for each DB2 Instance. We recommend that a new group be created which should be used as a primary group for the DB2 instance user.

Note: You will also need another dedicated user name and group name for fenced user defined functions (UDFs) and stored procedures. For security reasons, we recommend that you do not use the DB2 instance user name and group name for the fenced UDFs and stored procedures.

How do I assign a group name and user name for the Administration Server?
You need a dedicated group name and user name for the Administration Server. For security reasons, we recommend that you do not use the DB2 Instance user name and group name for the Administration Server.



When you use the DB2 Installer program, you should be aware of the following:

- The DB2 Installer's **db2setup** command only works with Bourne and Korn shells. Other shells are not supported.
- You can generate a trace log, *db2setup.trc*, to record errors experienced during the installation. Run the **db2setup** command as follows:

  ```
  db2setup -d
  ```

  This creates a trace file, `/tmp/db2setup.trc`.

Installation Steps

To install the DB2 products, perform the following steps:

1. Identify and record parameter values.
2. Update kernel configuration parameters. (This step is not required on AIX.)
3. Mount the CD-ROM.
4. Install DB2 products.

Step 1. Identify and Record Parameter Values

Table 7 will help you determine the values required to install DB2 products, set up a DB2 instance and configure the Administration Server. Before proceeding with the installation and configuration, complete the *Your Value* column in the table. If you want to choose the default value for a parameter, you do not need to provide any value in the *Your Value* column for that parameter. In Table 7, the only parameter for which a value is required is *DB2 Product Name*. All other parameters either have a default value or are optional.

Table 7 (Page 1 of 2). Parameter Values Required for Installation

Information Required for DB2 Installer	Default Value	Your Value
Product/Component		
• DB2 Product Name[1]	None	
• DB2 Product Messages[2]	None	
• Documentation[2]	None	
DB2 Instance		
• User Name	db2inst1	
• UID	System-generated UID	
• Group Name	db2iadm1	
• GID	System-generated GID	
• Password	ibmdb2	
• TCP/IP Service Name	db2cdb2inst10	
• TCP/IP Port Number	50000	
• IPX/SPX File Server Name	*	
• IPX/SPX Object Name	*	
• IPX/SPX Socket Number	879E	
• IPX/SPX NetWare User ID	None	
• IPX/SPX NetWare Password	None	
• User Name (UDF)	db2fenc1	
• UID (UDF)	System-generated UID	
• Group Name (UDF)	db2fadm1	
• GID (UDF)	System-generated GID	
• Password (UDF)	ibmdb2	
Administration Server		

Table 7 (Page 2 of 2). Parameter Values Required for Installation

Information Required for DB2 Installer	Default Value	Your Value
• User Name (DAS)	db2as	
• UID (DAS)	System-generated UID	
• Group Name (DAS)	db2asgrp	
• GID (DAS)	System-generated GID	
• Password (DAS)	ibmdb2	
• TCP/IP Port Number (DAS)	523	523
• IPX/SPX File Server Name (DAS)	*	
• IPX/SPX Object Name (DAS)	*	
• IPX/SPX Socket Number (DAS)	87A2	87A2
• IPX/SPX NetWare User ID (DAS)	None	
• IPX/SPX NetWare Password (DAS)	None	

Notes:

1. You must select at least one DB2 product to install. See Table 6 on page 36 for a list of DB2 products available for installation.
2. You can optionally select one or more filesets in this product. There is a separate fileset for each locale. See Appendix B, “Contents of the DB2 Products” on page 309 for the names of filesets for DB2 Product Messages and Documentation.

To continue the installation on AIX systems, proceed to “Step 3. Mount the CD-ROM” on page 46.

Step 2. Updating Kernel Configuration Parameters

Depending on your workstation's operating system and its kernel configuration, you may have to update the kernel configuration parameters.

This step is not required on AIX.

Recommended Values for HP-UX Version 10 and Version 11

The values in Table 8 are recommended for the HP-UX kernel configuration parameters.

Table 8. HP-UX Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Physical Memory		
	64MB - 128MB	**128MB - 256MB**	**256MB+**
maxuprc	256	384	512
maxfiles	256	256	256
nproc	512	768	1024
nflocks	2048	4096	8192
ninode	512	1024	2048
nfile	(4 * ninode)	(4 * ninode)	(4 * ninode)
msgseg	8192	16384	32768
msgmnb	65535 (1)	65535 (1)	65535 (1)
msgmax	65535 (1)	65535 (1)	65535 (1)
msgtql	256	512	1024
msgmap	130	258	258
msgmni	128	256	256
msgssz	16	16	16
semmni	128	256	512
semmap	130	258	514
semmns	256	512	1024
semmnu	256	512	1024
shmmax	67108864	134217728 (2)	268435456 (2)
shmseg	16	16	16
shmmni	300	300	300

Notes:

1. Parameters *msgmnb* and *msgmax* must be set to 65535.
2. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in the preceding table.
3. Parameter *shmmax* should be set to 134217728 or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set *shmmax* to 184968806 (176*1024*1024).

To continue with the installation on HP-UX systems, proceed to “Step 3. Mount the CD-ROM” on page 46.

Recommended Values for SCO UnixWare 7

The values in Table 9 are recommended for SCO UnixWare 7 kernel configuration parameters.

Table 9. SCO UnixWare 7 Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Recommended Value
msgmax	65535 (1)
msgmnb	65535 (1)
msgssz	524288
msgmni	256
shmmax	268435456
shmmni	300
shmseg	15
semmi	1024

Notes:

1. Parameters *msgmax* and *msgmnb* should be set at least to 65535.
2. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in the preceding table.
3. Parameter *shmmax* should be set to the suggested value in the above table or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set *shmmax* to 184968806 (176*1024*1024).

To change a value, do the following:

1 Enter the **scoadmin** command to start the System Administration tool.

2 Double-click on the **System** folder.

3 Double-click on the **System Tuner** icon.

4 Click on the drop down box and select the **Inter-Process Communication (IPC) Parameters**.

5 Select the parameter to be changed and enter the new value.

6 Click on **OK** when you have finished changing all the parameters.

7 Click on the **Yes** push button to rebuild the kernel.

8 Reboot the system so that the changes can take effect.

Recommended Values for Solaris

The values in Table 10 are recommended for Solaris kernel configuration parameters.

Table 10. Solaris Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Physical Memory			
	64MB - 128MB	128MB - 256MB	256MB - 512MB	512MB+
msgsys:msginfo_msgmax	65535 (1)	65535 (1)	65535 (1)	65535 (1)
msgsys:msginfo_msgmnb	65535 (1)	65535 (1)	65535 (1)	65535 (1)
msgsys:msginfo_msgmap	130	258	258	258
msgsys:msginfo_msgmni	128	256	256	256
msgsys:msginfo_msgssz	16	16	16	16
msgsys:msginfo_msgtql	256	512	1024	1024
msgsys:msginfo_msgseg	8192	16384	32768	32768
shmsys:shminfo_shmmax	67108864	134217728 (2)	268435456 (2)	536870912 (2)
shmsys:shminfo_shmseg	16	16	16	16
shmsys:shminfo_shmmni	300	300	300	300
semsys:seminfo_semmni	128	256	512	1024
semsys:seminfo_semmap	130	258	514	1026
semsys:seminfo_semmns	256	512	1024	2048
semsys:seminfo_semmnu	256	512	1024	2048

Notes:

1. The *msgsys:msginfo_msgmnb* and *msgsys:msginfo_msgmax* parameters must be set to 65535.
2. The *shmsys:shminfo_shmmax* parameters should be set to the suggested value in the above table, or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set the *shmsys:shminfo_shmmax* parameter to 184968806 (176*1024*1024).

To set a kernel parameter, add a line at the end of the `/etc/system` file as follows:

```
set parameter_name = value
```

For example, to set the value of the *msgsys:msginfo_msgmax* parameter, add the following line to the end of the `/etc/system` file:

```
set msgsys:msginfo_msgmax = 65535
```

Sample files for updating the kernel configuration parameters are provided in the `/opt/IBMdb2/V5.0/cfg` directory. The names for these files are as follows:

- `kernel.param.64MB` for systems with 64MB–124MB of physical memory
- `kernel.param.128MB` for systems with 128MB–256MB of physical memory
- `kernel.param.256MB` for systems with more than 256MB–512MB of physical memory
- `kernel.param.512MB` for systems with more than 512MB of physical memory

Depending upon the amount of physical memory in your system, append the appropriate kernel configuration parameter file to the `/etc/system` file. If necessary, change the value of the *shmsys:shminfo_shmmax* parameter as described in Note 2 above.

After updating the `/etc/system` file, reboot the system.

Step 3. Mount the CD-ROM

To install DB2 products using the DB2 Installer program, you must first mount the CD-ROM. Once you have mounted the CD-ROM, you can start installing DB2 Universal Database.

Mounting on AIX Systems

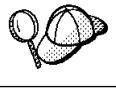
Perform the following steps to mount the CD-ROM on AIX operating systems:

1. Log in as user with root authority.
2. Insert the DB2 Client Application Enablers CD-ROM in the drive.
3. Create a directory to mount the CD-ROM by entering the following command:

   ```
   mkdir -p /cdrom
   ```

   where `cdrom` is the CD-ROM mount directory.
4. Allocate a CD-ROM file system by entering the following command:

   ```
   smitty storage
   ```
5. Select **File Systems**.
6. Select **Add / Change / Show / Delete File Systems**.
7. Select **CDROM File Systems**.
8. Select **Add CDROM File System**.
9. Select **Device Name**.



Device names for CD-ROM file systems must be unique. If there is a duplicate device name, you may need to delete a previously-defined CD-ROM file system or use another name for your directory.

10. In the pop-up window, enter `/cdrom` as the **mount point**.

11. Mount the CD-ROM file system by entering the following command:

    ```
    smit mountfs
    ```

12. Select the **FileSystem** name. For example, the name could be `/dev/cd0`.
13. Select the **Directory** name, `/cdrom`.
14. Select the **Type of filesystem**, `cdrfs`.
15. Set the **Mount as READ-ONLY system** to `Yes`.
16. Log off.

After mounting the CD-ROM, proceed to "Step 4. Install the DB2 Products" on page 48.

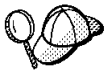
Mounting on HP-UX Systems

Perform the following steps to mount the CD-ROM on HP-UX operating systems:

1. Log in as user with root authority.
2. Insert the CD-ROM in the drive and mount it as in the following example:

   ```
   mkdir /cdrom
   /usr/sbin/mount /dev/dsk/c0t2d0 /cdrom
   ```

   where `/cdrom` is the CD-ROM mount directory.
3. Log out.



The CD-ROM may also be mounted using the System Administration (**SAM**) tool. Consult your HP-UX documentation for more information about **SAM**.

After mounting the CD-ROM, proceed to "Step 4. Install the DB2 Products" on page 48.

Mounting on SCO UnixWare 7 Systems

Perform the following steps to mount the CD-ROM on SCO UnixWare 7 operating systems:

1. Log in as user with root authority.

2. Insert the CD-ROM in the drive and mount it as in the following example:

   ```
   mount -F cdfs -o ro /dev/cdrom/c0b0t5l0 /cdrom
   ```

   where `c0b0t5l0` is the device name found under the `/dev/cdrom` directory and varies with the particular hardware on the system.

3. Log out.

After mounting the CD-ROM, proceed to "Step 4. Install the DB2 Products."

Mounting on Solaris Systems

Perform the following steps to mount the CD-ROM on Solaris operating systems:

1. Log in as user with root authority.
2. If the Volume Manager is not running on your system, enter the following commands to mount the CD-ROM:

   ```
   mkdir -p /cdrom/unnamed_cdrom
   mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom/unnamed_cdrom
   ```

   where */cdrom/unnamed_cdrom* is the CD-ROM mount directory.

If you are mounting the CD-ROM drive from a remote system using NFS, the CD-ROM file system on the remote machine must be exported with *root* access. You must also mount that file system with *root* access on the local machine.

   If the Volume Manager (vold) is running on your system, the CD-ROM is automatically mounted as:

   ```
   /cdrom/unnamed_cdrom
   ```

3. Log out.

Step 4. Install the DB2 Products

After you mount the CD-ROM file system, use the DB2 Installer program to install the DB2 products.

Note: DB2 Version 5 products can co-exist on the same machine with previous versions of DB2.

If you are installing the DB2 Client Application Enabler from a remote server, it is better to use the **telnet** command to open a telnet session instead of using the **rlogin** command to connect to your remote server.

1. Log in as root.
2. Insert the DB2 CD-ROM in the drive.
3. If required, mount the CD-ROM. Refer to “Step 3. Mount the CD-ROM” on page 46 if required.
4. Change to the directory where the CD-ROM is mounted by entering the following command:

   a. On AIX, HP-UX, or SCO UnixWare 7:

      ```
      cd /cdrom
      ```

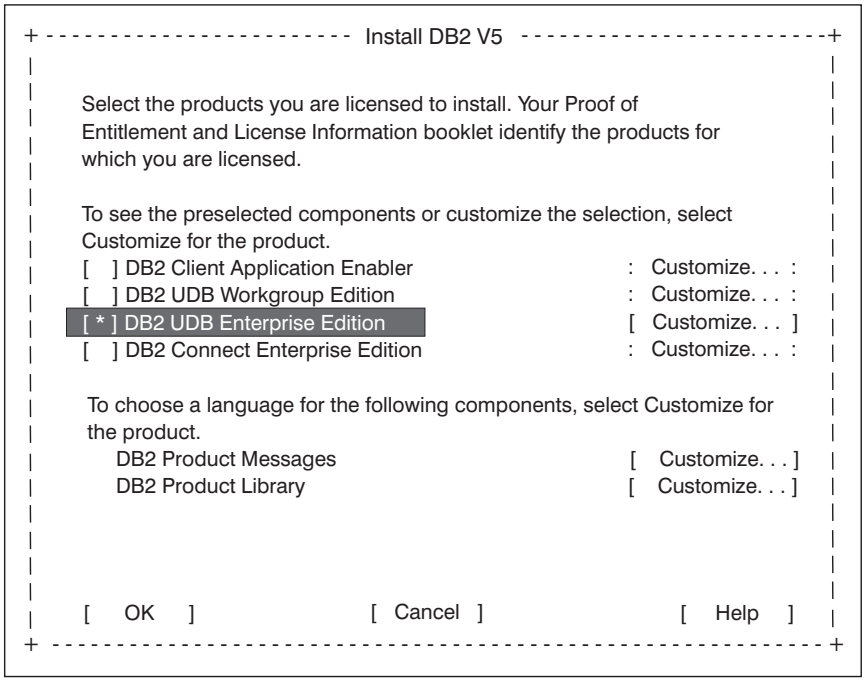
   b. On Solaris:

      ```
      cd /cdrom/unnamed_cdrom
      ```

   where `/cdrom` is the mount point of the CD-ROM drive on AIX, HP-UX, or SCO UnixWare 7, and `/cdrom/unnamed_cdrom` is the mount point of the CD-ROM on Solaris.

5. Enter the **./db2setup** command to start the DB2 Installer program. The Install DB2 V5 screen opens.

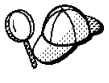
It will take some time for the DB2 Installer program to start up, as it is scanning your system for information.



Note: Your screen will be different if you have already installed DB2 Version 5 products on your system. See Chapter 4, “Installing Additional Products and Creating Additional Instances” on page 53 for more information.

6 From the product list on the Install DB2 V5 screen, select the products that you want to install.

To display the required and optional components for the product you want to install, select **Customize**. The optional components that are most typically used are pre-selected for you. Make your selections and select **OK**.

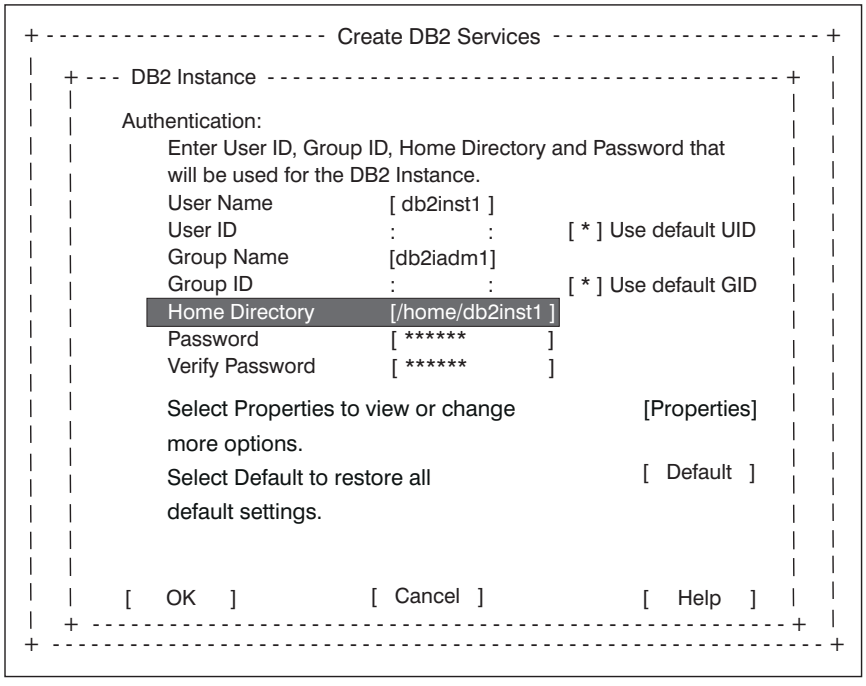


To refresh the current screen, press the F5 key or Ctrl+L.

When you have finished selecting the DB2 product and its components, select **OK**. The Create DB2 Services window opens. To undo any selections you made, select **Cancel**.

Go to step 9 on page 52 if you do not want to create a DB2 Instance at this time.

7 At the Create DB2 Services panel, select **Create a DB2 Instance** to display the DB2 Instance screen.



You can use the default values displayed at the DB2 Instance screen or change them:

- To change the default **User ID** for the instance, deselect **Use Default UID** and enter a new value for the **User ID**.
- To configure the communication protocol(s), select **Properties**. Only the protocols you selected from the product component lists will display on the DB2 Instance Protocol screen. For example, if you selected TCP/IP and wanted to change the protocol properties, select **Properties**. At the TCP/IP screen, enter a new Service Name and Port Number. To restore all system-generated values, select **Default**.
- To create a sample database for the DB2 Instance, select **Properties** and then select **Create a sample database for DB2 Instance**. The sample database requires a minimum of 15 MB.

8 When you have finished entering the values for the DB2 Instance, select **OK**.

You will have to enter values for the User ID, Group ID and Password for the fenced user defined Functions (UDFs) and stored procedures. The User Name for the fenced UDFs should be different from the User Name of the DB2 Instance for security reasons.

Go to step 11 on page 52 if you do not want to create an Administration Server at this time.

9. At the Create DB2 Services screen, select **Create an Administration Server** to display the Administration Server screen.

   You can use the default values on the Administration Server screen or change them:

   - To change the default **User ID** for the server, deselect **Use default UID** and enter a new value for **User ID**.
   - To configure the communication protocol(s), select **Properties**. Only the protocols you selected from the product component lists will display on the Administration Server Protocol screen. To restore all system-generated values, select **Default**.

10. When you have finished entering the values for the Administration Server, select **OK**.

11. After entering the values for the DB2 Instance and the Administration Server, select **OK** to display the Summary Report. The report lists the installable items as well as the location of the installation log file.

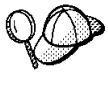
12. Use the Up or Down arrow keys to review the Summary Report. If the selections are not correct, press the **F3** key to return to the previous screen. Otherwise, select **Continue** to start the installation.

13. When the installation is complete, use the Up or Down arrow keys to review the Status Report. Select **View Log** to view the installation log file. Select **OK** to close the Status Report screen.

14. Select **Close** from the DB2 Installer screen to terminate the DB2 Installer program.

After the installation is complete, the software is installed in the `DB2DIR` directory,

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris



You can use the DB2 Installer program to create additional DB2 instances after you have installed DB2 on your system. To start the DB2 Installer program from the DB2 product directory, type the following command:

On AIX

```
/usr/lpp/db2_05_00/install/db2setup
```

On HP-UX, Solaris, or SCO UnixWare 7

```
/opt/IBMdb2/V5.0/install/db2setup
```

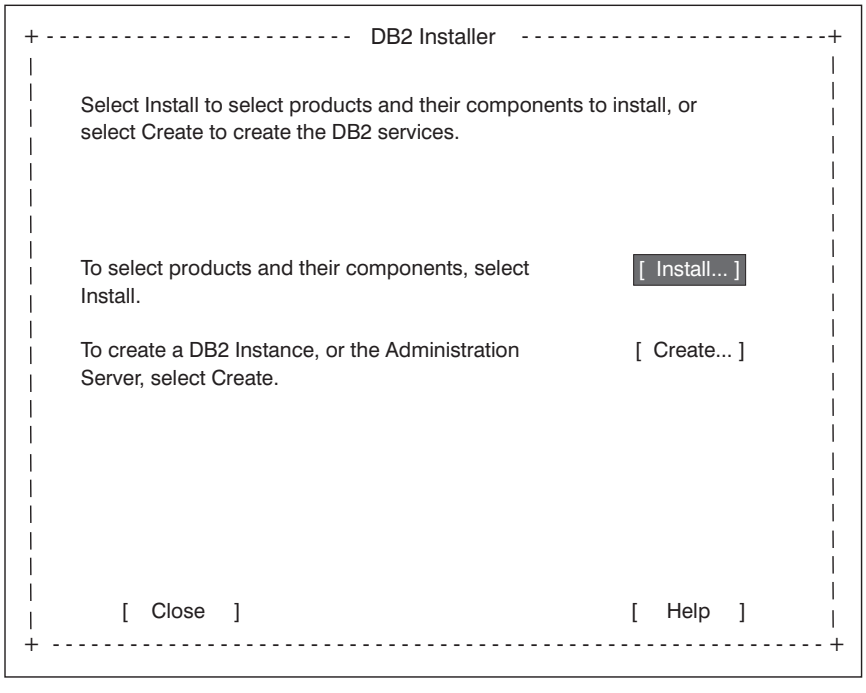
Go to Chapter 4, “Installing Additional Products and Creating Additional Instances” on page 53 for information on optional installation activities or proceed directly to Chapter 6, “Migrating from Previous Versions and Releases” on page 57.

Chapter 4. Installing Additional Products and Creating Additional Instances

The installation activities in this section are optional following the installation of DB2 Version 5 on your system. It describes what you need to do if you plan to:

- Install additional products.
- Create additional DB2 Instances.
- Create an Administration Server (if required and not created previously).

The next time you start the DB2 Installer program, after having installed DB2 Version 5, the main menu will look like the following screen.



To select additional products and their components for installation, select **Install** from the DB2 Installer screen. See step 6 on page 50 for instructions.

To create a DB2 instance or the Administration Server on your system, select **Create** from the DB2 Installer screen. See step 7 on page 50 and step 9 on page 52 for instructions.

To terminate the DB2 Installer program, select **Close**.

Chapter 5. Setup for Developing Applications with Micro Focus COBOL

If you intend to develop DB2 applications with the Micro Focus COBOL compiler and debugger (or Animator), you must add the DB2 generic API entry that points to the Micro Focus run-time library. To do this, you must enter the **mkrts** command that is supplied by Micro Focus as follows:

1. Log in as root.
2. Enter the **mkrts** command with the arguments supplied in the following directories:

   AIX
   `/usr/lpp/db2_05_00/lib/db2mkrts.args`

   HP-UX, Solaris or SCO UnixWare 7
   `/opt/IBMdb2/V5.0/lib/db2mkrts.args`

   For example:

   ```
   mkrts 'cat /usr/db2_05_00/lib/db2mkrts.args'
   ```

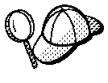
Chapter 6. Migrating from Previous Versions and Releases

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. It also discusses how to migrate between different levels of UNIX operating systems.

Migrating from Previous Versions of DB2

DB2 migration involves the following:

- "Pre-Migration" on page 58
- "Migrating Instances" on page 58.
- "Migrating Databases" on page 66.
- "Post Migration" on page 67.



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 71 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 71 for further information. You should also run the **dasiupdt** command to update the Administration Server. See "Updating the Administration Server" on page 159

Pre-Migration

To ensure that all databases on your system can be migrated to DB2 Version 5, perform the following steps:

1. Complete all database transactions.
2. Ensure all applications disconnect from each database.
3. Ensure all databases are cataloged.

The **db2ckmig** database migration verification command does not verify uncataloged databases. See "Verify that Databases Can Be Migrated" on page 60 for a description of how **db2ckmig** works.

4. Make a backup copy of all databases. Refer to the *Administration Guide* for your DB2 product for information on making a backup copy of a database 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.
6. Ensure that the instance environment setup script, **db2profile**, under the instance's `INSTHOME/sqllib/` directory is of the proper Bourne shell syntax.

   Ensure that each export statement is separated into different lines. For example:

   ```
   DB2INSTANCE=db2inst1
   export DB2INSTANCE
   ```

Migrating Instances

This procedure describes how to migrate DB2 instances that were created using a previous version of DB2.

If you upgraded your installed DB2 product by applying a PTF or a FixPak, you should update instances, using the **db2iupdt** command, instead of migrating them. Instance updates do not apply to `DB2 Product Documentation` and `DB2 Product Messages`. See "Updating Instances" on page 154 for further information.

Before you can migrate an instance to use the latest version of DB2, you must install DB2 Version 5 on your system.

If there are several DB2 instances using previous versions of DB2, you do not need to migrate all of these instances at this time. Instances that are not migrated will continue to use previous versions of DB2.

Each DB2 instance must be migrated separately. To successfully migrate a DB2 instance, you need to perform the following steps:

1. Prepare the DB2 instance for migration.
2. Ensure that the user exit program can be migrated.
3. Migrate the DB2 instance.

If you want to migrate several instances, you must repeat these steps for each instance.

Prepare the DB2 Instance for Migration

Before you can migrate a DB2 instance, all applications using any databases owned by this instance must be completed. To prepare a DB2 instance for migration, you need to perform the following steps:

1. Log in as the DB2 instance owner.
2. Ensure that there are no applications using any databases owned by this DB2 instance. To get a list of all applications owned by the instance, enter the **db2 list applications** command.

   You can end a session by entering the **db2 terminate** command. It is not recommended to force termination of applications using the **db2 force applications all** command, since some applications may have unexpected behavior when terminated using this command. See the *Command Reference* for usage and details of this command.
3. When all applications are complete, stop all database server processes owned by the DB2 instance by entering the **db2stop** command.
4. Stop the DB2 license daemon by entering the **db2licd end** command.
5. Stop all command line processor sessions by entering the **db2 terminate** command in each session that was running the command line processor.
6. Enter the **db2_kill** command to clean up any remaining DB2 resources.
7. Log off.

The DB2 instance is now ready for migration.

Verify that Databases Can Be Migrated

DB2 provides the **db2ckmig** migration command. The **db2ckmig** command can be used to verify whether all cataloged databases can be migrated. The **db2imigr** command in "Migrate the DB2 Instance" on page 65 uses the **db2ckmig** command to verify whether the cataloged databases can be migrated.

To ensure that you can migrate the instance to DB2 Version 5, you should run the **db2ckmig** command. If instance migration failed, you must correct errors reported by this command. You can choose to run the **db2ckmig** command again to verify that the errors have been corrected and then migrate the instance.

For detailed information about the **db2ckmig** command, refer to the Version 5 *Command Reference*.

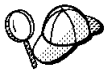
To verify that all cataloged databases can be migrated, you need to perform the following steps:

1. Log in as the instance owner.
2. Enter the following command:

   ```
   DB2DIR/bin/db2ckmig -h -a 0 -l INSTHOME/migration.log
   ```

   where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
   = /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

   and `INSTHOME` is the home directory of the instance and `migration.log` is the name for the output file.
3. Check the log file; for example, `INSTHOME/migration.log`.



The log file displays the errors that occur when you run the **db2ckmig** command. If it shows any errors, see Table 11 on page 61 for suggested corrective actions..

4. Check that the log file is empty before continuing with the instance migration.
5. Backup the database after making corrections.

Table 11 (Page 1 of 3). Correcting Error Messages

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. **Note:** This error does not apply to DB2 Parallel Edition or Version 1.x single-partition database.
A database is in an inconsistent state	Restart the database to return it to a consistent state.
The database contains database objects that have a schema name of SYSCAT, SYSSTAT, or SYSFUN	These schema names are reserved for the Version 5 database manager. To correct this error, do the following: **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.

Table 11 (Page 2 of 3). Correcting Error Messages

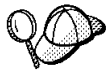
Error	Action
The database contains database objects that have a dependency on the SYSFUN.DIFFERENCE function. Possible violated database objects are: • Constraint • Function • Trigger • View	The SYSFUN.DIFFERENCE function must be dropped and recreated during database migration. However, if there is a database object that is dependent on this function, migration will fail. To correct this error, do the following: **Constraint** Enter the **alter table** command to drop the constraint. **Function** Enter the **drop function** command to drop the function dependent on SYSFUN.DIFFERENCE. **Trigger** Enter the **drop trigger** command to drop the trigger. **View** Enter the **drop view** command to drop the view. **Notes:** 1. 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. 2. This error does not apply to DB2 Parallel Edition or Version 1.x. single-partition database.

Table 11 (Page 3 of 3). Correcting Error Messages

Error	Action
The database contains user-defined distinct types that use the type name of DATALINK or REFERENCE.	The data type names, DATALINK and REFERENCE are reserved for the Version 5 database manager. To correct this error, do the following: **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. **Note:** This error does not apply to DB2 Parallel Edition or Version 1.x single-partition database.

All local databases now have the same authentication type as the instance where they reside; the authentication type in the database directory is ignored by DB2 Version 5.2 servers. If a warning is logged due to a conflicting authentication type, and you want a database to retain its previous authentication type, then you can do one of the following:

- Change the authentication type of the instance to the previous one.
- Move the database to another instance that has the required authentication type.



Before changing the authentication type of the instance, you should make sure that the new authentication type will be appropriate for all databases residing there. Be certain to consider the security implications of the different authentication types.

If there are databases that you do not want to migrate, you can uncatalog them (along with all aliases); the **db2ckmig** command does not perform any verification of uncataloged databases.

Refer to the *Administration Guide* for more information about the actions required to correct these conditions.

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 **db2uext2** in Version 5. (In previous versions, it was called **db2uexit**.)

The following should be considered before migrating instances:

- If the pre-Version 5 **db2uexit** program is installed in the `INSTHOME/sqllib/adm` directory before migration, it will remain in this directory after migration. The DB2 Version 5 **db2uext2** program will be also installed in this directory. Its function is to invoke **db2uexit** using the pre-Version 5 interface. This allows the old user exit program to be used on DB2 Version 5.
- If **db2uexit** is installed in a directory other than `INSTHOME/sqllib/adm`, it will not be installed after migration. For example, if **db2uexit** was in the `INSTHOME/sqllib/bin` directory, after migration the **db2uexit** file will not be in the `INSTHOME/sqllib/bin` directory. If you want to continue using the old user exit after migration, you must copy **db2uexit** to the `INSTHOME/sqllib/adm` directory. Then, you can do one of the following:
  - If you are migrating from DB2 Version 1.x or Version 2.x, copy **db2uext2.v2** from the `DB2DIR/misc` directory to the `INSTHOME/sqllib/adm` directory and rename it to **db2uext2**. You can use the following command to copy the file:

    ```
    cp DB2DIR/misc/db2uext2.v2 INSTHOME/sqllib/adm/db2uext2
    ```

    where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
    = /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

  - If you are migrating from DB2 Parallel Edition Version 1.x, copy **db2uext2.pe** from the `DB2DIR/misc` directory to the `INSTHOME/sqllib/adm` directory and rename it to **db2uext2**. You can use the following command to copy the file:

    ```
    cp DB2DIR/misc/db2uext2.pe INSTHOME/sqllib/adm/db2uext2
    ```

    where `DB2DIR` = `/usr/lpp/db2_05_00` on AIX.

  Note: You must ensure that **db2uext2** is owned by the instance owner and is executable by the owner.

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 INSTHOME/sqllib/bin directory, used to support the pre-Version 5 user exit program, **db2uexit**, which should be removed.

Migrate the DB2 Instance

Only local cataloged databases that are owned by the DB2 instance are checked for migration. Uncataloged databases may be unusable after the instance has been migrated. Refer to the *Administration Guide* for further information.

After an instance is ready for migration, use the **db2imigr** command to migrate the instance as follows:

1 Log in as user with root authority.

If the *library_path* environment variable is set to /usr/lib on AIX or /opt/lib on HP-UX, SCO UnixWare 7, or Solaris, and there is a link in /usr/lib or /opt/lib to the Version 5 libdb2.a DB2 library, this can cause an error when using the **db2imigr** command. To fix the error, you should reset the *library_path* environment variable so that it does not reference the libraries in those paths by entering the following command:

```
unset library_path
```

where *library_path* is:

- *LIBPATH* on AIX
- *SHLIB_PATH* on HP-UX
- *LD_LIBRARY_PATH* on Solaris or SCO UnixWare 7

After migrating the DB2 instance, you should reset *LIBPATH* to its original setting.

2 Run the **db2imigr** command as follows:

```
DB2DIR/instance/db2imigr [-d] [-a AuthType] [-u fencedID] InstName
```

where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
= /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

and where:

-d Sets the debug mode that you can use for problem determination

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are (SERVER), (CLIENT), and (DCS). If the *-a* parameter is not specified, the

authentication type defaults to (SERVER), if a DB2 server is installed. Otherwise, the *AuthType* is set to (CLIENT).

Notes:

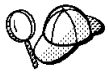
a. The authentication type of the instance applies to all databases owned by the instance.

b. While authentication type (DCE) is an optional parameter, it is not valid to choose (DCE) for this command.

-u *fencedID* Is the user under which the fenced user-defined functions (UDFs) and stored procedures will execute.

InstName Is the login name of the instance owner.

3 If there are any errors in verifying that all databases can be migrated, see Table 11 on page 61 and take the suggested corrective actions. Then, reenter the **db2imigr** command.



If you are migrating a DB2 Version 2.1 instance, created on AIX, and the instance uses the environment variable *DB2SORT* set to a keyword *SMARTSORT*, you must set the registry value *db2sort* after the instance is migrated to Version 5. Set the *db2sort* registry value to the run time library for the sort command as follows:

```
db2set DB2SORT="/usr/lib/libsort.a"
```

Migrating Databases

To migrate databases owned by an instance, you need to perform the following steps:

1 Log on with a user ID that has SYSADM authority.

2 Ensure that the databases you want to migrate are cataloged.

3 Migrate the database. Refer to the *Command Reference* for the database migration syntax command.

Trouble-Shooting Database Migration Errors

The SQL1704N warning message, **Database migration failed** with reason code 6, is received because of a failure to relocate the database. To correct this error, you should check the `db2diag.log` file and do the following if you receive the following messages.

Table 12. Correcting SQL1704N warning messages	
Error Message	**Cause/Action**
Mount unmounted filesystem(s) failed. Manual remount is required before retrying migration. Please check **fname**.	A problem was encountered when unmounting some of the filesystems. Attempts to clean up these unmounted filesystems also failed. Check the **fname** file and remount the listed filesystems before retrying the migration.
Unmount mounted filesystem(s) failed. No database relocation is done.	Attempts to unmount some filesystem(s) failed. Remount the unmounted filesystems and try the database migration again.
Remount unmounted filesystem(s) to new mount points failed. Restart of database migration is needed.	Attempts to remount filesystems to new mount points failed. Retry database migration.

Post Migration

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 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 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 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 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.

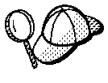
For better performance of SQL statements, we recommend that the Version 2 explain tables be dropped and new explain tables be created; refer to the *SQL Reference* and the *Administration Guide* for details on creating new explain tables. If, however, there are Version 2 explain tables that you need 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 `sqllib/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 ...
```

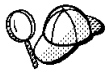
The **db2upd52** command also adds the REAL data type to the SYSIBM.SYSDATATYPES catalog table and the function signature (SYSFUN.UCASE) to the SYSIBM.SYSFUNCTIONS table. For information on the syntax of the **db2upd52** command, refer to the *Command Reference*.

Migrating from HP-UX Version 10 to Version 11

Before you migrate databases or instances from HP-UX Version 10 to Version 11, you need to perform the following steps:

1. Stop all DB2 instances using the **db2stop** command (see "Starting and Stopping an Instance" on page 150 for more information).
2. Stop the Administration Server using the **db2admin stop** command (see "Stopping the Administration Server" on page 158 for more information).
3. Back up all files under the instance's sqllib directory and any tablespaces. Refer to the *Administration Guide* for more detailed information.
4. Back up all files under the Administration Server's sqllib directory, if you created an Administration Server.
5. Back up the DB2 registry repository located under /var/opt/db2/v5.
6. Back up or record all DB2-related entries from the /etc/passwd, /etc/group, and /etc/services files.
7. Remove the entire DB2 product using the instructions in Chapter 29, "Removing DB2 Products" on page 289.

You can now migrate your operating system to HP-UX Version 11.



You may also want to migrate all your other non-DB2 applications at the same time. In this case, you need to perform all your pre-migration procedures at this time before the operating system is migrated. Please refer to the migration instructions that provided with those applications for more information. DB2 pre-migration procedures should be independent of the order of your overall pre-migration plan.

After your operating system has been migrated and all your standard Unix services are up and running, perform the following steps:

1. Adjust the kernel parameters if you have not done so. You can find a list of recommended kernel parameters in "Step 2. Updating Kernel Configuration Parameters" on page 42.

2. Install DB2 (see Chapter 3, “Installing and Configuring DB2” on page 39).
3. Recreate the DB2-specific user, group and TCP services entries in the `/etc/passwd`, `/etc/group`, and `/etc/services` files.
4. Restore from backup the instance's `sqllib` directory. Refer to the *Administration Guide* for further information.
5. Restore from backup the Administration Server's `sqllib` directory. Refer to the *Administration Guide* for further information.
6. Restore the DB2 registry repository to `/var/opt/db2/v5`.
7. Update the DB2 instances that you have, one at a time (see “Updating Instances” on page 154).
8. Update the Administration Server (see Chapter 17, “Using the Administration Server” on page 157).
9. Perform the post migration procedures described in “Post Migration” on page 67.
10. Start all DB2 instances and the Administration Server as required (see “Starting and Stopping an Instance” on page 150 and “Starting the Administration Server” on page 158).

Migrating from SCO OpenServer to SCO UnixWare 7

You cannot migrate databases or instances created using DB2 for SCO OpenServer to DB2 for SCO UnixWare 7. Instead you have to perform the following steps before installing DB2 for SCO UnixWare 7:

1. Install the **db2ckmig** command on DB2 for SCO OpenServer.

   The **db2ckmig** command is available in the `/cdrom` directory on the DB2 Universal Database for SCO UnixWare 7 CD-ROM. “Verify that Databases Can Be Migrated” on page 60 describes how to use this command. Before proceeding to the next step, ensure that you correct any errors generated by the **db2ckmig** command.
2. Backup any databases that you want to migrate - see “Pre-Migration” on page 58.
3. Record instance-level information and drop any instances. Collect the following information using the command line processor:
   - Database manager configuration. Enter the following command:

     ```
     db2 get database manager configuration
     ```

- Node directory. Enter the following command:

  ```
  db2 list node directory show detail
  ```

  You should note the following details:

  – TCP/IP service names and numbers; these will need to be mapped to `/etc/services`.

  – Any communications configurations which you will need to set up later.

- DCS directory. Enter the following command:

  ```
  db2 list dcs directory
  ```

  You should note any communications configurations which you will need to set up later.

- Database directory for database file systems/locations. Enter the following command:

  ```
  db2 list database directory
  ```

- Database configuration for log file paths. Enter the following command:

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

See “Migrating Instances” on page 58 for further information.

4 Identify any stored procedures and User Defined Functions (UDFs). All stored procedures, UDFs and user exits must be recompiled and relinked after installing SCO UnixWare 7 and DB2 Version 5.2.

5 Drop all databases and DB2 instances on the SCO OpenServer machine by entering the **db2idrop** command. See “Removing Instances” on page 156 for further information.

6 Install SCO UnixWare 7 and configure the operating system as described in the product documentation.

7 Install and configure DB2 for SCO UnixWare 7 using the recorded instance-level information - see Chapter 3, “Installing and Configuring DB2” on page 39.

8 Restore database backups. Refer to the *Administration Guide* for further information.

Due to incompatibilities between file system layout and raw device names used in SCO OpenServer and those used in SCO UnixWare 7, Version 2 databases that use DMS tablespaces or SMS tablespaces with containers on absolute paths may need to be restored using the **redirected restore** command.

9 Verify the database manager and database configuration parameters are tuned for DB2 for SCO UnixWare 7. Refer to the *Administration Guide* for further information.

To install clients, see Chapter 7, “Installing DB2 Clients” on page 79.

Part 3. Installing DB2 Clients

Chapter 7. Installing DB2 Clients

Use the instructions in this section to install a DB2 client. You can also run unattended installations of DB2 to remote target machines running OS/2, Windows 3.x, and Windows 32-bit operating systems.

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. To configure your client to access remote servers, see Chapter 13, “Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 111, or for UNIX users, see Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.

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 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 8, “Installing DB2 Clients on Windows 32-Bit Operating Systems” on page 81.
- Chapter 9, “Installing DB2 Clients on Windows 3.x Operating Systems” on page 85.
- Chapter 10, “Installing DB2 Clients on OS/2 Operating Systems” on page 89.
- Chapter 11, “Installing DB2 Clients on UNIX Operating Systems” on page 91.
- Chapter 12, “Installing DB2 Clients on Macintosh Operating Systems” on page 107.

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.

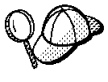
The DB2 Software Developer's Kits are on the DB2 Software Developer's Kit CD-ROM that comes with the Application Developer's Kit. The DB2 Client Application Enablers (except the DOS client) are on the DB2 Client Application Enablers CD-ROM.

Chapter 8. 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:

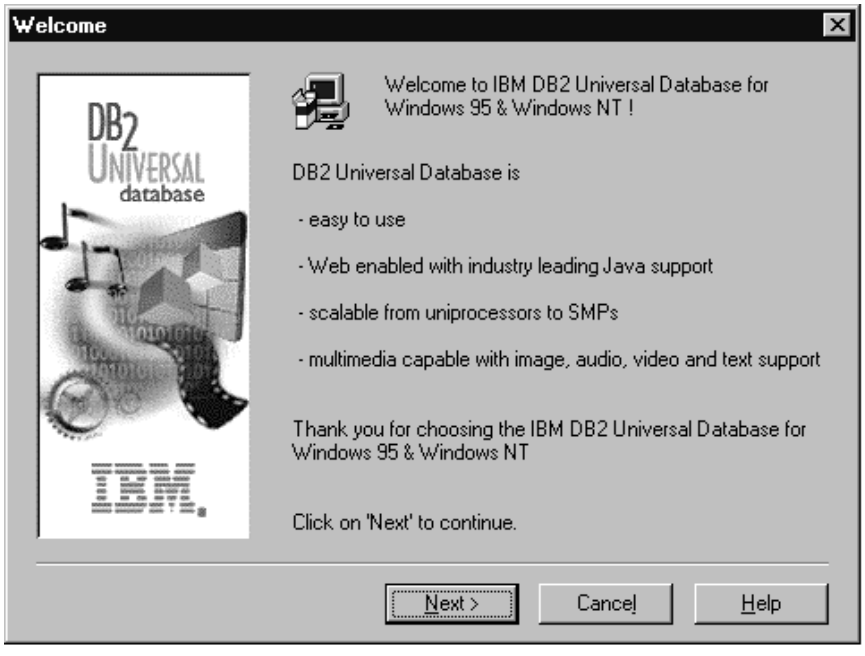
1. Click on **Start** and select the **Run** option.
2. Type the following in the **Open** field:

   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 38 on page 316 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 13, “Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 111.

Chapter 9. Installing DB2 Clients on Windows 3.x Operating Systems

Note: 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 this installation procedure.

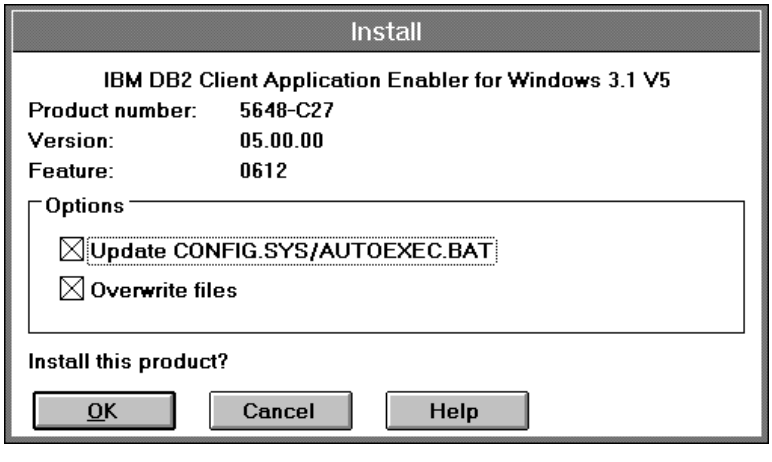
To install the DB2 Client Application Enabler on Windows 3.x:

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 38 on page 316 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.

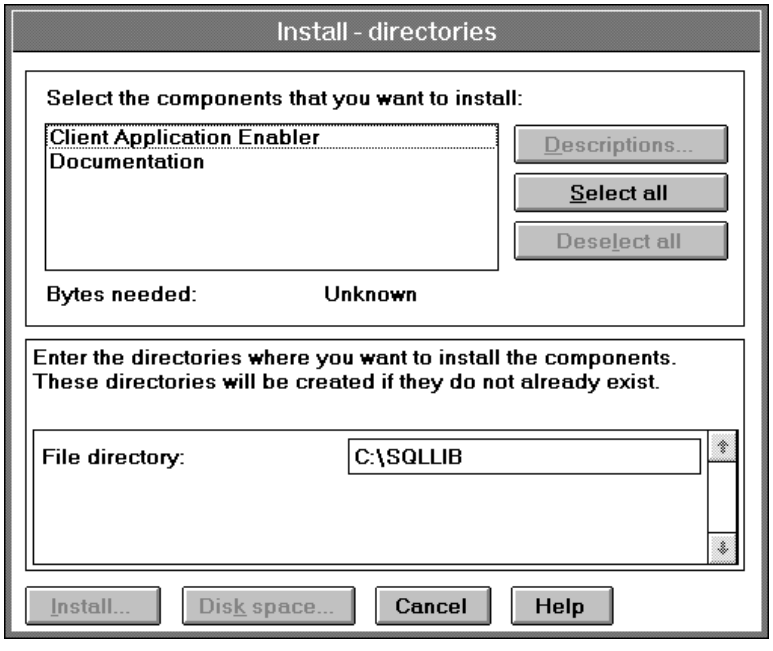
   Note: The config.sys file will not be modified by the installation.

   - 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.

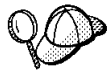
The **Overwrite files** check box is selected by default, it is recommend that you do not change this setting.

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.

Chapter 10. 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 38 on page 316 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:\sqllib`.

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 13, “Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 111.

Chapter 11. Installing DB2 Clients on UNIX Operating Systems

This section contains the information that you need to install the DB2 Client Application Enabler on UNIX operating systems. See the following sections as appropriate:

- "Installing DB2 Version 5 Clients" on page 92.
- "Installing DB2 Version 2 Clients" on page 101.

If you want to remove the DB2 Client Application Enabler, go to Chapter 29, "Removing DB2 Products" on page 289.

Before You Begin

Before you begin installing DB2 products using the DB2 Installer program, you need to gather the following information:

Where is the CD-ROM mount point?
You need to mount the CD-ROM before you can install the DB2 product. To mount the CD-ROM, you must know where the mount point is. For example, the CD-ROM mount point could be `/cdrom`.

Note: On the Solaris operating system, the CD-ROM is automatically mounted if the Volume Manager (vold) is running.

How do I assign a group name and user name for a DB2 instance?
A separate user name must be used for each DB2 Instance. We recommend that a new group be created which should be used as a primary group for the DB2 instance user.

Installing DB2 Version 5 Clients

To install the DB2 Client Application Enabler on AIX, HP-UX, SCO UnixWare 7, Silicon Graphics IRIX, and Solaris systems, perform the following steps:

Step 1. Update Kernel Configuration Parameters

To run DB2 Client Application Enabler for UNIX systems, you may have to update some kernel configuration parameters.

This step is not required on AIX. If you are installing an AIX client, skip this step and go to "Step 2. Mount the DB2 Client Application Enablers CD-ROM" on page 94.

Recommended Values for HP-UX Version 10 and Version 11

Table 13 lists the recommended values for the specified HP-UX kernel configuration parameters. These values are valid for HP-UX 10 and HP-UX 11.

Table 13. HP-UX Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Recommended Value
msgseg	8192
msgmnb	65535 (1)
msgmax	65535 (1)
msgssz	16

Notes:

1. Parameters `msgmnb` and `msgmax` must be set to 65535.
2. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in Table 13.

To change a value:

1 Enter the **sam** command to start the System Administration Manager (SAM) program.

2 Double-click on the **Kernel Configuration** icon.

3 Double-click on the **Configurable Parameters** icon.

4. Double-click on the parameter that you want to change and enter the new value in the **Formula/Value** field.
5. Click on **OK**.

   The HP-UX operating system automatically reboots after you change the values for the kernel configuration parameters.

Go to "Step 2. Mount the DB2 Client Application Enablers CD-ROM" on page 94 to continue with the installation.

Recommended Values for SCO UnixWare 7

Table 14 lists the recommended values for the specified SCO UnixWare 7 kernel configuration parameters.

Table 14. SCO UnixWare 7 Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Recommended Value
msgmax	65535 (1)
msgmnb	65535 (1)
msgssz	524288
msgmni	256

Notes:

1. Parameters *msgmax* and *msgmnb* should be set at least to 65535.
2. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in the preceding table.

To change a value, do the following:

1. Enter the **scoadmin** command to start the System Administration tool.
2. Double-click on the **System** folder.
3. Double-click on the **System Tuner** icon.
4. Click on the drop down box and select the **Inter-Process Communication (IPC) Parameters**.
5. Select the parameter to be changed and enter the new value.
6. Click on **OK** when you have finished changing all the parameters.
7. Click on the **Yes** push button to rebuild the kernel.
8. Reboot the system so that the changes can take effect.

Go to “Step 2. Mount the DB2 Client Application Enablers CD-ROM” on page 94 to continue with the installation.

Recommended Values for Solaris

Table 15 lists the recommended values for the specified Solaris kernel configuration parameters.

Table 15. Solaris Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Recommended Value
msgsys:msginfo_msgmax	65535 (1)
msgsys:msginfo_msgmnb	65535 (1)
msgsys:msginfo_msgseg	8192
msgsys:msginfo_msgssz	16

Notes:

1. Parameters `msgsys:msginfo_msgmnb` and `msgsys:msginfo_msgmax` must be set to 65535.

To set a kernel parameter, add a line at the end of the `/etc/system` file as follows:

```
set parameter_name = value
```

where *parameter_name* represents the parameter you want to change.

For example, to set the value of parameter *msgsys:msginfo_msgmax*, add the following line to the end of the `/etc/system` file:

```
set msgsys:msginfo_msgmax = 65535
```

After changing the kernel parameters, reboot the system so that the changes can take effect.

Go to “Step 2. Mount the DB2 Client Application Enablers CD-ROM” to continue with the installation.

Step 2. Mount the DB2 Client Application Enablers CD-ROM

To install DB2 Client Application Enabler using the DB2 Installer, you must first mount the CD-ROM. Once you have mounted the CD-ROM, you can start installing the DB2 products.

Mounting on AIX Systems

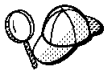
Perform the following steps to mount the CD-ROM on AIX operating systems:

1. Log in as user with root authority.
2. Insert the DB2 Client Application Enablers CD-ROM in the drive.
3. Create a directory to mount the CD-ROM by entering the following command:

   ```
   mkdir -p /cdrom
   ```

   where `cdrom` is the CD-ROM mount directory.
4. Allocate a CD-ROM file system by entering the following command:

   ```
   smitty storage
   ```
5. Select **File Systems**.
6. Select **Add / Change / Show / Delete File Systems**.
7. Select **CDROM File Systems**.
8. Select **Add CDROM File System**.
9. Select **Device Name**.



Device names for CD-ROM file systems must be unique. If there is a duplicate device name, you may need to delete a previously-defined CD-ROM file system or use another name for your directory.

10. In the pop-up window, enter `/cdrom` as the **mount point**.
11. Mount the CD-ROM file system by entering the following command:

    ```
    smit mountfs
    ```
12. Select the **FileSystem** name. For example, the name could be `/dev/cd0`.
13. Select the **Directory** name, `/cdrom`.
14. Select the **Type of filesystem**, `cdrfs`.
15. Set the **Mount as READ-ONLY system** to `Yes`.
16. Log off.

Go to “Step 3. Perform the Installation” on page 98 to continue with the installation.

Mounting on HP-UX Systems

Perform the following steps to mount the CD-ROM on HP-UX operating systems:

1. Log in as user with root authority.
2. Insert the CD-ROM in the drive and mount it as in the following example:

   ```
   mkdir /cdrom
   /usr/sbin/mount /dev/dsk/c0t2d0 /cdrom
   ```

   where `/cdrom` is the CD-ROM mount directory.
3. Log out.

The CD-ROM may also be mounted using the System Administration (**SAM**) tool. Consult your HP-UX documentation for more information about **SAM**.

Go to “Step 3. Perform the Installation” on page 98 to continue with the installation.

Mounting on SCO UnixWare 7

Perform the following steps to mount the CD-ROM on SCO UnixWare 7 operating systems:

1. Log in as user with root authority.
2. Insert the DB2 Client Application Enablers CD-ROM into the drive and mount it. For example, to mount the CD-ROM as `/cdrom`, type:

   ```
   mount -F cdfs -o ro /dev/cdrom/c0b0t5l0 /cdrom
   ```

   where `c0b0t5l0` is the device name found under `/dev/cdrom` directory and varies with the particular hardware on the system.
3. Log out.

Go to “Step 3. Perform the Installation” on page 98 to continue with the installation.

Mounting on Silicon Graphics IRIX

Perform the following steps to mount the CD-ROM on Silicon Graphics IRIX operating systems:

1. Log in as user with root authority.
2. Insert the DB2 Client Application Enablers CD-ROM into the drive and mount it using the **mount** command as follows:

   ```
   mount -t iso9660 device mount_point
   ```

   For example, to mount the CD-ROM as `/cdrom`, enter the following commands:

   ```
   mkdir /cdrom
   mount -t iso9660  /dev/scsi/sc0d7l0 /cdrom
   ```

3. Log out.

Go to "Step 3. Perform the Installation" on page 98 to continue with the installation.

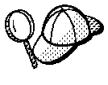
Mounting on Solaris Systems

Perform the following steps to mount the CD-ROM on Solaris operating systems:

1. Log in as user with root authority.
2. If the Volume Manager is not running on your system, enter the following commands to mount the CD-ROM:

   ```
   mkdir -p /cdrom/unnamed_cdrom
   mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom/unnamed_cdrom
   ```

   where */cdrom/unnamed_cdrom* is the CD-ROM mount directory.



If you are mounting the CD-ROM drive from a remote system using NFS, the CD-ROM file system on the remote machine must be exported with *root* access. You must also mount that file system with *root* access on the local machine.

   If the Volume Manager (vold) is running on your system, the CD-ROM is automatically mounted as:

   ```
   /cdrom/unnamed_cdrom
   ```

3. Log out.

Go to “Step 3. Perform the Installation” on page 98 to continue with the installation.

Step 3. Perform the Installation

After you mount the CD-ROM file system, use the DB2 Installer program to install the DB2 products.

If you are installing the DB2 Client Application Enabler from a remote server, it is better to use the **telnet** command to open a telnet session instead of using the **rlogin** command to connect to your remote server.

1. Log in as user with root authority.
2. Insert the DB2 Client Application Enablers CD-ROM in the drive and if required, mount the CD-ROM. Refer to “Step 2. Mount the DB2 Client Application Enablers CD-ROM” on page 94 if you need to mount the CD-ROM.
3. Change to the directory where the DB2 install CD-ROM is mounted and run the following command:

   ```
   ./db2setup
   ```

4. DB2 install images are available in the following directories (assuming that the mount point for the CD-ROM is `/cdrom`):
   - On AIX:

     ```
     cd /cdrom/db2/aix
     ```

   - On HP-UX:

     ```
     cd /cdrom/db2/hpux10 (for HP-UX Version 10)
     cd /cdrom/db2/hpux11 (for HP-UX Version 11)
     ```

   - On SCO UnixWare 7:

     ```
     cd /cdrom/db2/scouw
     ```

   - On SGI:

     ```
     cd /cdrom/db2/sgi
     ```

   - On Solaris:

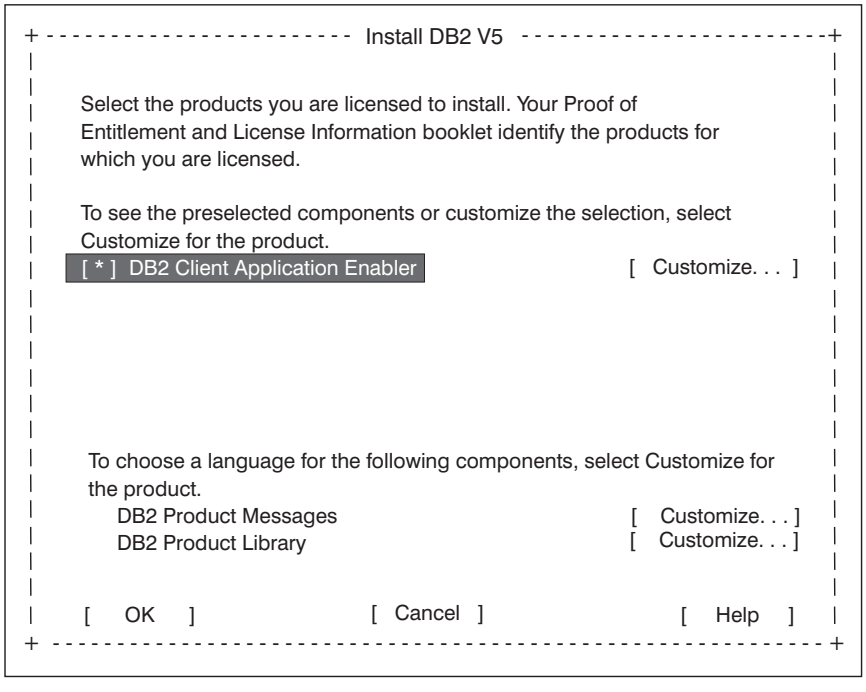
     ```
     cd /cdrom/unnamed_cdrom/db2/solaris
     ```

5. To install the client, enter the following commands:

   ```
   cd /cdrom
   ./db2setup
   ```

The installation program will proceed to install the appropriate client on your workstation.

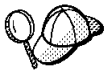
It will take some time for the DB2 Installer program to start up, as it is scanning your system for information.



Note: Your screen will be different if you have already installed DB2 Version 5 products on your system. See Chapter 4, “Installing Additional Products and Creating Additional Instances” on page 53 for more information.

6 From the product list on the Install DB2 V5 window, select the DB2 products that you want to install.

To display the required and optional components for a product that you want to install, select **Customize**. The optional components that are most typically used are pre-selected for you.

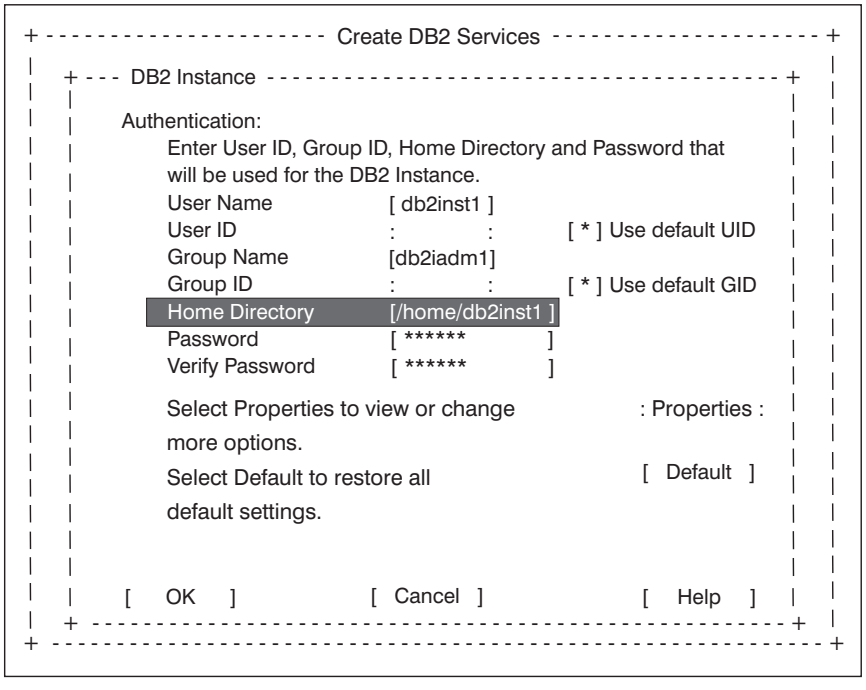


To refresh the current screen, press the F5 key or Ctrl+L.

When you have finished your selection, select **OK**. To undo any selections you made, select **Default**. The Create DB2 Services window opens.

Go to step 9 on page 100 if you do not want to create a DB2 Instance at this time.

7 At the Create DB2 Services window, select the **Create a DB2 Instance** option. The DB2 Instance window opens:



You can use the default values displayed at the DB2 Instance window or change them. To change the default **User ID** for the instance, deselect **Use default UID** and enter a new value for the **User ID**.

8 Complete the fields and select **OK**.

9 Select **OK** and the Summary Report window opens. The report lists the installable items as well as the location of the installation log file.

10 Select the Summary Report field and use the Up or Down arrow keys to review the Summary Report. If the selections are not correct, press the **ESC** or **F3** key, to return to the previous screen. Otherwise select **Continue** to start the installation.

11 When the installation is complete, use the Up or Down arrow keys to review the Status Report. Select **View Log** to view the installation log file. Select **OK** to close the Status Report screen.

12 Select **Close** from the DB2 Installer screen to terminate the DB2 Installer program.

After the installation is complete, the software is installed in the DB2DIR directory,

where DB2DIR	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, SGI IRIX, or Solaris

Proceed to Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.

Installing DB2 Version 2 Clients

This section details installing DB2 products on SINIX and SCO OpenServer Clients.

Installing DB2 Version 2 Clients on SINIX

To install the DB2 Client Application Enabler on SINIX, perform the following steps:

1. Log in as user with root authority.
2. Insert the DB2 Client Application Enablers CD-ROM into the drive and mount it using either the **sysadm** or the **mount** command as follows:

   ```
   mkdir /cdrom
   mount -F hs /dev/ios0/sdisk006s0 /cdrom
   ```

   where /cdrom is the CD-ROM mount directory.

   The install images for DB2 for SINIX products are available in the /cdrom/db2/sinix/IBMdb2 directory.
3. Install the DB2 Client Application Enabler. Type the following command:

   ```
   pkgadd -d /cdrom/db2/sinix/IBMdb2 db2cliv21 db2msgDe db2conv
   ```

Some kernel configuration parameters must be changed to use DB2 Client Application Enabler for SINIX. Use the SINIX **idtune** command to update the values as given in Table 16 on page 102. After doing so, rebuild the SINIX kernel and reboot the system for the changes to take effect.

Table 16 on page 102 lists the recommended values for the specified SINIX kernel configuration parameters.

Table 16. SINIX Kernel Configuration Parameters (Recommended Values)	
SINIX RM600 Kernel Parameter	**Recommended Value**
msgmax	65535
msgmnb	65535
msgseg	8192
msgssz	16

Note: The kernel configuration parameters msgmnb and msgmax must be changed to 65535.

Installing DB2 Version 2 Clients on SCO OpenServer

To install the DB2 Client Application Enabler on SCO OpenServer, perform the following steps:

1. Log in as user with root authority.
2. Insert the DB2 Client Application Enablers CD-ROM into the drive and mount the CD-ROM by entering:

   ```
   mkdir /cdrom
   mount /dev/cd0 /cdrom
   ```

   where */cdrom* is the CD-ROM mount directory.

   The install images for the DB2 products are available in the /cdrom/db2/sco/IBMdb2 directory.
3. Enter the **custom** command. The Software Manager window opens.
4. From the Software Manager window, do the following:
   a. From the **Software** pulldown menu, select **Install New**.
   b. If you are doing a remote install, select the host from which you want to install. Otherwise, select **Continue.**
   c. From the **Media Images** menu, select **Media Images** and select **Continue**.
   d. In the **Image Directory** field, enter:

      ```
      /cdrom/db2/sco/IBMdb2
      ```

   e. Choose the software you want to install:
      - To install all components, select **Full**.
      - To install selected components, choose **Partial**. On the Software Selection window, highlight the products you want to install:
        – DB2 Client Application Enabler for SCO OpenServer

– DB2 Software Developer's Kit for SCO OpenServer

– DB2 Product Messages for SCO OpenServer.

Press Enter.

f Update the kernel configuration parameters.

Change the values as given in Table 17 by updating the `/etc/conf/cf.d/stune` file. Then, rebuild the kernel and reboot the system for the changes to take effect.

Table 17. SCO OpenServer Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Recommended Value
msgssz	64
semmap	100
semmi	100
semmns	100
semmnu	100

Create or Assign Groups and Users

If you do not already have a user ID to own the instance and a system administration group ID that is the primary group of the instance owner, create these as follows:

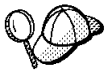
1 Log in as user with root authority.

2 Create a system administration group ID using either the **sysadm** or **groupadd** command. For example, to create a group named `dbadmin1`, type:

```
groupadd dbadmin1
```

3 Create a user ID that will be the instance owner using either the **sysadm** or **useradd** command. For example, to create a user named `inst1` type:

```
useradd -g dbadmin1 -G dbadmin1 -d /home/inst1 inst1
passwd inst1
```



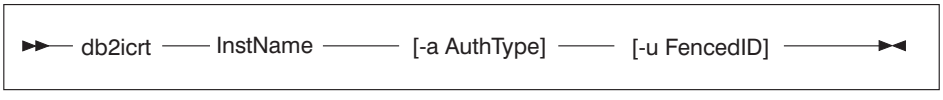
- When choosing a name for a group or user, follow the rules described in Appendix D, “Naming Rules” on page 317.
- Dedicate the instance-owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

Create an Instance of the Product

Use the **db2icrt** command to create an instance of the product. The **db2icrt** command is located in the `/opt/IBMdb2/V5.0/instance` directory.

On SCO OpenServer, go to the `/usr/IBMdb2/V2.1/instance` directory. On SINIX, go to the `/opt/IBMdb2/V2.1/instance` directory.

The syntax of the **db2icrt** command is:



where:

instance_name Is the login name of the instance owner.

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are SERVER, CLIENT, and DCS. If the -a parameter is not specified, the authentication type defaults to SERVER, if a server product is installed. Otherwise, the *AuthType* is set to CLIENT.

Note: The authentication type of the instance applies to all databases under the instance.

-u *FencedID* Is the user under which the fenced UDFs and stored procedures will execute. This is not required if you install the DB2 Client Application Enabler or the DB2 Software Developer's Kit. For other products, this is an required parameter.

Note: *FencedID* may not be root or bin.

To create an instance for the DB2 client, you can use the following command:

```
db2icrt db2inst1
```

When an instance is created, its name is also added to the list of instances on the system.

The **db2icrt** command creates the *INSTHOME*/`sqllib` directory, where *INSTHOME* is the home directory of the instance owner.

Set Up the DB2 Operating Environment

Before starting DB2, you must execute a script to set up the DB2 operating environment and select an instance. The sample script files, `db2profile` (for Bourne or Korn shell) and `db2cshrc` (for C shell) are provided to help you set up the operating environment for a DB2 instance.

The instance owner may customize these scripts for all users of an instance. These scripts are available in the INSTHOME/sqllib directory, where INSTHOME is the home directory of the instance.

Determine the most appropriate way to execute the commands in the sample db2profile or db2cshrc script file in your environment and instruct your users to do it. For example you could add the following to the start-up script:

```
. INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/db2cshrc (for C shell)
```

Create Links for DB2 Files

If you are developing or running applications, you may want to create links for DB2 libraries to avoid specifying the full path to the product libraries and the include files. To create these links, use the **db2ln** command.

On SCO OpenServer systems, type:

```
/usr/IBMdb2/V2.1/cfg/db2ln
```

On SINIX systems, type:

```
/opt/IBMdb2/V2.1/cfg/db2ln
```

This creates links from the DB2 libraries to /usr/lib, and from the DB2 include files to /usr/include.

To configure your client to access remote servers, go to Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.

Chapter 12. Installing DB2 Clients on Macintosh Operating Systems

To install the DB2 Client Application Enabler on a Macintosh operating system, perform the following steps:

1. Insert the DB2 Client Application Enablers CD-ROM into the drive.
2. Double-click on the **DB2_CAE_V502** icon.
3. Double-click on the **MAC** folder.
4. Double-click on the *language* folder, where *language* is the two-character country code that represents your language (for example, EN for English). Table 38 on page 316 lists the codes for each available language.
5. Double-click on the **CLIENT** folder.
6. Double-click on the **CAEMAC** icon to start the installation. The Welcome window opens.
7. Follow the instructions provided to complete the installation.
8. When the install program is finished, click on the **Restart** push button to restart your computer and complete the installation.

To configure your client to access a remote DB2 server, see Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.

Part 4. Configuring DB2 Communications

Chapter 13. 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.

To configure all other clients, see Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.

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.

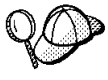
     a. Click on the **Browse** push button and select the appropriate access profile.

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.

  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:

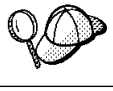
- The Administration Server *DISCOVER* configuration parameter on the remote system is set to DISCOVER=DISABLE.
- The *DB2COMM* registry value and the *DISCOVER_COMM* configuration parameter are not configured on the Administration Server of the remote system with a discovery protocol being used on the 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.

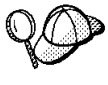
6. The Confirmation window opens. Click on the **Test Connection** push button to test the connection to this database.

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 140 for more information.

1. Start the database manager by entering the **db2start** command on the server.

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 141.

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:

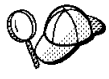
```
"select tabname from syscat.tables"
```

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 15, “Getting Started with DB2 Universal Database” on page 139 for details.

Chapter 14. Configuring Client-to-Server Communications Using the Command Line Processor

This section describes how to use commands to configure a client to communicate with a DB2 server using the command line processor.



If you are planning to use an OS/2 or Windows 32-bit client for communications, the Client Configuration Assistant (CCA) makes it easy to automate the tasks of configuring and administering DB2 clients to communicate with DB2 servers. If you have installed the CCA, it is recommended that you use this utility to configure your DB2 clients for communications. See Chapter 13, "Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 111 for more information.

For instructions on using the command line processor, see Chapter 19, "Entering DB2 Commands and SQL Statements" on page 181. For a description of database naming rules, see Appendix D, "Naming Rules" on page 317.

Go to the communication protocol that you want your client to use to access a remote server.

- TCP/IP - see "Configuring TCP/IP on the Client"
- IPX/SPX - see "Configuring IPX/SPX on the Client" on page 126
- APPC - refer to the *Installing and Configuring DB2 Clients* online manual for more information.

Configuring TCP/IP on the Client

This section assumes that TCP/IP is functional on the client and DB2 server workstations. See "Software Requirements" on page 27 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 34 for the supported communication protocols for your particular client and server.

The following steps are required to set up a DB2 client to use TCP/IP communications:

1. Identify and record parameter values.
2. Configure the client:

a Resolve the server's host address.

b Update the `services` file.

c Catalog a TCP/IP node.

d Catalog the database.

3 Test the connection between the client and server.

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.

Table 18 (Page 1 of 2). TCP/IP Values Required at the Client

Parameter	Description	Sample Value	Your Value

Hostname (*hostname)*
or
IP address (*ip_address*)
	Use the *hostname* or *ip_address* of the remote server workstation.
	To resolve this parameter:
	• Issue the **hostname** command at the server to obtain the *hostname.*
	• Contact the server administrator to obtain the *ip_address.* (On AIX, you can enter the **ping** *server_hostname* command to obtain the *ip_address*).
		`serverhost`
		or
		`9.21.15.235`	

Table 18 (Page 2 of 2). TCP/IP Values Required at the Client

Parameter	Description	Sample Value	Your Value
Connection Port • Connection Service name (*svcename*) • Port number/Protocol (port_number/tcp)	Values required in the `services` file. The Connection Service name is an arbitrary name used to represent the port number (*port_number*) on the client. The port number for the client must be the same as the port number that the *svcename* parameter maps to in the `services` file at the server. (The *svcename* parameter is located in the database manager configuration file on the DB2 server.) This value must not be in use by any another applications, and must be unique within the `services` file.	`server1` `3700/tcp`	
Node name (*node_name*)	A local alias, or nickname, that describes the node where the database resides. You can choose any name you want, however, all node name values within your local node directory must be unique.	`db2node`	

Step 2. Configure the Client

The following steps describe how to configure this protocol on the client. Replace the sample values with your worksheet values.

A. Resolve the Server's Host Address

If your network has a name server, or you are planning to directly specify the IP address (*ip_address*) of the server, skip this step and proceed to “Step B. Update the Services File” on page 121.

The client must know the address of the server to which it is attempting to establish communications. If a name server does not exist on your network, you may directly specify a hostname that maps to the IP address (*ip_address*) of the host in the local `hosts` file. See Table 19 on page 120 for the location of the `hosts` file for your particular platform.

If you are planning on supporting a UNIX client that is using Network Information Services (NIS), and you are not using a name server on your network, you must update the `hosts` file located on your NIS master server.

Table 19. Location of the Local Hosts and Services Files

Platform	Location
Macintosh	The `hosts` file is located in the folder called **System Folder**. **Note:** This operating system does not use a `services` file. You must catalog this node using the *port_number* parameter. See "Step C. Catalog a TCP/IP Node" on page 121 for more information.
OS/2	Specified by the *etc* environment variable. Enter the **set etc** command to determine the location of your local `hosts` or `services` files. **Note:** For DOS and WIN-OS2 sessions, you might need to update the `hosts` and `services` files located in the `tcpip_product\dos\etc` directory.
Windows 3.x	Typically in the `tcpip_product\etc` directory, but it depends on the products that you have installed. Refer to your TCP/IP documentation for more information.
Windows 95 and Windows 98	`windows` directory
Windows NT	`winnt\system32\drivers\etc` directory
UNIX	`/etc` directory

Using a text editor, add an entry to the client's `hosts` file for the server's hostname. For example:

```
9.21.15.235     serverhost   # host address for serverhost
```

where:

9.21.15.235 is the *ip_address*

serverhost is the *hostname*

is a comment describing the entry

Notes:

1. If the server is not in the same domain as the client, you must provide a fully qualified domain name such as `serverhost.vnet.ibm.com`, where `vnet.ibm.com` is the domain name.
2. For specific information on resolving host addresses, refer to your TCP/IP documentation.

Step B. Update the Services File

If you are planning to catalog a TCP/IP node using the port number (*port_number*), skip this step and go to "Step C. Catalog a TCP/IP Node."

If you are configuring a DB2 for Macintosh client, you must catalog the TCP/IP node using the port number (*port_number*). Go to "Step C. Catalog a TCP/IP Node" to catalog the node.

Using a local text editor, add the Connection Service name and port number to the client's `services` file for TCP/IP support. For example:

```
server1  3700/tcp  # DB2 connection service port
```

where:

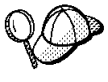
server1 is the Connection Service name

3700 is the port number for the Connection Port

`tcp` is the communication protocol that you are using

`#` is a comment describing the entry

The port number used on the client must match the port number used on the server.



If you are planning on supporting a UNIX client that is using Network Information Services (NIS), you must update the `services` file located on your NIS master server.

The file called `services` is located in the same directory as the local `hosts` file that you may have edited in "A. Resolve the Server's Host Address" on page 119.

See Table 19 on page 120 for the location of the `services` file for your particular platform.

Step C. Catalog a TCP/IP Node

You must add an entry to the client's node directory to describe the remote node.

This entry specifies the chosen alias (*node_name*), the *hostname* (or *ip_address*), and the *svcename* (or *port_number*) that the client will use to access the remote server.

To catalog a TCP/IP node, perform the following steps:

1. Log on to the system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority.

If you have trouble logging on to the system, see “Logging on to the System” on page 139.

If you are configuring communications for a Windows 3.x client, you can use the Client Setup tool to complete this task. Go to “Using the Client Setup Tool to Configure a Windows 3.x Client” on page 134 to catalog a node using the Client Setup tool.

2 If you are using a UNIX client, set up the instance environment and invoke the DB2 command line processor. Run the start-up script as follows:

```
. INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/db2cshrc (for C shell)
```

where *INSTHOME* is the home directory of the instance.

3 Catalog the node by entering the following commands in the command line processor:

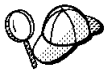
```
catalog tcpip node node_name remote [ hostname | ip_address ] server [ svcename | port_number ]
terminate
```

For example, to catalog the remote server *serverhost* on the node called *db2node*, using the service name *server1*, use:

```
catalog tcpip node db2node remote serverhost server server1
terminate
```

To catalog a remote server with the IP address *9.21.15.235* on the node called *db2node*, using the Port number *3700*, use:

```
catalog tcpip node db2node remote 9.21.15.235 server 3700
terminate
```



If you need to change values that were set with the **catalog node** command, first run the **uncatalog node** command in the command line processor as follows:

```
uncatalog node node_name
```

Recatalog the node with the value that you want to use.

Step D. Catalog the Database

Before a client application can access a remote database, the database must be cataloged on the server node and on any client nodes that will connect to it. When you create a database, it is automatically cataloged on the server with the database alias (*database_alias*) the same as the database name (*database_name*). The information in the database directory, along with the information in the node directory, is used on the client to establish a connection to the remote database.

To catalog a database on the client, perform the following steps.

1. Log on to the system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority.

If you have trouble logging on to the system, see "Logging on to the System" on page 139.

2. Fill in the *Your Value* column in the following worksheet.

Table 20. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (*database_name*)	The database alias (*database_alias*) of the *remote* database. When you create a database, it is automatically cataloged on the server with the database alias (*database_alias*) the same as the database name (*database_name*).	`sample`	
Database alias (*database_alias*)	An arbitrary local nickname for the remote database, on the client. If you do not provide one, the default is the same as the database name (*database_name*). This is the name that you use when connecting to a database from a client.	`tor1`	
Node name (node_name)	The name of the node directory entry that describes where the database resides. Use the same value for node name (*node_name*) that you used to catalog the node in the previous step.	`db2node`	

If you are configuring communications for a Windows 3.x client, you can use the Client Setup tool to complete this task. Go to "Using the Client Setup Tool to Configure a Windows 3.x Client" on page 134 to catalog a database using the Client Setup tool.

3. If you are using a UNIX client, set up the instance environment and invoke the DB2 command line processor. Run the start-up script as follows:

```
. INSTHOME/sqllib/db2profile     (for Bourne or Korn shell)
source INSTHOME/sqllib/db2cshrc (for C shell)
```

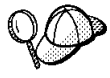
where *INSTHOME* is the home directory of the instance.

4. Catalog the database by entering the following commands in the command line processor:

```
catalog database database_name as database_alias at node node_name
terminate
```

For example, to catalog a remote database called *sample* so that it has the alias *tor1*, on the node *db2node*, use:

```
catalog database sample as tor1 at node db2node
terminate
```



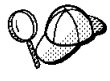
If you need to change values that were set with the **catalog** *database* command, first run the **uncatalog** *database* command in the command line processor as follows:

```
uncatalog database database_alias
```

Recatalog the database with the value that you want to use.

Step 3. Test the Client-to-Server 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 140 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. Enter 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 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 tabname 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 server. See Chapter 15, “Getting Started with DB2 Universal Database” on page 139 for details.

Troubleshooting the Client-to-Server Connection

If the connection fails, check the following items:

At the *server*:

1. The *db2comm* registry value includes the value `tcpip`.

Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, see Chapter 20, “Controlling Your DB2 Environment” on page 187.

2. The `services` file was updated correctly.
3. The service name (*svcename*) parameter was updated correctly in the database manager configuration file.
4. The database was created and cataloged properly.
5. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).

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.

Refer to the *Troubleshooting Guide* for information on the `db2diag.log` file.

At the *client*:

1. If used, the `services` and `hosts` files were updated correctly.
2. The node was cataloged with the correct hostname (*hostname*) or IP address (*ip_address)*.
3. The port number must match, or the service name must map to, the port number used on the server.
4. The node name (*node_name*), specified in the database directory, points to the correct entry in the node directory.

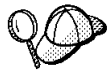
5. The database was cataloged properly, using the *server's* database alias (*database_alias*) that was cataloged when the database was created on the server, as the database name (*database_name*) on the client.

After you verify these items, refer to the *Troubleshooting Guide* if the connection still fails.

Configuring IPX/SPX on the Client

This section assumes that IPX/SPX is functional on the client and DB2 server workstations. See "Software Requirements" on page 27 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 34 for the supported communication protocols for your particular client and server.

A client can access a DB2 server via Direct Addressing or File Server Addressing. See Table 21 for a list of the available IPX/SPX clients and their supported addressing methods. For a description of Direct Addressing or File Server Addressing, see "Configuring IPX/SPX on the Server" on page 256.



You need to know the IPX/SPX addressing method that was used to configure the server before you configure your DB2 client. If the DB2 server was configured to use Direct Addressing, you must configure your client to use Direct Addressing to communicate with the server. If the DB2 server was configured for File Server Addressing, you can choose to configure your client to use either Direct Addressing or File Server Addressing, provided that your client supports the method that you want to use. See Table 21 for a list of client specific IPX/SPX supported addressing methods.

Table 21. IPX/SPX Supported Communication Methods for a DB2 Client

Client Platform	Direct Addressing	File Server Addressing
Macintosh	no support	
OS/2	√	√
SCO UnixWare 7	√	
UNIX (except SCO UnixWare 7)	no support	
SCO OpenServer	√	
Windows 3.x	√	√
Windows 95 and Windows 98	√	
Windows NT	√	

The following steps are required to set up a DB2 client to use IPX/SPX communications:

1. Identify and record parameter values.
2. Configure the client:
   a. Catalog the IPX/SPX node.
   b. Catalog the database.
3. Test the connection between the client and server.

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.

Table 22 (Page 1 of 2). IPX/SPX Values Required at the Client

Parameter	Description	Sample Value	Your Value
File server name (*FILESERVER*)	**Direct Addressing** A * value indicates that you are using Direct Addressing. **File Server Addressing** The name of the NetWare file server where the database server instance is registered. This parameter must be entered in UPPERCASE. Locate this parameter in the database manager configuration file on the server.	**Direct Addressing** * **File Server Addressing** NETWSRV	

Table 22 (Page 2 of 2). IPX/SPX Values Required at the Client

Parameter	Description	Sample Value	Your Value

DB2 server object name (*OBJECTNAME*)	**Direct Addressing**
	The server's IPX/SPX internetwork address of the form:
	`netid (8 byte).nodeid (12 byte). socket # (4 byte)`
	To resolve this parameter, enter the **db2ipxad** command at the server. See "A. Catalog the IPX/SPX Node" on page 128 for more information.
	File Server Addressing
	The database manager server instance, represented as the object *OBJECTNAME* on the NetWare file server. The server's IPX/SPX internetwork address is stored and retrieved from this object.
	This parameter must be entered in UPPERCASE and be unique on the NetWare file server system.
	Locate this parameter in the database manager configuration file on the server.
		Direct Addressing
		`09212700.400011527745.879E`
		File Server Addressing
		`DB2INST1`	

Node name (*node_name*)	A local alias, or nickname, that describes the node where the database resides. You can choose any name you want, however, all node name values within your local node directory must be unique.	`db2node`	

Step 2. Configure the Client

The following steps describe how to configure this protocol on the client. Replace the sample values with your worksheet values.

A. Catalog the IPX/SPX Node

You must add an entry to the client's node directory to describe the remote node.

This entry specifies the chosen alias (*node_name*), the file server name (*FILESERVER*), and the DB2 server object name (*OBJECTNAME*) that the client will use to access the remote DB2 server.

To catalog the IPX/SPX node, perform the following steps:

1. Log on to the system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority.

If you have trouble logging on to the system, see "Logging on to the System" on page 139.

If you are configuring communications for a Windows 3.x client, you can use the Client Setup tool to complete this task. Go to "Using the Client Setup Tool to Configure a Windows 3.x Client" on page 134 to catalog a node using the Client Setup tool.

2. If you are using a UNIX client, set up the instance environment and invoke the DB2 command line processor. Run the start-up script as follows:

```
. INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/db2cshrc (for C shell)
```

   where *INSTHOME* is the home directory of the instance.

3. Catalog the node by issuing the following commands in the command line processor:

```
catalog ipxspx node node_name remote FILESERVER server OBJECTNAME
terminate
```

   Direct Addressing Example

   You must assign a * to the *FILESERVER* parameter and specify the server's IPX/SPX internetwork address value as the *OBJECTNAME* parameter on the client.

   To determine the value for the *OBJECTNAME* parameter, enter the **db2ipxad** command on the server. (This command is located in the `sqllib/misc/` directory.)

   Make note of the output that is generated, and use that value in place of the sample value (`09212700.400011527745.879E`) in the example that follows.

   To catalog a remote node called *db2node*, using the IPX/SPX internetwork address *09212700.400011527745.879E* as the *OBJECTNAME*, use:

```
catalog ipxspx node db2node remote * server 09212700.400011527745.879E
terminate
```

   File Server Addressing Example

   To catalog a remote node called *db2node* at the file server *NETWSRV*, in the instance *DB2INST1*, use:

```
catalog ipxspx node db2node remote NETWSRV server DB2INST1
terminate
```

If you need to change values that were set with the **catalog node** command, first run the **uncatalog node** command in the command line processor as follows:

```
uncatalog node node_name
```

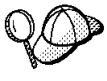
Recatalog the node with the value that you want to use.

B. Catalog the Database

Before a client application can access a remote database, the database must be cataloged on the server node and on any client nodes that will connect to it. When you create a database, it is automatically cataloged on the server with the database alias (*database_alias*) the same as the database name (*database_name*). The information in the database directory, along with the information in the node directory, is used on the client to establish a connection to the remote database.

To catalog a database on the client, perform the following steps.

1 Log on to the system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority.



If you have trouble logging on to the system, see “Logging on to the System” on page 139.

2 Fill in the *Your Value* column in the following worksheet.

Table 23. Worksheet: Parameter Values for Cataloging Databases			
Parameter	**Description**	**Sample Value**	**Your Value**
Database name (*database_name*)	The database alias (*database_alias*) of the *remote* database. When you create a database, it is automatically cataloged on the server with the database alias (*database_alias*) the same as the database name (*database_name*).	`sample`	
Database alias (*database_alias*)	An arbitrary local nickname for the remote database, on the client. If you do not provide one, the default is the same as the database name (*database_name*). This is the name that you use when connecting to a database from a client.	`tor1`	
Node name (node_name)	The name of the node directory entry that describes where the database resides. Use the same value for node name (*node_name*) that you used to catalog the node in the previous step.	`db2node`	

If you are configuring communications for a Windows 3.x client, you can use the Client Setup tool to complete this task. Go to "Using the Client Setup Tool to Configure a Windows 3.x Client" on page 134 to catalog a database using the Client Setup tool.

3 If you are using a UNIX client, set up the instance environment and invoke the DB2 command line processor. Run the start-up script as follows:

```
. INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/db2cshrc (for C shell)
```

where *INSTHOME* is the home directory of the instance.

4 Catalog the database by entering the following commands in the command line processor:

```
catalog database database_name as database_alias at node node_name
terminate
```

For example, to catalog a remote database called *sample* so that it has the alias *tor1*, on the node *db2node*, use:

```
catalog database sample as tor1 at node db2node
terminate
```

If you need to change values that were set with the **catalog** *database* command, first run the **uncatalog** *database* command in the command line processor as follows:

```
uncatalog database database_alias
```

Recatalog the database with the value that you want to use.

Step 3. Test the Client-to-Server 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 140 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. Enter 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 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 tabname 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 server. See Chapter 15, “Getting Started with DB2 Universal Database” on page 139 for details.

Troubleshooting the Client-to-Server Connection

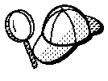
If the connection fails, check the following items:

At the *server*:

1. The *db2comm* registry value includes the value `ipxspx`. *db2comm* can be set in the `db2profile` or the `db2cshrc`script file.

Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, see Chapter 20, "Controlling Your DB2 Environment" on page 187.

2. The *FILESERVER*, *OBJECTNAME*, and *IPX_SOCKET* parameters were updated correctly in the database manager configuration file.
3. The database was created and cataloged properly.
4. If you are using File Server Addressing, ensure that the DB2 server was registered at the NetWare file server *after* the database manager configuration file was updated with the required IPX/SPX parameters. For more information, see "Configuring IPX/SPX on the Server" on page 256.
5. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).



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.

Refer to the *Troubleshooting Guide* for information on the `db2diag.log` file.

At the *client*:

1. If you are using Direct Addressing, check that the node was cataloged with a value of * for *FILESERVER*, and the correct IPX/SPX internetwork address value for the *OBJECTNAME* parameter.
2. The node name (*node_name*), specified in the database directory, points to the correct entry in the node directory.
3. The database was cataloged properly, using the *server's* database alias (*database_alias*) that was cataloged when the database was created on the server, as the database name (*database_name*) on the client.

After you verify these items, refer to the *Troubleshooting Guide* if the connection still fails.

Configuring APPC on the Client

This section assumes that APPC is functional on the client and server workstations. See “Software Requirements” on page 27 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 34 for the supported communication protocols for your particular client and server.

Refer to the online *Installing and Configuring DB2 Clients* manual for information on how to configure clients to communicate with a DB2 server using the APPC protocol.

Using the Client Setup Tool to Configure a Windows 3.x Client

The Client Setup tool provides you with a graphical interface that you can use to complete the catalog node and database instructions in Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117. This section describes how to use the Client Setup tool to catalog a node or database for a Windows 3.x client.

Use the instructions in this section, along with the protocol-specific instructions in Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117, to configure your Windows 3.x client for communications with a DB2 server.

To invoke the Client Setup tool, double-click on the **Client Setup** icon, located in the DB2 product folder for your particular platform.

Cataloging the Node

1. Select **Node->New**, the New Node window opens.
2. Fill in the node name (*node_name*) parameter in the **Node** field and optionally add a comment describing the node in the **Comment** field.
3. Select the **Protocol** radio button for the protocol that you want the Windows 3.x client to use to communicate with the server.
4. Fill in the required parameters in the **Protocol Settings** box to catalog the node for the protocol that you selected.
5. Click on **OK**.

Cataloging the Database

1. Click on the **Databases** push button, the DB2 Client Setup - Databases window opens.
2. Select **Database**->**New**, the New Database window opens.
3. Fill in the **Name** (*database_name*) and **Alias** (*database_alias*) parameters and optionally add a comment in the **Comment** field describing the database.
4. Select the **Location** radio button that describes the location of the database that you are trying to connect to.
   - If the database resides on a remote DB2 server, select the **Other Node** radio button and select the name of the Node that represents the remote server from the **At Node** drop down box.
   - If the database resides locally, select the **This node** radio button and select the drive where the database is located from the **On** drop down box.
5. Select the **Authentication Type (DB2 1.x Database)** check box if you are configuring communications to a DB2 Version 1.x database.

After you have cataloged the node and database for the Windows 3.x client, and completed the other specific steps for the protocol that you are using in Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117, you can test the database connection by clicking on the **Test Database Connection** push button.

If the connection is not successful, see the "Troubleshooting the Client to DB2 Server Connection" section for the protocol that you are trying to configure.

Part 5. Using DB2 Universal Database

Chapter 15. 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

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.

UNIX
Any valid DB2 username that belongs to the primary group of the instance owner's user ID.

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

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

See Appendix D, "Naming Rules" on page 317 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 15 MB of disk space and by default is created in the instance's home directory. It is automatically cataloged with the database alias SAMPLE when it is created.

To create the sample database, perform the following steps:

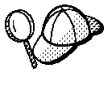
1. Log in as the instance owner.
2. Run the start up script as follows:

   ```
   . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
   source INSTHOME/sqllib/db2cshrc (for C shell)
   ```

   where INSTHOME is the home directory of the instance
3. Create the sample database by issuing the following command:

   ```
   db2sampl
   ```

   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 as the instance owner in 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:

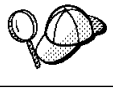
```
connect to sample
```

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 line processor:

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

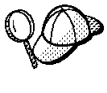
  (Be sure to choose values for *userid* and *password* that are valid on the server system.



UNIX passwords and userids are case sensitive.

- To connect a client to a remote database called SAMPLE, using CLIENT authentication, enter the following command in 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 Server        = DB2/6000 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.

UNIX
Any valid DB2 username that belongs to the primary group of the instance owner's user ID.

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.

Counting Concurrent Users

DB2 Enterprise Edition enables an unlimited number of users to connect to the server.

DB2 Workgroup Edition and DB2 Connect Enterprise Edition products are priced per user. If you need to support more users than currently entitled, you must acquire entitlements for additional users. An entitlement for additional users does not require a new license key. It is up to you, the purchaser of the product, to ensure that you have sufficient entitlements for the number of users.

To assist you in tracking the number of concurrent users connecting to a DB2 Workgroup Edition server or a DB2 Connect Enterprise Edition server, DB2 uses the License Use Runtime client. It provides a way to monitor the number of concurrent

users on your system, but it does not enforce the maximum limit; however, the runtime client will log an entry in the db2diag.log file each time the number of entitlements that you are licensed to is exceeded.

To update the number of concurrent users you are entitled to support, use the Nodelock Administration Tool (NAT) as follows:

To update the number of concurrent users you are entitled to support, use the Nodelock Administration Tool (NAT):

This tool is available only on AIX operating systems.

1. Log in as root.
2. If you are using NAT Version 3.0, append the directory /var/ifor to the PATH environment variable, for example:

   For Bourne or Korn shell:

   ```
   PATH=$PATH:/var/ifor ; export PATH
   ```

   For C shell:

   ```
   setenv PATH "${PATH}:/var/ifor"
   ```

3. Use the **/var/ifor/i4nat** command to invoke NAT.
4. Select the product that you want to monitor the number of concurrent users for.
5. Update the **Concurrent licenses** field.

Check the db2diag.log file (located in the INSTHOME/sqllib/db2dump directory, where INSTHOME is the home directory of the instance) to see if you have exceeded the maximum number of concurrent users that you are entitled to support. For example, an entry in the db2diag.log file might look like this:

```
SQL8009W The number of concurrent users of the DB2 Server
         product has exceeded the defined entitlement of
         "5". Concurrent user count is "6".
```

For information on the db2diag.log file, refer to the *Troubleshooting Guide* .

To acquire additional entitlements, you must purchase additional licenses from IBM or an IBM dealer. After you acquire the additional entitlements, use the NAT tool to update the number of concurrent users allowed.

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.

If you install a DB2 product as Try-and-Buy, and you buy a different product, you must uninstall the Try-and-Buy product and then install the new one that you have purchased. For example, if you install DB2 Workgroup Edition as a Try-and-Buy and then purchase DB2 Enterprise Edition, you must uninstall DB2 Workgroup Edition and then install DB2 Enterprise Edition. At installation time, the program will automatically install the required license.

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. Log in as root.
2. Mount the CD-ROM, if required.
3. Update your DB2 product license with the following command:

   ```
   DB2DIR/cfg/db2licinst license_filename
   ```

   where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
   = /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

   and `license_filename` is the full pathname and file name for the license file that corresponds to the product you have purchased. The names of the license files for these products are:

   db2entr.lic DB2 Universal Database Enterprise Edition

   For example, if the CD-ROM is mounted in the `/cdrom` directory and the name of the license file is `db2entr.lic`, the command should be as follows:

   - For AIX, HP-UX, or SCO UnixWare 7:

     ```
     DB2DIR/cfg/db2licinst /cdrom/db2/license/db2entr.lic
     ```

   - For Solaris:

     ```
     DB2DIR/cfg/db2licinst /cdrom_unnamed/db2/license/db2entr.lic
     ```

     where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
     = /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

The DB2 product license key can be found in the `nodelock` file, located as follows:

- For AIX:

  `/usr/lib/netls/conf/nodelock`

- For HP-UX:

  `/usr/netls/nodelock`

- For Solaris or SCO UnixWare 7:

  `/var/netls/nodelock`

Chapter 16. 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 SYSMAINT authority for each instance.
- Optimize the database manager configuration for each instance.
- 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)
- 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 *INSTHOME*/sqllib directory, where *INSTHOME* is the home directory of the instance owner.

Creating Instances

The instance owner and the group that is the System Administrative (SYSADM) group are associated with every instance. The instance owner and the SYSADM group are assigned during the process of creating the instance. One user ID can be used for only one instance. That user ID is also referred to as the *instance owner*.

Each instance owner must have a unique home directory. All of the files necessary to run the instance are created in the home directory of the instance owner's user ID. If it becomes necessary to remove the instance owner's user ID from the system, you could potentially lose files associated with the instance and lose access to data stored in this instance. For this reason, it is recommended that you dedicate an instance owner user ID to be used exclusively to run DB2.

The primary group of the instance owner is also important. This primary group automatically becomes the system administrative group for the instance and gains SYSADM authority over the instance. Other user IDs that are members of the primary group of the instance owner also gain this level of authority. For this reason, you may want to assign the instance owner's user ID to a primary group that is reserved for the administration of instances. (Also make sure that you assign a primary group to the instance owner user ID; otherwise, the system-default primary group is used.)

If you already have a group that you want to make the system administrative group for the instance, you can simply assign this group as the primary group when you create the instance owner user ID. To give other users administrative authority on the instance, add them to the group that is assigned as the system administrative group.

To separate SYSADM authority between instances, ensure that each instance owner user ID uses a different primary group. However, if you choose to have common SYSADM authority over multiple instances, you can use the same primary group for multiple instances.

You can use the DB2 Installer program or the **db2icrt** command to create an instance. See Chapter 4, “Installing Additional Products and Creating Additional Instances” on page 53 to create instances by using the DB2 Installer program. The DB2 Installer program is available for AIX, HP-UX, SCO UnixWare 7, SGI IRIX, and Solaris operating systems.

To create additional instances using the **db2icrt** command, enter:

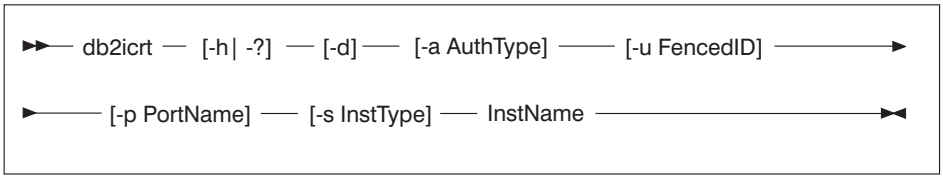
```
DB2DIR/instance/db2icrt -u FencedID instance_name
```

where *instance_name* is an alphanumeric string up to eight characters long (see “Usernames, Group Names, and Instance Names” on page 319 for more information) and

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris

Note: To create instances on a SINIX operating system, see “Create an Instance of the Product” on page 104.

The syntax of the **db2icrt** command is:



where:

-h | -? Display a help menu for this command.

-d Sets the debug mode that you can use for problem determination.

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are (SERVER), (CLIENT), and (DCS). If the *-a* parameter is not specified, the authentication type defaults to (SERVER), if a DB2 server is installed. Otherwise, the *AuthType* is set to (CLIENT).

Notes:

1. The authentication type of the instance applies to all databases owned by the instance.
2. While authentication type (DCE) is an optional parameter, it is not valid to choose (DCE) for this command.

-u *FencedID* Is the user under which the fenced UDFs and stored procedures will execute. This is not required if you install the DB2 Client Application Enabler or the DB2 Software Developer's Kit. For other products, this is a required parameter.

Note: *FencedID* may not be *root* or *bin*.

-p *PortName* Is an optional parameter that specifies the TCP/IP service name or port number to be used. This value will then be set in the instance's database configuration file.

-s *InstType* Is an optional parameter that allows different types of instances to be created. Valid instance types are *ee*, *eee* and *client*.

instance_name Is the login name of the instance owner.

If a server product is installed, the syntax is:

```
db2icrt -u FencedID instance_name
```

Examples:

- To create an instance for a DB2 server, you can use the following command:

  ```
  db2icrt -u db2fenc1 db2inst1
  ```

- If you installed the DB2 Connect Enterprise Edition only, you can use the instance name as the Fenced ID also:

  ```
  db2icrt -u db2inst1 db2inst1
  ```

- To create an instance for a DB2 client, you can use the following command:

  ```
  db2icrt db2inst1
  ```

When an instance is created, its name is also added to the list of instances on the system.

Listing Instances

To get a list of all the instances that are available on a system, enter the **db2ilist** command. The **db2ilist** command is located in:

```
DB2DIR/bin/db2ilist
```

where DB2DIR	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	HP-UX, SCO UnixWare 7, Solaris, or SGI IRIX
	= /usr/IBMdb2/V2.1	on SCO OpenServer
	= /opt/IBMdb2/V2.1	on SINIX

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

```
db2 get instance
```

Note: If you have changed to another instance, run the **db2 terminate** command before checking the instance.

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 in as the instance owner.
2. Run the start up script as follows:

   ```
   . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
   source INSTHOME/sqllib/db2cshrc (for C shell)
   ```

   where `INSTHOME` is the home directory of the instance that you want to use.
3. Start the DB2 database manager by entering the **db2start** command from a command line.

To stop a database instance:

1. Log in as the instance owner.
2. Run the start up script as follows:

   ```
   . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
   source INSTHOME/sqllib/db2cshrc (for C shell)
   ```

   where `INSTHOME` is the home directory of the instance that you want to use.
3. Stop the DB2 database manager by entering the **db2stop** command from a command line.

Auto-Starting Instances

To enable an instance to auto-start after each system restart, enter the following command:

```
DB2DIR/bin/db2iauto -on instance_name
```

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris

and where *instance_name* is the login name of the instance.

To prevent an instance from auto-starting after each system restart, enter the following command:

```
DB2DIR/bin/db2iauto -off instance_name
```

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris

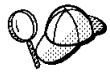
and where *instance_name* is the login name of the instance.

Setting the Operating Environment for Each Instance User

Before using DB2, the database environment for each user must be updated so that it can access an instance and run the DB2 programs. This applies to all users (including administrative users).

The sample script files, `db2profile` (for Bourne or Korn shell) and `db2cshrc` (for C shell), are provided to help you set the database environment. These scripts are available in the *INSTHOME*/`sqllib` directory, where *INSTHOME* is the home directory of the instance owner.

The instance owner (or any user that belongs to the instance's SYSADM group) can customize the script for all users of an instance. Alternatively, the script can be copied and customized for each user.



The sample script contains statements to:

- Update a user's PATH by adding the following directories to the existing search path:

  ```
  INSTHOME/sqllib/bin
  INSTHOME/sqllib/adm
  INSTHOME/sqllib/misc
  ```

  where `INSTHOME` is the home directory of the instance.
- Set the *DB2INSTANCE* environment variable to the instance name.

Setting the DB2 Environment Automatically

By default, the scripts affect the user environment for the duration of the current session only. You can change the `.profile` file to enable it to run the `db2profile` script automatically when the user logs on using the Bourne or Korn shell. For users of the C shell, you can change the `.login` file to enable it to run the `db2cshrc` script file.

Add one of the following statements to the `.profile` or `.login` script files.

- For users who share one version of the script, add:

  ```
  . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
  source INSTHOME/sqllib/db2cshrc (for C shell)
  ```

  where `INSTHOME` is the home directory of the instance that you want to use.
- For users that have a customized version of the script in their home directory, add:

```
. USERHOME/db2profile                (in Bourne or Korn Shell)
source USERHOME/db2cshrc             (in C Shell)
```

where USERHOME is the home directory of the user.

Setting the DB2 Environment Manually

To choose which instance that you want to use, enter one of the following statements at a command prompt. The period (.) and the space are required.

- For users who share one version of the script, add:

  ```
  . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
  source INSTHOME/sqllib/db2cshrc (for C shell)
  ```

  where INSTHOME is the home directory of the instance that you want to use.

- For users that have a customized version of the script in their home directory, add:

  ```
  . USERHOME/db2profile                (in Bourne or Korn Shell)
  source USERHOME/db2cshrc             (in C Shell)
  ```

  where USERHOME is the home directory of the user.

If you want to work with more than one instance at the same time, run the script for each instance that you want to use in separate windows. For example, assume that you have two instances called test and prod, and their home directories are /u/test and /u/prod.

In window 1:

- In Bourne or Korn Shell, enter:

  ```
  . /u/test/sqllib/db2profile
  ```

- In C Shell, enter:

  ```
  source /u/test/sqllib/db2cshrc
  ```

In window 2:

- In Bourne or Korn Shell, enter:

  ```
  . /u/prod/sqllib/db2profile
  ```

- In C Shell, enter:

  ```
  source /u/prod/sqllib/db2cshrc
  ```

Use window 1 to work with the test instance and window 2 to work with the prod instance.

Enter the **which db2** command to ensure that your search path has been set up correctly. This command returns the absolute path of the DB2 CLP executable. Verify that it is located under the instance's sqllib directory.

Updating Instances

Existing instances are designed to be as independent as possible from the effects of subsequent installation and removal of products.

In most cases, existing instances automatically inherit or lose access to the function of the product being installed or removed. However, if certain executables or components are installed or removed, existing instances do not automatically inherit the new system configuration parameters or gain access to all the additional function. The instance must be updated.

If DB2 is updated by installing a PTF or a patch, all the existing DB2 instances should be updated using the **db2iupdt** command. You should also update the Administration Server using the **dasiupdt** command. See "Updating the Administration Server" on page 159 for further information.

Running the **db2iupdt** command updates the specified instance by performing the following:

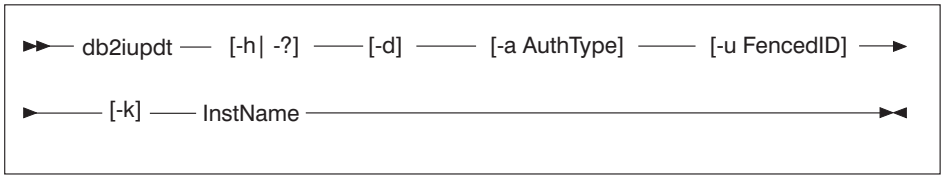
1. Replaces the files in the `INSTHOME/sqllib` directory, where `INSTHOME` is the home directory of the instance.
2. If the node type is changed, then a new database manager configuration file is created. This is done by merging relevant values from the existing database manager configuration file with the default database manager configuration file for the new node type. If a new database manager configuration file is created the old file is backed up to:

   ```
   INSTHOME/sqllib/backup/db2systm.old
   ```

The **db2iupdt** command is available in the `DB2DIR/instance` directory,

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris

The syntax of the **db2iupdt** command is:



where:

-h | -? Displays a help menu for this command.

-d Sets the debug mode, which you can use for problem analysis.

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are (SERVER), (CLIENT), and (DCS). If the *-a* parameter is not specified, the authentication type defaults to (SERVER), if a DB2 server is installed. Otherwise, the *AuthType* is set to (CLIENT).

Notes:

1. The authentication type of the instance applies to all databases owned by the instance.
2. While authentication type (DCE) is an optional parameter, it is not valid to choose (DCE) for this command.

-u *FencedID* Is the user under which the fenced UDFs and stored procedures will run. This is not required if you install the DB2 Software Developer's Kit or the DB2 Client Application Enabler. For other products, this is a required parameter.

Note: *FencedID* may not be root or bin.

-k Is an optional parameter that preserves the current instance type. If you do not specify this parameter, the current instance will be upgraded to the highest instance type available in the following order:

- Partitioned database server with local and remote clients (DB2 Extended Enterprise Edition default instance type)
- Database Server with local and remote clients (DB2 Universal Database Enterprise Edition default instance type)
- Client (DB2 Client Application Enabler default instance type)

instance_name Is the login name of the instance owner.

If a server product is installed, the syntax is:

```
db2iupdt -u FencedID instance_name
```

Examples:

- If you installed DB2 Universal Database Workgroup Edition or DB2 Universal Database Enterprise Edition after the instance was created, enter the following command to update that instance:

  ```
  db2iupdt -u db2fenc1 db2inst1
  ```

- If you installed DB2 Connect Enterprise Edition after creating the instance, you can use following command to update that instance:

  ```
  db2iupdt -u db2inst1 db2inst1
  ```

- To update client instances, you can use the following command:

  ```
  db2iupdt db2inst1
  ```

The **db2iupdt** command updates the files in the INSTHOME/sqllib directory, where INSTHOME is the home directory of the instance owner.

Removing Instances

To remove an instance, perform the following steps:

1. Log in as the instance owner.
2. Stop the instance by issuing the **db2stop** command.
3. Back up files in the INSTHOME/sqllib directory, if needed.

   You might want to save the database manager configuration file, db2systm, the db2nodes.cfg file, or user defined function or fenced stored procedure applications in INSTHOME/sqllib/function, where INSTHOME is the home directory of the instance owner.
4. Log off as the instance owner.
5. Log in as user with root authority.
6. Remove the instance using the **db2idrop** command. You can enter the DB2IDROP command as follows:

   ```
   DB2DIR/instance/db2idrop InstName
   ```

   InstName is the login name of the instance.

where DB2DIR	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	HP-UX, SCO UnixWare 7, Solaris, or SGI IRIX
	= /usr/IBMdb2/V2.1	on SCO OpenServer
	= /opt/IBMdb2/V2.1	on SINIX

7. Optionally, as a user with root authority, remove the instance owner's user ID and group (if used only for that instance). Do not remove these if you are planning to re-create the instance.

   This step is optional since the instance owner and the instance owner group may be used for other purposes.

The **db2idrop** command removes the instance entry from the list of instances and removes the INSTHOME/sqllib directory.

Chapter 17. 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.

Creating the Administration Server

Use the **dasicrt** command to create an Administration Server. You must have root authority to issue this command.

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

- On AIX:

  ```
  /usr/lpp/db2_05_00/instance/dasicrt ASName
  ```

- On HP-UX, SCO UnixWare 7 or Solaris:

  ```
  /opt/IBMdb2/V5.0/instance/dasicrt ASName
  ```

where *ASName* is the name of the Administration Server, which is composed of a string of up to eight alphanumeric characters long. See Appendix D, “Naming Rules” on page 317 for more information.

Note: If you are running NIS and NIS+, you need to set up the user and group names in such a way that:

- The primary group of the Administration Server must be in the secondary group of all the instances.
- The secondary group of the Administration Server must contain the primary group of all the instances

Secondary group lists will be modified automatically only if NIS and NIS+ is not running on your system.

Because a user ID can only own one instance, you must have a separate user ID to own each Administration Server that you create.

Starting the Administration Server

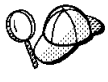
To start the Administration Server, you must perform the following steps:

1. Log in as the Administration Server owner.
2. Run the start up script as follows:

   ```
   . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
   source INSTHOME/sqllib/db2cshrc (for C shell)
   ```

   where *INSTHOME* is the home directory of the instance.
3. Start the Administration Server using the **db2admin** command as follows:

   ```
   db2admin start
   ```



The Administration Server is automatically started after each system reboot.

Stopping the Administration Server

To stop the Administration Server, you must perform the following steps:

1. Log in as the Administration Server owner.
2. Run the start up script as follows:

   ```
   . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
   source INSTHOME/sqllib/db2cshrc (for C shell)
   ```

   where *INSTHOME* is the home directory of the instance.
3. Stop the Administration Server using the **db2admin** command as follows:

   ```
   db2admin stop
   ```

where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
= /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

Listing the Administration Server

To obtain the name of the Administration Server on your machine, issue the **dasilist** command as follows:

```
DB2DIR/bin/dasilist
```

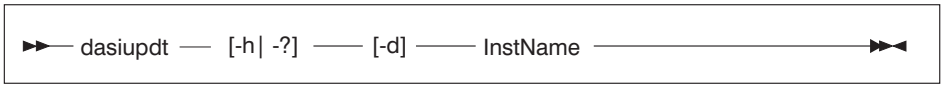
where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
= /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

Updating the Administration Server

If DB2 is updated by installing a PTF or a patch, all Administration Servers as well as all existing instances should be updated. To update an Administration Server, use the **dasiupdt** command available in the `DB2DIR/instance` directory,

where *DB2DIR* = /usr/lpp/db2_05_00 on AIX
= /opt/IBMdb2/V5.0 on HP-UX, SCO UnixWare 7, or Solaris

The syntax of the **dasiupdt** command is:



where:

-h | -? Displays a help menu for this command.

-d Sets the debug mode, which you can use for problem analysis.

InstName Is the login name of the instance owner.

Removing the Administration Server

To remove an Administration Server, you must perform the following steps:

1. Log in as the Administration Server owner.
2. Run the start up script as follows:

   ```
   . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
   source INSTHOME/sqllib/db2cshrc (for C shell)
   ```

   where *INSTHOME* is the home directory of the instance.
3. Stop the Administration Server using the **db2admin** command as follows:

   ```
   db2admin stop
   ```
4. Backup the files in the ASHOME/sqllib directory, if needed, where ASHOME is the home directory of the Administration Server.
5. Log off.
6. Log in as *root* and remove the Administration Server using the **dasidrop** command as follows:

   ```
   DB2DIR/instance/dasidrop ASName
   ```

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris

   and ASName is the name of the instance being removed.

   Note: The **dasidrop** command removes the sqllib directory under the home directory of the Administration Server.

Chapter 18. 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. 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: `INSTHOME/sqllib/bnd`
  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 clibind.msg grant public"
     ```

     In this example, `bind.msg` and `clibind.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 161 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.

After the platform specific instructions there are further details on the following topics:

- "How to Bind the DB2 CLI/ODBC Driver to the Database" on page 174
- "How to Set CLI/ODBC Configuration Keywords" on page 174
- "Configuring db2cli.ini" on page 175

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 165
- "OS/2 Client Access to DB2 using CLI/ODBC" on page 167
- "UNIX Client Access to DB2 using CLI/ODBC" on page 169

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 265 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* online manual.

   For information on manually editing this file (`db2cli.ini`), see "Configuring db2cli.ini" on page 175.

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\MSODBC16` 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.

If you require additional information at this point you can refer to the following topics in "Detailed Configuration Information" on page 173:

- "How to Bind the DB2 CLI/ODBC Driver to the Database" on page 174
- "How to Set CLI/ODBC Configuration Keywords" on page 174
- "Configuring db2cli.ini" on page 175

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 265.

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.

For information on manually editing this file (`db2cli.ini`), see “Configuring db2cli.ini” on page 175.

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.

If you require additional information at this point you can refer to the following topics in "Detailed Configuration Information" on page 173:

- "How to Bind the DB2 CLI/ODBC Driver to the Database" on page 174
- "How to Set CLI/ODBC Configuration Keywords" on page 174
- "Configuring db2cli.ini" on page 175

UNIX Client Access to DB2 using CLI/ODBC

Before DB2 CLI and ODBC applications can successfully access a DB2 database from a UNIX 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 command line processor.

For more information see Chapter 14, "Configuring Client-to-Server Communications Using the Command Line Processor" on page 117 or the **CATALOG DATABASE** and **CATALOG NODE** commands in the *Command Reference*.

2 The DB2 CLI/ODBC driver is automatically installed when DB2 is installed, and therefore nothing needs to be done.

3 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 When using an ODBC application you must ensure that the Visigenic or Intersolv ODBC Driver Manager components are installed and that each user that will use ODBC has access to it.

- If DB2 installed the Driver Manager it would be located in the `sqllib/odbclib` subdirectory.
- If the Driver Manager was installed by another program, the file `.odbc.ini` (begins with a period) would be located in the home directory of the user ID that runs the ODBC application.
- DB2 does not install an ODBC Driver Manager on the SCO UnixWare operating system. You must use the ODBC Driver Manager that was supplied with your ODBC client application or ODBC SDK in order to access DB2 data using that application.

b The Driver Manager uses two initialization files. A sample template of these files are provided in the `sqllib/odbclib` subdirectory.

`odbcinst.ini` ODBC Driver Manager's configuration file indicating which database drivers are installed. Each user that will use ODBC must have access to this file.

`.odbc.ini` End-user's data source configuration. Each user has a separate copy of this file. The sample template file is called odbc.ini (without the first dot).

Setting up odbcinst.ini

The settings in this file impact all of the ODBC drivers on the machine.

Use an ASCII editor to update this file. It must have a stanza (section) called [IBM DB2 ODBC DRIVER], with a line starting with "Driver" indicating the full path to the DB2 ODBC driver (`db2.o`). For example, if the home directory of your end user is `/u/thisuser/` and the sqllib directory is installed there, then the correct entry would be:

```
[IBM DB2 ODBC DRIVER]
Driver=/u/thisuser/sqllib/lib/db2.o
```

See the sample file in the `sqllib/odbclib` subdirectory for an example.

Setting up odbc.ini

The settings in this file are associated with a particular user on the machine; different users can have different `odbc.ini` files.

The `odbc.ini` file must be copied into the end user's home directory and called `.odbc.ini`. Update this file, using an ASCII editor, to reflect the appropriate data source configuration information. To register a DB2 database as an ODBC data source there must be one stanza (section) for each DB2 database.

The sample `odbc.ini` template shows:

- line 2 (part of the [ODBC Data Source] stanza):

  ```
  SAMPLE=IBM DB2 ODBC DRIVER
  ```

  Indicates that there is a data source called SAMPLE that used the IBM DB2 ODBC DRIVER.

- lines 4-6 (part of the [SAMPLE] stanza):

  ```
  [SAMPLE]
  Driver=/u/thisuser/sqllib/lib/db2.o
  Description=Sample DB2 ODBC Database
  ```

  Indicates that the SAMPLE database is part of the DB2 instance located in the directory `/u/thisuser`.

- line 11 (part of the [ODBC] stanza):

  ```
  InstallDir=/u/thisuser/sqllib/odbclib
  ```

  Indicates that `/u/thisuser/sqllib/odbclib` should be treated as the location where ODBC is installed.

  Note: If your application specifically installed the ODBC components (such as the Driver Manager) elsewhere, you must update the line starting with InstallDir under the [ODBC] stanza to reflect this new location.

For example, if ODBC has been installed in `/opt/odbc`, the [ODBC] stanza would look like:

```
[ODBC]
Trace=0
TraceFile=odbctrace.out
InstallDir=/opt/odbc
```

See the sample file in the `sqllib/odbclib` subdirectory for an example. You can also see "How to Configure ODBC.INI" on page 175 for more detailed information.

Once the .ini files are set up you can run your ODBC application and access DB2 databases. Refer to the documentation that comes with your ODBC application for additional help and information.

4 Configure the DB2 CLI/ODBC driver (**optional**).

There are various keywords and values that can be used to modify the behavior of DB2 CLI/ODBC and the applications using it. The keywords are associated with the database *alias name*, and affect all DB2 CLI/ODBC applications that access the database.

For information on manually editing this file (`db2cli.ini`), see "Configuring db2cli.ini" on page 175. For information about the specific keywords see the *CLI Guide and Reference*.

If you require additional information at this point you can refer to the following topics in "Detailed Configuration Information" on page 173:

- "How to Bind the DB2 CLI/ODBC Driver to the Database" on page 174
- "How to Set CLI/ODBC Configuration Keywords" on page 174
- "Configuring db2cli.ini" on page 175

Macintosh Client Access to DB2 using CLI/ODBC

Before DB2 CLI and ODBC applications can successfully access a DB2 database from a Macintosh 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 command line processor.

For more information see "Configuring Database Connections" on page 265.

2 If you are using ODBC applications to access DB2 data, perform the following steps. (If you are using only CLI applications, skip to step #4.)

a An ODBC driver manager must be installed to handle the communications between the ODBC applications and the DB2 ODBC driver. DB2 does not supply a Macintosh ODBC driver manager, only the DB2 ODBC driver itself.

If your ODBC application does not supply an ODBC driver manager then you can obtain the MacODBC driver manager directly from Apple.

b Register the DB2 database with the ODBC driver manager list of data sources (contained in the ODBC Preferences file) as a *data source* using either of these methods:

- Running **ODBC Setup** (in a 68K environment) or **ODBC Setup PPC** (in a PowerMacintosh environment) from the Control Panel
- Editing the initialization (ODBC Preferences or ODBC Preferences PPC) file directly; refer to the next step in this process for more information.

3 Configuring ODBC Preferences or ODBC Preferences PPC: The ODBC initialization file is used to record information such as the available drivers and data sources. See the documentation for your driver manager for procedures on updating this file.

The MacODBC driver manager uses the ODBC Preferences file or the ODBC Preferences PPC file to record information about the available drivers and data sources.

It is also possible to modify these files manually. Do not change any of the existing entries in the files.

a Use an ASCII editor to edit the `ODBC Preferences` file or the `ODBC Preferences PPC` file.

68K Environment

The following is a sample `ODBC Preferences` file:

```
[ODBC Data Sources]
GLOBALDB=IBM ODBC DB2 for Macintosh
SAMPLE=IBM ODBC DB2 for Macintosh

[GLOBALDB]
Driver=appl:ODBC$DB2DriverFunctionSet
Description=My GLOBAL database

[SAMPLE]
Driver=appl:ODBC$DB2DriverFunctionSet
Description=My SAMPLE database
```

PowerMacintosh Environment

The following is a sample `ODBC Preferences PPC`:

```
[ODBC Data Sources]
SAMPLE=IBM ODBC DB2 for PPC

[SAMPLE]
Driver=DB2ODBCDriverPPC
Description=My SAMPLE database
```

b The [ODBC Data Source] section lists the name of each available data source and the description of the associated driver.

For each data source listed in the [ODBC Data Source] section, there is a section that lists additional information about that data source. These are called the *Data Source Specification* sections.

Under the `[ODBC DATA SOURCE]` entry, add the following line:

68K Environment
database_alias=IBM ODBC DB2 for Macintosh

PowerMacintosh Environment
database_alias=IBM ODBC DB2 for PPC

where *database_alias* is the alias of the database cataloged in the database directory (the database name used by the Command Line Processor CONNECT TO statement).

c Add a new entry in the Data Source Specification section to associate the data source with the driver:

68K Environment

```
[database_alias]
Driver=appl:ODBC$DB2DriverFunctionSet
```

PowerMacintosh Environment

```
[database_alias]
Driver=DB2ODBCDriverPPC
```

Where *database_alias* is the alias of the database cataloged in the database directory, and listed under the Data Source Specification section.

4 The DB2 CLI/ODBC driver can be configured further by editing the `db2cli.ini` file. This file contains various keywords and values that can be used to modify the behavior of DB2 CLI and the applications using it. The keywords are associated with the database *alias name*, and affect all DB2 CLI applications that access the database. For a complete description of all the keywords and their usage, refer to an appendix in the *Installing and Configuring DB2 Clients* manual.

5 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 164 should provide you with all of the information you require. The following additional information is useful where DB2 tool support is not available, and for administrators who require more detailed information.

The following topics are covered in this section:

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

The CLI/ODBC driver will autobind on the first connection to the database, provided the user has the appropriate privilege or authorization. The administrator may want to perform the first connect or explicitly bind the required files.

See "Binding Database Utilities" on page 161 for more information.

How to Set CLI/ODBC Configuration Keywords

DB2 CLI can be configured further by using either the CCA or the DB2 Client Setup administration tool, whichever is applicable for your platform, or by manually editing the `db2cli.ini` file.

This file contains various keywords and values that can be used to modify the behavior of DB2 CLI and the applications using it. The keywords are associated with the database *alias name*, and affect all DB2 CLI and ODBC applications that access the database.

By default, the location of the CLI/ODBC configuration keyword file is as follows:

Platform:	Location:
OS/2	`sqllib` directory
Windows NT	`sqllib` directory
Windows 95	`sqllib` directory
Windows 98	`sqllib` directory
Windows 3.x	`sqllib\win` directory
UNIX	`sqllib/cfg` directory of the database instance running the CLI/ODBC applications

The environment variable DB2CLIINIPATH can also be used to override the default and specify a different location for the file.

The configuration keywords enable you to:

- Configure general features such as data source name, user name, and password.
- Set options that will affect performance.
- Indicate query parameters such as wild card characters.
- Set patches or work-arounds for various ODBC applications.
- Set other, more specific features associated with the connection, such as code pages and IBM Graphic data types.

For a complete description of all the keywords and their usage, refer to the *Installing and Configuring DB2 Clients*.

Configuring db2cli.ini: The `db2cli.ini` initialization file is an ASCII file which stores values for the DB2 CLI configuration options. A sample file is shipped to help you get started. Refer to the *CLI Guide and Reference* for information on each keyword.

See "Platform Specific Details for CLI/ODBC Access" on page 164 for more information on how to modify this file on your platform.

How to Configure ODBC.INI

Microsoft's 16-bit ODBC Driver Manager and Visigenic's ODBC Driver Manager use the `odbc.ini` file to record information about the available drivers and data sources. Visigenic's ODBC Driver Manager also uses the `odbcinst.ini` file on UNIX platforms. Although the necessary files are updated automatically by the tools on most platforms, users of ODBC on UNIX platforms will have to edit them manually. The file `odbc.ini` (and `odbcinst.ini` where required) are located:

Platform:	Location:
Windows	*x*:\windows (where *x* is the drive where Windows is installed)
WIN-OS/2	*x*:\os2\mdos\winos2 (where *x* is the drive where OS/2 is installed)
UNIX	Home directory of user ID running ODBC application (on UNIX the odbc.ini file name has a dot before it: .odbc.ini)

It is also possible to modify this file manually. Do not change any of the existing entries in the file.

1 Use an ASCII editor to edit the `odbc.ini` file.

The following is an example odbc.ini file:

```
[ODBC Data Sources]
MS Access Databases=Access Data (*.mdb)

[MS Access Databases]
Driver=D:\WINDOWS\SYSTEM\simba.dll
FileType=RedISAM
SingleUser=False
UseSystemDB=False
```

The [ODBC Data Sources] section lists the name of each available data source and the description of the associated driver.

For each data source listed in the [ODBC Data Sources] section, there is a section that lists additional information about that data source. These are called the *Data Source Specification* sections.

2 Under the [ODBC DATA SOURCE] entry, add the following line:

database_alias=IBM DB2 ODBC DRIVER

where *database_alias* is the alias of the database cataloged in the database directory (the database name used by the command line processor CONNECT TO statement).

3 Add a new entry in the Data Source Specification section to associate the data source with the driver:

```
[database_alias]
Driver=x:\windows\system\db2cliw.dll
```

Where:

- *database_alias* is the alias of the database cataloged in the database directory, and listed under the Data Source Specification section.
- *x* is the drive where Windows is installed.

The following shows the example file with the IBM data source entries added:

```
[ODBC Data Sources]
MS Access Databases=Access Data (*.mdb)
SAMPLE=IBM DB2 ODBC DRIVER

[MS Access Databases]
Driver=D:\WINDOWS\SYSTEM\simba.dll
FileType=RedISAM
SingleUser=False
UseSystemDB=False

[SAMPLE]
Driver=D:\WINDOWS\SYSTEM\db2cliw.dll
Description=Sample DB2 Client/Server database
```

Note: If you are running the ODBC application under WIN-OS/2, specify the equivalent path \OS2\MDOS\WINOS2\SYSTEM in place of the \WINDOWS\SYSTEM path.

UNIX Configuration of .ini files

The section "UNIX Client Access to DB2 using CLI/ODBC" on page 169 contains detailed steps on how to update both the odbc.ini and odbcinst.ini files.

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 `sqllib/java/db2java.zip`
- On AIX, SCO UnixWare, and Solaris: LD_LIBRARY_PATH includes the directory `sqllib/lib`; on HP-UX: SHLIB_PATH includes the directory `sqllib/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:

- `sqllib/java/sqlj.zip` (required to build SQLJ programs)
- `sqllib/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 `sqllib/java/db2java.zip` file into the same directory as these other files. For SQLJ applets, also copy the `sqllib/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 19. 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**
Macintosh	• command line processor
OS/2	• Command Center • command line processor • command prompt
UNIX	• command line processor • command line prompt
Windows 3.x	• command line processor
Windows 32-bit operating systems	• 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.

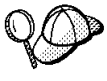
To invoke the Command Center on OS/2:

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
```

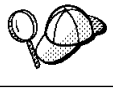
To enter operating system commands in interactive mode, precede the operating-system command with an exclamation mark (!). For example:

```
!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 *Command Reference* 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.
- On Macintosh:
  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:

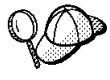
```
db2 =>
```

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 UNIX, 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*.

Chapter 20. 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 188.
- To set DB2 environment variables, see:
  - “Setting Your Environment on OS/2” on page 189.
  - “Setting Your Environment on Windows 32-Bit Operating Systems” on page 190.
  - “Setting Your Environment on UNIX Systems” on page 191.
  - “Setting Your Environment on Windows 3.x” on page 191.
- To set database manager configuration parameters on a client instance, see “Configuration Parameters” on page 198.

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. On UNIX platforms, changing your DB2 environment required you to change the script files, `db2profile` or `db2cshrc` in the home directory of the instance.

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 191 for more information.

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

- The environment variable settings, or a variable set in the script files, *db2profile* or *db2cshrc*.
- 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 193 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 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.

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'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
  ```

  Notes:

  1. 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*.
  2. On UNIX, you must have system administrative (SYSADM) authority to change registry values for an instance. Only users with root authority can change parameters in global-level registries.

- 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 instance level, enter:

  ```
  db2set parameter= -g instance
  ```

- 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 32-bit operating systems have 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.

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 UNIX 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.

On UNIX, you must set the system environment variable *DB2INSTANCE*.

The sample script files, `db2profile` (for Bourne or Korn shell) and `db2cshrc` (for C shell), are provided to help you set the database environment. These scripts are available in the *INSTHOME*/`sqllib` directory, where *INSTHOME* is the home directory of the instance owner.

Note: Except for *PATH* and *DB2INSTANCE*, all other DB2-supported variables must be set in the DB2 profile registry. To set variables that are not supported by DB2, define them in your script files, `db2profile` or `db2cshrc`.

An instance owner or SYSADM user may customize these script files for all users of an instance. Alternatively, you can copy and customize a script file, then invoke a script directly or add it to your `.profile` or `.login` file.

To work with another instance for the current session, execute the `db2profile` script file (for Bourne shell or Korn shell) or the `db2cshrc` script file (for C shell) in the `sqllib` directory of the instance's home directory.

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 file `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 191 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 *DB2 Connect User's Guide* for details.
DB2BIDI	All	Default: (not set)	Enables bi-directional CCSID processing by UDB. Refer to the *DB2 Connect User's Guide* 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 *db2codepage* unless explicitly stated in DB2 documents, or asked to do so by DB2 service. Setting *DB2CODEPAGE* to a value not supported by the operating system can produce unexpected results. Normally, you do not need to set *DB2CODEPAGE* 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.

Parameter	Operating System	Values	Description
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.
DB2ENVLIST	UNIX	Default: (not set)	Lists specific variable names for either stored procedures or user defined functions. By default, the **db2start** command filters out all user environment variables except those prefixed with **DB2** or **db2**. If specific environment variables must be passed to either stored procedures or user defined functions, you can list the variable names in the *db2envlist* environment variable. Separate each variable name by one or more spaces. DB2 constructs its own PATH and LIBPATH, so if PATH or LIBPATH is specified in *db2envlist*, the actual value of the variable name is appended to the end of the DB2-constructed value. For example, LIBPATH becomes DB2LIBPATH.
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 *Embedded SQL Programming Guide* for descriptions of how *db2include* 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 *DB2INSTANCE* is not defined.

Parameter	Operating System	Values	Description
DB2LIBPATH	UNIX	Default: (not set)	Specifies the value of LIBPATH in the *db2libpath* environment variable. On UNIX operating systems, the value of LIBPATH cannot be inherited between parent and child processes if the user ID has changed. Since the **db2start** executable is owned by root and its execute permissions are setuid to root, DB2 cannot inherit the LIBPATH settings of end users. If you list the variable name, LIBPATH, in the *db2envlist* environment variable, you must also specify the value of LIBPATH in the *db2libpath* registry value. The **db2start** executable will then read the value of *db2libpath* and append this value to the end of the DB2-constructed LIBPATH.
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 *db2slogon*=YES DB2 does not write user IDs and passwords to a file, but instead uses a segment of memory to maintain them. When *db2slogon* 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 *db2timeout* is set to a value between 1 and 30, the default minimum value will be used. If *db2timeout* is set to a value of 0, or a negative value, the timeout feature is disabled. This feature is disabled by default.
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 *db2tracename*=DB2WIN.TRC, and is saved in your current instance directory (for example, \sqllib\db2). The default on Macintosh is *db2tracename*=DB2MAC.TRC. We strongly recommend that you specify the full path name when naming the trace file.

Parameter	Operating System	Values	Description
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 *Troubleshooting Guide* 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, *db2trcflush* can be used in conjunction with *db2traceon*=Y. *db2trcflush*=Y 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 *db2trcflush*=N. 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.
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 *db2yield*=YES. If your system crashes, you will need to set *db2yield*=NO. 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=*DB2INSTDEF* 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 *DB2INSTANCE*.

Parameter	Operating System	Values	Description
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 *DB2PATH*.
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.
Communications			
DB2COMM	All, server only	Default (not set), values: any combination of APPC, IPXSPX, NETBIOS, 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.
DCE Directories			
DB2DIRPATHNAME	OS/2 and UNIX	Default: (not set)	Specifies a temporary override of the DIR_PATH_NAME parameter value in the database manager configuration file. If a directory server is used and the target of a CONNECT statement or ATTACH command is not explicitly cataloged, then the target is concatenated with *db2dirpathname* (if specified) to form the fully qualified DCE name. **Note:** The *db2dirpathname* value has no effect on the instance's global name, which is always identified by the database manager configuration parameters DIR_PATH_NAME and DIR_OBJ_NAME.
DB2CLIENTCOMM	OS/2 and UNIX	Default: (not set)	Specifies a temporary override of the DFT_CLIENT_COMM parameter value in the database manager configuration file. If both DFT_CLIENT_COMM and *db2clientcomm* are not specified, then the first protocol found in the object is used. If either one or both of them are specified, then only the first matching protocol will be used. In either case, no retry is attempted if the first connect fails.

Parameter	Operating System	Values	Description
DB2ROUTE	OS/2 and UNIX	Default: (not set)	Specifies the name of the Routing Information Object the client uses when it connects to a database with a different database protocol. The *db2route* value overrides the ROUTE_OBJ_NAME parameter value in the database manager configuration file.

Note: There are three registry values that are related to ADSM:

- *DSMI_CONFIG*
- *DSMI_DIR*
- *DSMI_LOG*

Refer to the *Administration Guide* for information about ADSM and these values.

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.

Database configuration parameters can be set only on the server, or remotely from the Control Center.

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.

To control database management configuration parameters on a client instance on UNIX, use the command line processor.

- For information on using Client Configuration Assistant, see Chapter 27, “Using the Client Configuration Assistant” on page 265.
- For information on using the command line processor, see “Using the Command Line Processor” on page 183.

Part 6. Advanced Installation and Configuration

Chapter 21. Other Methods to Install DB2 for AIX

This section describes the other methods you can use to install DB2 Universal Database products. Some of the advanced methods to install DB2 are detailed in this section. The DB2 Installer program can perform all of the steps described in this section.

We recommend that you use the DB2 Installer program when installing DB2, but if you prefer not to use the DB2 Installer program, you can use any one of the methods described in this chapter. See Chapter 3, “Installing and Configuring DB2” on page 39 for information on how to use the DB2 Installer program to install the DB2 product. See Chapter 7, “Installing DB2 Clients” on page 79 for information on how to install the DB2 clients.

Installation Steps

To install DB2, perform the following steps:

1. Install DB2. As part of this installation step, the following installation activities are optional:
   a. Install the DB2 product online documentation.
   b. Install the DB2 product translated messages.
2. Create or assign groups and user IDs.
3. Create a DB2 instance.
4. Create the Administration Server.
5. Install the license key.
6. Create links for DB2 files (optional).

You can install DB2 using any of the following methods:

Using dsh
Allows you to simultaneously install DB2 on multiple RS/6000 SP nodes.

Using SMIT
Allows you to install a subset of products and filesets on a single machine. If you want to install only a selected set of DB2 filesets or components, see “Install Products or Filesets Using SMIT” on page 204.

Installing software bundles using SMIT
A software bundle contains a list of filesets that are suited for a particular use. Installation, although easier with software bundles, only occurs on a single machine.

If you prefer to use the **installp** command, you must first mount the DB2 CD-ROM. You also need to mount the CD-ROM if you want to view the *Installation Notes* or print the postscript manuals prior to installation.

Step 1. Install DB2

The DB2 CD-ROM contains several products. Select the products you are licensed to install. *Your Proof of Entitlement* and *License Information* booklet identify the products for which you are licensed.

Go to the installation method that you would like to use to install DB2.

- “Install Products or Filesets Using SMIT.”
- “Install Software Bundles Using SMIT” on page 205.
- Chapter 3, “Installing and Configuring DB2” on page 39.

Install Products or Filesets Using SMIT

1. Log in as user with root authority.
2. Insert and mount the DB2 CD-ROM.
3. Enter the **smit install_latest** command. The Software Installation and Maintenance menu opens.
4. Specify the `INPUT device/directory` for DB2 and press Enter.

   For example, if `/cdrom` is the CD-ROM mount directory, enter the following as the INPUT device:

   ```
   /cdrom/db2
   ```

If you are installing the DB2 Client Application Enabler or the DB2 Software Developer's Kit, enter the following as the INPUT device:

```
/cdrom/db2/aix
```

5. Press F4 to display a list of software to install. For a detailed description of the software packages, refer to Appendix B, "Contents of the DB2 Products" on page 309 for more information.
6. Press F7 to mark one or more packages and filesets for installation.
7. Press Enter to install DB2.
8. Press F10 to exit when the command displays **OK**.

After the installation is complete, DB2 will be installed in the `/usr/lpp/db2_05_00` directory on that machine; now you must repeat the above steps on all machines.

Now that you have finished installing DB2, go to "Install the DB2 Product Library (Optional)" on page 206 and "Install the DB2 Product Messages (Optional)" on page 207 to optionally install the product documentation and messages. If you do not want to install product documentation or messages, proceed to the next step.

Install Software Bundles Using SMIT

The following procedure shows how to install DB2 Extended Enterprise Edition using the *Software Bundles* feature. A software bundle is a collection of filesets. When you install a software bundle, every fileset in that bundle is installed.

1. Log in as root.
2. Insert and mount the DB2 CD-ROM in the drive.
3. To proceed directly to the automatic bundle processing application, issue the **smit easy_install_bundle** command.
4. Specify the installation device or directory for the installation media by pressing F4 to display a list.
5. Select the CD-ROM drive in which you placed the DB2 CD-ROM, then press Enter.
6. If you did not install the DB2 Media-Defined software bundle on this system, use the following procedure to install it:
   a. In the **Bundle to Install** menu, select the **Media-Defined** option, then press Enter.

b In the **Install Bundle Contents** menu, press Enter to continue with the installation. Messages may appear to indicate the status of the installation. Press Enter to continue.

c Press F3 to return to the **Install Bundle Contents** menu.

d Press F4 to re-display the list of software bundles.

7 From the displayed menu, select one of the following software bundles:

- DB2V5-UDB-Enterprise
- DB2V5-Connect-Enterprise
- DB2V5-SDK
- DB2V5-Client

Notes:

a. Every required and recommended fileset for a product is installed with its bundle.

b. For a list of filesets included in a product, see Table 6 on page 36.

8 Press Enter to start the installation.

Messages will appear to indicate the status of the installation. Press Enter to continue, or F3 if you want to cancel.

9 Repeat the steps in this section if you want to install software bundles on other machines.

Now that you have finished installing DB2, go to "Install the DB2 Product Library (Optional)" and "Install the DB2 Product Messages (Optional)" on page 207 to optionally install the product documentation and messages. If you do not want to install product documentation or messages, proceed to the next step.

Install the DB2 Product Library (Optional)

When you install DB2 using SMIT, the DB2 Product Library (HTML) is not automatically installed.

To install the DB2 Product Library, do the following:

1 Log in as user with root authority.

2 Insert and mount the DB2 CD-ROM.

3 Enter the **smit install_latest** command. The Software Installation and Maintenance menu opens.

4 Enter the directory for the **INPUT device/directory for software**. For example, if `/cdrom` is the CD-ROM mount directory, enter the following as the INPUT device:

/cdrom/db2

5. Press F4 to display a list of software to install. For a detailed description of the software packages, refer to Appendix B, “Contents of the DB2 Products” on page 309 for more information.

6. Select the fileset for the DB2 Product Library (HTML) that you want to install. A separate fileset exists for every language in which DB2 documentation is translated.

   Note: Not every manual is translated into multiple languages. The *DB2 Product Library (HTML) - English* fileset contains a complete set of books in English. When you select any non-English HTML fileset without selecting English, the English HTML fileset is also installed.

The DB2 Product Library (HTML) fileset is listed as follows:

`DB2 Product Library (HTML) - language`

7. Press ENTER to start the installation.

8. Press F10 to exit SMIT when the command completes.

9. Run the following command to uncompress and untar the HTML files:

   ```
   /usr/lpp/db2_05_00/doc/db2insthtml locale
   ```

   where *locale* is the desired locale; for example, locale=en_US for English.

10. From your Web browser, open the following file URL to view DB2 online manuals:

    ```
    file:/usr/lpp/db2_05_00/doc/en_US/html/index.htm
    ```

To recover some disk space, you may want to remove the compressed tar files. It is recommended that you do not just remove the compressed tar HTML files; instead, you should remove the fileset for the DB2 Product Library.

After the fileset is removed, only compressed tar files will be removed. You will still be able to view the online DB2 books.

Install the DB2 Product Messages (Optional)

When you install DB2 using SMIT, DB2 product messages for locales, other than en_US (English - ISO 88591), are not automatically installed.

To install DB2 product messages in other locales, do the following:

1. Log in as user with root authority.

2. Insert and mount the DB2 CD-ROM.

3. Enter the **smit install_latest** command. The Software Installation and Maintenance menu opens.

4. Enter the directory for the **INPUT device/directory for software**. For example, if `/cdrom` is the CD-ROM mount directory, enter the following as the INPUT device:

   ```
   /cdrom/db2
   ```

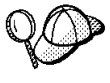
5. Press F4 to display a list of software to install. For a detailed description of the software packages, refer to Appendix B, “Contents of the DB2 Products” on page 309 for more information.

6. List the software to be installed using the **SOFTWARE to install** option.

7. Select the messages options for the locales you want to install.

Step 2. Create or Assign a Group and User ID

If you want to use an existing user or group ID, you do not need to create new ones at this time. Instead, you can proceed to the next step.

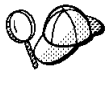


If you are using Network Information Services (NIS), NIS+, or any other tools to manage system configuration files, you must perform this step on the master server. After you have completed the following, ensure that the changes have taken effect on all DB2 servers.

You must perform this step using the *root* user ID.

1. Create a system administration (SYSADM) group that will be the primary group of the user ID for the instance owner. Any user of the instance owner that belongs to the SYSADM group will have system administrator authority for a given instance.

   Next, create a user ID that will be the instance owner. This user ID will be the name of the instance. Make this user's primary group the SYSADM group you created. In our examples, the instance user ID is `db2inst1` and the SYSADM group is `dbadmin1`.



Dedicate the instance owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

2. Use the **mkgroup** command to create groups, and the **mkuser** command to create users. For example, to create a user called `db2inst1` with user ID *1004*, which will use `dbadmin1` with group ID *999* as its primary group and use `/home/db2inst1` as its home directory, type:

   ```
   mkgroup id=999 dbadmin1
   mkuser id=1004 pgrp=dbadmin1 groups=dbadmin1 home=/home/db2inst1 db2inst1
   passwd db2inst1
   ```

3. Create a group and user ID for fenced user defined functions and stored procedures.

   For example:

   ```
   mkgroup id=991 db2fadm1
   mkuser id=1001 pgrp=db2fadm1 groups=db2fadm1 home=/home/db2fenc1 db2fenc1
   passwd db2fenc1
   ```

   Fenced user defined functions (UDFs) and stored procedures will execute under this user and group. The group must be the primary group of the user. The user for fenced UDFs and stored procedures is specified as a parameter of the instance creation script. The group for fenced UDFs and stored procedures is implicitly set to the primary group of this specified user (`db2fenc1`).

For security reasons, we recommend that you do not use the instance name as the FencedID. However, if you are not planning to use fenced user defined functions or stored procedures, you can set the FencedID to the instance name instead of creating another user for the FencedID.

Notes:

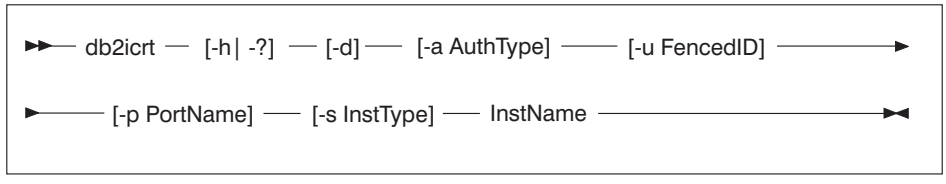
1. You have to repeat Steps 1 and 2 above to create a user name and group name for the Administration Server. You must use different user IDs for the Administration Server and a DB2 instance (for example, `db2as` and `db2inst`). For security reasons, you should not use the primary group of a DB2 instance as the primary group for the user ID for the Administration Server. It is recommended that you create a different group ID (for example, `db2asgrp`).
2. In addition to the rules imposed by the operating system for login names and groups, you must also adhere to the rules described in Appendix D, “Naming Rules” on page 317.

Step 3. Create a DB2 Instance

A DB2 instance is an environment where you store data and run applications. Use the **db2icrt** command to create an instance. You must have *root* authority to enter this command. For more information about database instances, see Chapter 16, “Working with Instances” on page 147.

The **db2icrt** command is in the DB2DIR/instance directory, where DB2DIR is /usr/lpp/db2_05_00.

The syntax of the **db2icrt** command is:



where:

-h | -? Display a help menu for this command.

-d Sets the debug mode that you can use for problem determination.

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are (SERVER), (CLIENT), and (DCS). If the *-a* parameter is not specified, the authentication type defaults to (SERVER), if a DB2 server is installed. Otherwise, the *AuthType* is set to (CLIENT).

Notes:

1. The authentication type of the instance applies to all databases owned by the instance.
2. While authentication type (DCE) is an optional parameter, it is not valid to choose (DCE) for this command.

-u *FencedID* Is the user under which the fenced UDFs and stored procedures will execute. This is not required if you install the DB2 Client Application Enabler or the DB2 Software Developer's Kit. For other products, this is a required parameter.

Note: *FencedID* may not be *root* or *bin*.

-p *PortName* Is an optional parameter that specifies the TCP/IP service name or port number to be used. This value will then be set in the instance's database configuration file.

-s *InstType* Is an optional parameter that allows different types of instances to be created. Valid instance types are *ee*, *eee* and *client*.

instance_name Is the login name of the instance owner.

If a server product is installed, the syntax is:

```
db2icrt -u FencedID instance_name
```

Examples:

- To create an instance for a DB2 server, you can use the following command:

  ```
  db2icrt -u db2fenc1 db2inst1
  ```

- If you installed the DB2 Connect Enterprise Edition only, you can use the instance name as the Fenced ID also:

  ```
  db2icrt -u db2inst1 db2inst1
  ```

- To create an instance for a DB2 client, you can use the following command:

  ```
  db2icrt db2inst1
  ```

When an instance is created, its name is also added to the list of instances on the system.

Step 4. Create the Administration Server

The Administration Server provides services to support client tools that:

- Automate the configuration of connections to DB2 databases.
- Administer DB2 from your server system or a remote client using the Command Center.

To create an Administration Server, see the procedures described in "Creating the Administration Server" on page 157.

Step 5. Install the License Key

The following steps describe how to upgrade the license:

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

1. Log in as a user with root authority.
2. The DB2 product license key can be found in the `nodelock` file:

   ```
   /usr/lib/netls/conf/nodelock
   ```

3. Update your DB2 product license with the following command:

   ```
   /usr/lpp/db2_05_00/cfg/db2licinst license_filename
   ```

   where `license_filename` is the full pathname and filename for the license file that corresponds to the product you have purchased. The name of the license file for these products is:

   db2entr.lic DB2 Universal Database Enterprise Edition

For example, if the CD-ROM is mounted in the /cdrom directory and the name of the license file is db2entr.lic, the command should be as follows:

```
/usr/lpp/db2_05_00/cfg/db2licinst /cdrom/db2/license/db2entr.lic
```

Step 6. Create Links for DB2 Files (Optional)

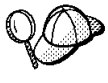
You can create links for the DB2 files to the /usr/lib directory, and for the include files to the /usr/include directory for a particular version and release level of the product.

You may want to create these links if you are developing or running applications and want to avoid having to specify the full path to the product libraries and include files.

Log in as a user with root authority and invoke the **db2ln** command as follows:

```
/usr/lpp/db2_05_00/cfg/db2ln
```

If there are existing links to the /usr/lib and /usr/include directories from previous versions of DB2, they will automatically be removed by entering the **db2ln** command to create links for this version of DB2. If you want to re-establish the links to the libraries or previous versions, then you must execute the **db2rmln** command from Version 5 before you execute the **db2ln** command from the previous versions of DB2.



Links can be established for only one version of DB2 on a given system.

Chapter 22. Other Methods to Install DB2 for HP-UX

This section describes the other methods you can use to install DB2 Universal Database products. Some of the advanced methods to install DB2 are detailed in this section. The DB2 Installer program can perform all of the steps described in this section.

We recommend that you use the DB2 Installer program when installing DB2, but if you prefer not to use the DB2 Installer program, you can use any one of the methods described in this chapter. See Chapter 3, “Installing and Configuring DB2” on page 39 for information on how to use the DB2 Installer program to install the DB2 product. See Chapter 7, “Installing DB2 Clients” on page 79 for information on how to install the DB2 clients.

Installation Steps

To install DB2 for HP-UX products, perform the following steps:

1. Install the product from the CD-ROM to the target machine. As part of this installation step, the following installation activities are optional:
   a. Install the DB2 product messages.
   b. Install the translated messages.
2. Update the kernel configuration parameters and reboot the system.
3. Create or assign groups and user IDs.
4. Create a DB2 instance.
5. Create an Administration Server.
6. Install the license key.
7. Create links for DB2 files (optional).

Step 1. Install DB2

Use the **swinstall** program to install DB2 for HP-UX. To install, perform the following steps:

1. Log in as a user with root authority.
2. Insert the DB2 CD-ROM into the CD-ROM drive and mount it. For example:

   ```
   mkdir /cdrom
   /usr/sbin/mount /dev/dsk/c0t2d0 /cdrom
   ```

   where `/cdrom` is the CD-ROM mount directory.
3. Run the **swinstall** program using the following command:

   ```
   swinstall -x autoselect_dependencies=true
   ```

   This opens the Software Selection window and the Specify Source window.
4. If necessary, change the **Source Host Name** in the Specify Source window.
5. Enter the following as the value for the **Source Depot Path** field:
   - For HP-UX Version 10:

     ```
     /cdrom/db2/hpux10
     ```

   - For HP-UX Version 11:

     ```
     /cdrom/db2/hpux11
     ```

   where `/cdrom` is the CD-ROM mount directory.
6. To return to the Software Selection window, select the **OK** button.
7. The Software Selection window contains a list of available software to install. Highlight one or more of the following products to install:

DB2V5CAE	DB2 Universal Database Client Application Enabler for HP-UX
DB2V5CONN	DB2 Connect Enterprise Edition for HP-UX
DB2V5ENTP	DB2 Universal Database Enterprise Edition for HP-UX
DB2V5HTML	DB2 Universal Database Product Library in HTML (all locales)
DB2V5MSG	DB2 Universal Database Product Messages (all locales)

Notes:

a. This CD-ROM contains several DB2 products. Select the products you are licensed to install. Your **Proof of Entitlement** and **License Information** booklet identify the products for which you are licensed.

b. Do not select products DB2V5MSG and DB2V5HTML. These products are collections of DB2 product messages and documentations in several languages. The procedure to install DB2 product messages and documentation (HTML) is described later in this section.

8 Select **Mark for Install** from the **Actions** menu to choose the product to be installed.

Note: Ignore the error message similar to the following:

```
The software "DB2V5CAE r=5.0.0,a=HP-UX_B.10.10_700/800,v=IBM" was
successfully marked, but it depends on the following software items
which could not be found in the source. However, these items may
already be in the target. This will be checked during the Analysis
Phase: OS-Core.CMDS-AUX,r>=B.10.10,a=HP-UX_?.10.??_7800,v=HP
OS-Core.CMDS-MIN,r>=B.10.10,a=HP-UX_?.10.??_7800,v=HP
SOE.SOE,r>=B.10.10,a=HP-UX_?.10.??_7800,v=HP
```

9 Select **OK** when the following message appears:

```
In addition to the software you just marked, other software was
automatically marked to resolve dependencies.  This message will
not appear again.
```

10 Select **Install (analysis)** from the **Actions** menu to begin product installation and to open the Install Analysis window.

11 Select **OK** in the Install Analysis window when the **Status** field displays a `Ready` message.

12 Select the **Yes** button in the Confirmation window windows to confirm that you want to install the HP-UX software products.

View the Install window to read processing data while the software is being installed, until the **Status** field indicates `Ready` and the Note window opens. The **swinstall** program loads the fileset, and runs the control scripts for the fileset.

13 Select **Exit** from the **File** menu to exit out of **swinstall**.

The HP-UX operating system provides detailed help for **swinstall**. For help, type:

```
man swinstall
```

Install the DB2 Product Library (Optional)

To view DB2 product documents online, HTML versions of these documents are provided for installation. These documents are translated into several languages. However, not every manual in the DB2 product library is translated into multiple languages. The English version of every manual is available in HTML format. When you select a non-English HTML fileset without selecting the English version, the English HTML fileset is also installed.

The DB2V5HTML product includes filesets for all the DB2 product documentation. A separate fileset exists for each language. For example, the fileset name for the DB2 product library in English is `DB2V5HTML.en_US`. For a complete list of filesets in the DB2V:1TML product, see Appendix B, "Contents of the DB2 Products" on page 309.

To install the DB2 Product Library (HTML) filesets, do the following:

1. Log in as a user with root authority.
2. Insert and mount the DB2 product CD-ROM as described in "Step 1. Install DB2" on page 214.
3. Run the **swinstall** command as follows:
   - For HP-UX Version 10:

     ```
     swinstall -s /cdrom/db2/hpux10
     ```

   - For HP-UX Version 11:

     ```
     swinstall -s /cdrom/db2/hpux11
     ```

4. From the list of products displayed, highlight the **DB2V:ITML** product and press Enter.

   This will display a list of all the filesets in the **DB2V:ITML** product.
5. Highlight one or more filesets that you want to install and select **Mark for Install** from the Actions menu.

   For example, to install the DB2 product Library in English, highlight the following fileset:

   ```
   en_US     DB2 Product Library (HTML) - English
   ```

6. Follow the remaining installation procedure, starting at Step 9 on page 215 in "Step 1. Install DB2" on page 214.

   This will install the **compressed-tar** files for the DB2 documents in the /opt/IBMdb2/V5.0/doc/<lang>/html directory, where `<lang>` is the language/locale identifier.
7. Run the following command to uncompress and un-tar the HTML files:

   ```
   /opt/IBMdb2/V5.0/doc/db2insthtml lang
   ```

where *lang* is the language/locale identified. For example, lang=en_US for English. For valid language/locale identifiers, see Table 35 on page 311.

8. From your Web browser, open the following URL to view DB2 online manuals:

   ```
   file:/opt/IBMdb2/V5.0/doc/en_US/html/index.htm
   ```

To recover some disk space, you may want to remove the compressed tar files. It is recommended that you do not just remove the compressed tar HTML files; instead, you should remove the fileset for the DB2 Product Library.

After the fileset is removed, only compressed tar files will be removed. You will still be able to view the online DB2 books.

Install DB2 Product Messages (Optional)

While DB2 Product Message documents are translated into several languages, the English language DB2 Product Messages are always installed. To install the DB2 Product Messages in other locales, you have to install one or more filesets in the DB2V5MSG product.

The DB2V5MSG product includes a separate fileset for every locale in which DB2 Product Messages are available. There is no fileset for the English (en_US.iso88591 or en_US.roman8) locale since these messages are installed with the base product. For example, to install the DB2 Product messages in the `fr_FR.iso88591` locale, you need to install the fileset `DB2V5MSG.fr_FR`. For a complete list of filesets in the DB2V5MSG product, see Appendix B, “Contents of the DB2 Products” on page 309.

To install DB2 Product Message filesets, do the following:

1. Log in as a user with root authority.
2. Insert and mount the DB2 product CD-ROM as described in “Installation Steps” on page 213.
3. Run the **swinstall** command as follows:
   - For HP-UX Version 10:

     ```
     swinstall -s /cdrom/db2/hpux10
     ```

   - For HP-UX Version 11:

     ```
     swinstall -s /cdrom/db2/hpux11
     ```

   where `/cdrom` is the CD-ROM mount directory.
4. From the list of products displayed, highlight the **DB2V5MSG** product and press Enter.

   This will display a list of all the filesets in the **DB2V5MSG** product.
5. Highlight one or more filesets that you want to install and select **Mark for Install** from the Actions menu.

6 Follow the remaining installation procedure, starting at Step 9 on page 215 in "Step 1. Install DB2" on page 214.

Step 2. Update the Kernel Configuration Parameters

To run DB2 for HP-UX, you may have to update some kernel configuration parameters; the following values are recommended:

Table 24. HP-UX Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Physical Memory		
	64MB - 128MB	**128MB - 256MB**	**256MB+**
maxuprc	256	384	512
maxfiles	256	256	256
nproc	512	768	1024
nflocks	2048	4096	8192
ninode	512	1024	2048
nfile	(4 * ninode)	(4 * ninode)	(4 * ninode)
msgseg	8192	16384	32768
msgmnb	65535 (1)	65535 (1)	65535 (1)
msgmax	65535 (1)	65535 (1)	65535 (1)
msgtql	256	512	1024
msgmap	130	258	258
msgmni	128	256	256
msgssz	16	16	16
semmni	128	256	512
semmap	130	258	514
semmns	256	512	1024
semmnu	256	512	1024
shmmax	67108864	134217728 (2)	268435456 (2)
shmseg	16	16	16
shmmni	300	300	300

Notes:

1. Parameters *msgmnb* and *msgmax* must be set to 65535.
2. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in the preceding table.
3. Parameter *shmmax* should be set to 134217728 or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set *shmmax* to 184968806 (176*1024*1024).

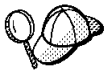
Step 3. Create or Assign Groups and User IDs

If you want to use an existing user or group ID, you do not need to create new ones at this time. Instead, you can proceed to the next step.

You must be a user with root authority to perform the following:

1 Create a system administration (SYSADM) group that will be the primary group of the user ID for the instance owner. Any user of the instance owner that belongs to the SYSADM group will have system administrator authority for a given instance.

Next, create a user ID that will be the instance owner. This user ID will be the name of the instance. Make this user's primary group the SYSADM group you created. In our examples, the instance user ID is `db2inst1` and the SYSADM group is `dbadmin1`.



Dedicate the instance owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

2 Use **SAM** or **groupadd** to create groups, and **SAM** or **useradd** to create users. For example:

```
groupadd dbadmin1
useradd -g dbadmin1 -d /home/instl -m db2inst1
passwd db2inst1
```

3 Create a group and user for fenced user defined functions and stored procedures.

For example:

```
groupadd db2fadm1
useradd -g db2fadm1 -d /home/db2fenc1 -m db2fenc1
passwd db2fenc1
```

Fenced user defined functions (UDFs) and stored procedures will execute under this user and group. The group must be the primary group of the user. The user for fenced UDFs and stored procedures is specified as a parameter of the instance creation script. The group for fenced UDFs and stored procedures is implicitly set to the primary group of this specified user (`db2fenc1`).

For security reasons, we recommend that you do not use the instance name as the FencedID. However, if you are not planning to use fenced user defined functions or stored procedures, you can set the FencedID to the instance name instead of creating another user for the FencedID.

Notes:

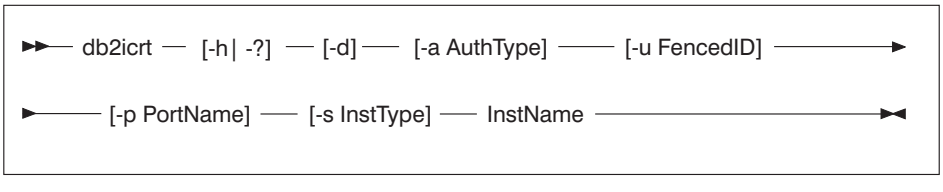
1. You have to repeat Steps 1 and 2 above to create a user name and group name for the Administration Server. You must use different user IDs for the Administration Server and a DB2 instance (for example, `db2as` and `db2inst`). For security reasons, you should not use the primary group of a DB2 instance as the primary group for the user ID for the Administration Server. It is recommended that you create a different group ID (for example, `db2asgrp`).
2. In addition to the rules imposed by the operating system for login names and groups, you must also adhere to the rules described in Appendix D, "Naming Rules" on page 317.

Step 4. Create a DB2 Instance

A DB2 instance is an environment where you store data and run applications. Use the **db2icrt** command to create an instance. You must have *root* authority to enter this command. For more information about database instances, see Chapter 16, "Working with Instances" on page 147.

The **db2icrt** command is located in the `DB2DIR/instance` directory, where `DB2DIR` is /opt/IBMdb2/V5.0.

The syntax of the **db2icrt** command is:



where:

-h | -? Display a help menu for this command.

-d Sets the debug mode that you can use for problem determination.

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are (SERVER), (CLIENT), and (DCS). If the *-a* parameter is not specified, the authentication type defaults to (SERVER), if a DB2 server is installed. Otherwise, the *AuthType* is set to (CLIENT).

Notes:

1. The authentication type of the instance applies to all databases owned by the instance.
2. While authentication type (DCE) is an optional parameter, it is not valid to choose (DCE) for this command.

-u *FencedID* Is the user under which the fenced UDFs and stored procedures will execute. This is not required if you install the DB2 Client Application Enabler or the DB2 Software Developer's Kit. For other products, this is a required parameter.

Note: *FencedID* may not be *root* or *bin*.

-p *PortName* Is an optional parameter that specifies the TCP/IP service name or port number to be used. This value will then be set in the instance's database configuration file.

-s *InstType* Is an optional parameter that allows different types of instances to be created. Valid instance types are *ee*, *eee* and *client*.

instance_name Is the login name of the instance owner.

If a server product is installed, the syntax is:

```
db2icrt -u FencedID instance_name
```

Examples:

- To create an instance for a DB2 server, you can use the following command:

  ```
  db2icrt -u db2fenc1 db2inst1
  ```

- If you installed the DB2 Connect Enterprise Edition only, you can use the instance name as the Fenced ID also:

  ```
  db2icrt -u db2inst1 db2inst1
  ```

- To create an instance for a DB2 client, you can use the following command:

  ```
  db2icrt db2inst1
  ```

When an instance is created, its name is also added to the list of instances on the system.

Step 5. Create an Administration Server

The Administration Server provides services to support client tools that:

- Automate the configuration of connections to DB2 databases.
- Administer DB2 from your server system or a remote client using the Command Center.

To create an Administration Server, see the procedures described in “Creating the Administration Server” on page 157.

Step 6. Install the License Key

The following steps describe how to upgrade the license:

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

1. Log in as a user with root authority.
2. The DB2 product license key can be found in the `nodelock` file:

   ```
   /usr/netls/nodelock
   ```

3. Update your DB2 product license with the following command:

   ```
   /opt/IBMdb2/V5.0/cfg/db2licinst license_filename
   ```

   where `license_filename` is the full pathname and filename for the license file that corresponds to the product you have purchased. The name of the license file for these products is:

   db2entr.lic DB2 Universal Database Enterprise Edition

   For example, if the CD-ROM is mounted in the `/cdrom` directory and the name of the license file is `db2entr.lic`, the command should be as follows:

   ```
   /opt/IBMdb2/V5.0/cfg/db2licinst /cdrom/db2/license/db2entr.lic
   ```

Step 7. Create Links for DB2 Files (Optional)

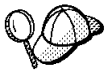
You can create links for the DB2 files to the /usr/lib directory, and for the include files to the /usr/include directory for a particular version and release level of the product.

You may want to create these links if you are developing or running applications and want to avoid having to specify the full path to the product libraries and include files.

Log in as root and invoke the **db2ln** command as follows:

```
/opt/IBMdb2/V5.0/db2ln
```

If there are existing links to the /usr/lib and /usr/include directories from previous versions of DB2, they will automatically be removed by entering the **db2ln** command to create links for this version of DB2. If you want to re-establish the links to the libraries or previous versions, then you must execute the **db2rmln** command from Version 5 before you execute the **db2ln** command from the previous versions of DB2.



Links can be established for only one version of DB2 on a given system.

Now that you have installed DB2 Enterprise Edition, see Chapter 26, “Setting up Communications on the Server Using the Command Line Processor” on page 251 for information on how to set up your server to accept in-bound client connections.

Chapter 23. Other Methods to Install DB2 for Solaris

This section describes the other methods you can use to install DB2 Universal Database products. Some of the advanced methods to install DB2 are detailed in this section. The DB2 Installer program can perform all of the steps described in this section.

We recommend that you use the DB2 Installer program when installing DB2, but if you prefer not to use the DB2 Installer program, you can use any one of the methods described in this chapter. See Chapter 3, “Installing and Configuring DB2” on page 39 for information on how to use the DB2 Installer program to install the DB2 product. See Chapter 7, “Installing DB2 Clients” on page 79 for information on how to install the DB2 clients.

Installation Steps

To install DB2 Extended Enterprise Edition for Solaris, perform the following steps:

1. Install DB2. As part of this installation step, the following installation activities are optional:
   a. Install the DB2 Product Library (documents).
   b. Install the DB2 Product Messages.
2. Update the kernel configuration parameters.
3. Create or assign groups and user IDs.
4. Create a DB2 instance.
5. Create an Administration Server.
6. Install the license key.
7. Create links for DB2 files (optional).

Step 1. Install DB2

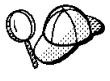
Use the **db2_install** command, which is available on the DB2 CD-ROM, to install DB2. If your system does not have a local CD-ROM drive, you can mount the CD-ROM on a different system and NFS mount on the current system.

To install DB2 from the DB2 CD-ROM, perform the following steps:

1. Log in as a user with root authority.
2. Insert the DB2 CD-ROM into the CD-ROM drive.
3. If the Volume Manager is not running on your system, enter the following commands to mount the CD-ROM:

   ```
   mkdir -p /cdrom/unnamed_cdrom
   mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom/unnamed_cdrom
   ```

   where */cdrom/unnamed_cdrom* is the CD-ROM mount directory.



If you are mounting the CD-ROM drive from a remote system using NFS, the CD-ROM file system on the remote machine must be exported with *root* access. You must also mount that file system with *root* access on the local machine.

   If the Volume Manager (vold) is running on your system, the CD-ROM is automatically mounted as:

   ```
   /cdrom/unnamed_cdrom
   ```

4. Run the **db2_install** command as follows:

   ```
   /cdrom/unnamed_cdrom/db2_install
   ```

   The **db2_install** command prompts for one or more of the following to be installed, and for the base directory where the product files are to be installed. The products are listed by keyword and product description.

   DB2.ENTP DB2 Universal Database Enterprise Edition for Solaris

   DB2.CONN DB2 Connect Enterprise Edition for Solaris

   DB2.CAE DB2 Universal Database Client Application Enabler

   NetQ IBM NetQuestion

   The command displays the following prompt: `Specify one or more of the keywords separated by spaces.`

5. Type the keyword of the products to be installed when prompted.
6. Type the name of the base directory when prompted. The default base directory is `/opt`.

If the default base directory is used, all files will be installed in the `/opt/IBMdb2/V5.0` directory.

7. Enter `Yes` to start the DB2 product installation.

Installing the DB2 Product Library (Optional)

To view DB2 product documents online, HTML versions of these documents are provided for installation. These documents are translated into several languages. However, not every manual in the DB2 product library is translated into multiple languages. The English version of every manual is available in HTML format. When you select a non-English HTML fileset without selecting the English version, the English HTML fileset is also installed.

A separate package exists for every language in which DB2 documents are translated. Packages for DB2 documentation are given names such as `db2ht%L50`, where `%L` represents two letters for the language and locale identifier.

For example, the package name for the DB2 Product Library in English is `db2hten50`. For a complete list of packages for the DB2 Product Library, see Appendix B, "Contents of the DB2 Products" on page 309.

To install the DB2 Product Library (HTML) package, do the following:

1. Log in as a user with root authority and mount the DB2 product CD-ROM as described in "Step 1. Install DB2" on page 226.

2. Run the **pkgadd** command as follows:

   ```
   pkgadd -d cdrom-dir/db2 pkgname
   ```

   where *cdrom-dir* is the CD-ROM mount directory and *pkgname* is the package name. For example, to install DB2 documentation in English, you can enter the following command:

   ```
   pkgadd -d /cdrom/unnamed_cdrom/db2 db2hten50
   ```

   This command installs the compressed-tar files for the DB2 documents in the `/opt/IBMdb2/V5.0`/doc/*lang*/html directory, where *lang* is the language and locale identifier.

3. Run the following command to uncompress and untar the HTML files:

   ```
   /opt/IBMdb2/V5.0/doc/db2insthtml lang
   ```

   where *lang* is the language and locale identifier.

   For example, *lang*=en_US for English. For valid language/locale identifiers, see Table 35 on page 311.

4. From your Web browser, open the following URL to view the DB2 manuals:

   ```
   file:/opt/IBMdb2/V5.0/doc/en_US/html/index.htm
   ```

To recover some disk space, you may want to remove the compressed tar files. It is recommended that you do not just remove the compressed tar HTML files; instead, you should remove the fileset for the DB2 Product Library.

After the fileset is removed, only compressed tar files will be removed. You will still be able to view the online DB2 books.

Installing the DB2 Product Messages (Optional)

A separate package exists for every locale in which DB2 Product Messages are translated. Packages for DB2 Product Messages are given names such as `db2ms%L50`, where `%L` represents two letters for the language name. For example, the package name for the DB2 Product Messages in the `de` (German) locale is `db2msde50`. For a complete list of DB2 Product Messages, see Appendix B, "Contents of the DB2 Products" on page 309.

To install DB2 Product Message packages, do the following:

1. Log in as a user with root authority and mount the DB2 CD-ROM as described in "Step 1. Install DB2" on page 226.
2. Run the **pkgadd** command as follows:

   ```
   pkgadd -d cdrom-dir/db2 pkgname
   ```

   where *cdrom-dir* is the CD-ROM mount directory and *pkgname* is the package name.

   For example, to install the DB2 Product Messages in the `de` (German) locale, enter the following command:

   ```
   pkgadd -d /cdrom/unnamed_cdrom/db2 db2msde50
   ```

Step 2. Update the Kernel Configuration Parameters

To run DB2 for Solaris, you may have to update some kernel configuration parameters; the following minimum values are recommended:

Table 25. Solaris Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Physical Memory			
	64MB - 128MB	128MB - 256MB	256MB - 512MB	512MB+
msgsys:msginfo_msgmax	65535 (1)	65535 (1)	65535 (1)	65535 (1)
msgsys:msginfo_msgmnb	65535 (1)	65535 (1)	65535 (1)	65535 (1)
msgsys:msginfo_msgmap	130	258	258	258
msgsys:msginfo_msgmni	128	256	256	256
msgsys:msginfo_msgssz	16	16	16	16
msgsys:msginfo_msgtql	256	512	1024	1024
msgsys:msginfo_msgseg	8192	16384	32768	32768
shmsys:shminfo_shmmax	67108864	134217728 (2)	268435456 (2)	536870912 (2)
shmsys:shminfo_shmseg	16	16	16	16
shmsys:shminfo_shmmni	300	300	300	300
semsys:seminfo_semmni	128	256	512	1024
semsys:seminfo_semmap	130	258	514	1026
semsys:seminfo_semmns	256	512	1024	2048
semsys:seminfo_semmnu	256	512	1024	2048

Notes:

1. The *msgsys:msginfo_msgmnb* and *msgsys:msginfo_msgmax* parameters must be set to 65535.
2. The *shmsys:shminfo_shmmax* parameters should be set to the suggested value in the above table, or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set the *shmsys:shminfo_shmmax* parameter to 184968806 (176*1024*1024).

To set a kernel parameter, add a line at the end of the `/etc/system` file as follows:

```
set parameter_name = value
```

For example, to set the value of the *msgsys:msginfo_msgmax* parameter, add the following line to the end of the `/etc/system` file:

```
set msgsys:msginfo_msgmax = 65535
```

Sample files for updating the kernel configuration parameters are provided in the `/opt/IBMdb2/V5.0/cfg` directory. The names for these files are as follows:

- `kernel.param.64MB` for systems with 64MB–124MB of physical memory
- `kernel.param.128MB` for systems with 128MB–256MB of physical memory
- `kernel.param.256MB` for systems with more than 256MB–512MB of physical memory
- `kernel.param.512MB` for systems with more than 512MB of physical memory

Depending upon the amount of physical memory in your system, append the appropriate kernel configuration parameter file to the `/etc/system` file. If necessary, change the value of the *shmsys:shminfo_shmmax* parameter as described in Note 2 above.

After updating the `/etc/system` file, reboot the system.

Step 3. Create or Assign Groups and User IDs

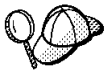
If you want to use an existing user or group ID, you do not need to create new ones at this time. Instead, you can proceed to the next step.

If you are using Network Information Services (NIS), NIS+, or any other tools to manage system configuration files, you must perform this step on the master server. After you have completed the following, ensure that the changes have taken effect on all DB2 servers.

You must perform the following as a user with root authority.

1. Create a system administration (SYSADM) group that will be the primary group of the user ID for the instance owner. Any user of the instance owner that belongs to the SYSADM group will have system administrator authority for a given instance.

   Next, create a user ID that will be the instance owner. This user ID will be the name of the instance. Make this user's primary group the SYSADM group you created. In our examples, the instance user ID is `db2inst1` and the SYSADM group is `dbadmin1`.



Dedicate the instance owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

2. Use **admintool** or **groupadd** to create groups, and **admintool** or **useradd** to create users. For example:

```
groupadd -g 999 dbadmin1
useradd -g dbadmin1 -u 1004 -d /export/home/db2inst1 -m db2inst1
passwd db2inst1
```

3. Similarly, create a group and user for fenced user defined functions and stored procedures.

   For example:

```
groupadd -g 991 db2fadm1
useradd -g db2fadm1 -u 1001 -d /export/home/db2fenc1 -m db2fenc1
passwd db2fenc1
```

Fenced user defined functions (UDFs) and stored procedures will execute under this user and group. The group must be the primary group of the user. The user for fenced UDFs and stored procedures is specified as a parameter of the instance creation script. The group for fenced UDFs and stored procedures is implicitly set to the primary group of this specified user (`db2fenc1`).

For security reasons, we recommend that you do not use the instance name as the FencedID. However, if you are not planning to use fenced user defined functions or stored procedures, you can set the FencedID to the instance name instead of creating another user for the FencedID.

Notes:

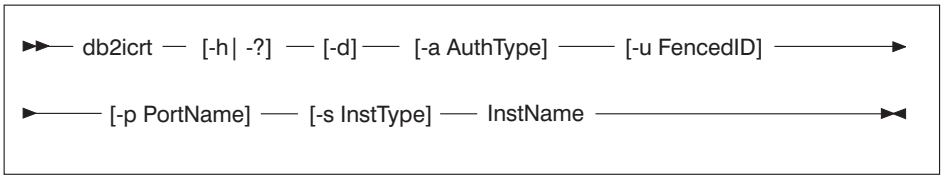
1. You have to repeat Steps 1 and 2 above to create a user name and group name for the Administration Server. You must use different user IDs for the Administration Server and a DB2 instance (for example, `db2as` and `db2inst`). For security reasons, you should not use the primary group of a DB2 instance as the primary group for the user ID for the Administration Server. It is recommended that you create a different group ID (for example, `db2asgrp`).
2. In addition to the rules imposed by the operating system for login names and groups, you must also adhere to the rules described in Appendix D, "Naming Rules" on page 317.

Step 4. Create a DB2 Instance

A DB2 instance is an environment where you store data and run applications. Use the **db2icrt** command to create an instance. You must have *root* authority to enter this command. For more information about database instances, see Chapter 16, "Working with Instances" on page 147.

The **db2icrt** command is located in the `DB2DIR/instance` directory, where `DB2DIR` is `/opt/IBMdb2/V5.0`.

The syntax of the **db2icrt** command is:



where:

-h | -? Display a help menu for this command.

-d Sets the debug mode that you can use for problem determination.

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are (SERVER), (CLIENT), and (DCS). If the *-a* parameter is not specified, the authentication type defaults to (SERVER), if a DB2 server is installed. Otherwise, the *AuthType* is set to (CLIENT).

Notes:

1. The authentication type of the instance applies to all databases owned by the instance.
2. While authentication type (DCE) is an optional parameter, it is not valid to choose (DCE) for this command.

-u *FencedID* Is the user under which the fenced UDFs and stored procedures will execute. This is not required if you install the DB2 Client Application Enabler or the DB2 Software Developer's Kit. For other products, this is a required parameter.

Note: *FencedID* may not be *root* or *bin*.

-p *PortName* Is an optional parameter that specifies the TCP/IP service name or port number to be used. This value will then be set in the instance's database configuration file.

-s *InstType* Is an optional parameter that allows different types of instances to be created. Valid instance types are *ee*, *eee* and *client*.

instance_name Is the login name of the instance owner.

If a server product is installed, the syntax is:

```
db2icrt -u FencedID instance_name
```

Examples:

- To create an instance for a DB2 server, you can use the following command:

  ```
  db2icrt -u db2fenc1 db2inst1
  ```

- If you installed the DB2 Connect Enterprise Edition only, you can use the instance name as the Fenced ID also:

  ```
  db2icrt -u db2inst1 db2inst1
  ```

- To create an instance for a DB2 client, you can use the following command:

  ```
  db2icrt db2inst1
  ```

When an instance is created, its name is also added to the list of instances on the system.

Step 5. Create the Administration Server

The Administration Server provides services to support client tools that:

- Automate the configuration of connections to DB2 databases.
- Administer DB2 from your server system or a remote client using the Command Center.

To create an Administration Server, see the procedures described in “Creating the Administration Server” on page 157.

Step 6. Install the License Key

The following steps describe how to upgrade the license:

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

1. Log in as a user with root authority.
2. The DB2 product license key can be found in the `nodelock` file:

   ```
   /var/netls/nodelock
   ```

3. Update your DB2 product license with the following command:

   ```
   /opt/IBMdb2/V5.0/cfg/db2licinst license_filename
   ```

   where `license_filename` is the full pathname and filename for the license file that corresponds to the product you have purchased. The name of the license file for these products is:

   db2entr.lic DB2 Universal Database Enterprise Edition

   For example, if the CD-ROM is mounted in the `/cdrom` directory and the name of the license file is `db2entr.lic`, the command should be as follows:

   ```
   /opt/IBMdb2/V5.0/cfg/db2licinst /cdrom_unnamed/db2/license/db2entr.lic
   ```

Step 7. Create Links for DB2 Files (Optional)

You can create links for the DB2 files to the /usr/lib directory, and for the include files to the /usr/include directory for a particular version and release level of the product.

You may want to create these links if you are developing or running applications and want to avoid having to specify the full path to the product libraries and include files.

Log in as a user with root authority and invoke the **db2ln** command as follows:

```
/opt/IBMdb2/V5.0/cfg/db2ln
```

If there are existing links to the /usr/lib and /usr/include directories from previous versions of DB2, they will automatically be removed by entering the **db2ln** command to create links for this version of DB2. If you want to re-establish the links to the libraries or previous versions, then you must execute the **db2rmln** command from Version 5 before you execute the **db2ln** command from the previous versions of DB2.

Links can be established for only one version of DB2 on a given system.

Chapter 24. Other Methods to Install SCO UnixWare 7

This section describes the other methods you can use to install DB2 Universal Database products. Some of the advanced methods to install DB2 are detailed in this section. The DB2 Installer program can perform all of the steps described in this section.

We recommend that you use the DB2 Installer program when installing DB2, but if you prefer not to use the DB2 Installer program, you can use any one of the methods described in this chapter. See Chapter 3, “Installing and Configuring DB2” on page 39 for information on how to use the DB2 Installer program to install the DB2 product. See Chapter 7, “Installing DB2 Clients” on page 79 for information on how to install the DB2 clients.

Installation Steps

To install SCO UnixWare 7 products, perform the following steps:

1. Install DB2. As part of this installation step, the following installation activities are optional:
   a. Install DB2 product messages.
   b. Install translated messages.
2. Update the kernel configuration parameters and reboot the system.
3. Create or assign groups and user IDs.
4. Create a DB2 instance.
5. Create an Administration Server.
6. Install the license key.
7. Create links for DB2 files (optional).

Step 1. Install DB2

Use the **db2_install** command, which is available on the DB2 CD-ROM, to install DB2. If your system does not have a local CD-ROM drive, you can mount the CD-ROM on a different system and NFS mount on the current system.

To install DB2 from the DB2 CD-ROM, perform the following steps:

1. Log in as a user with root authority.
2. Insert the DB2 CD-ROM into the CD-ROM drive and mount it as in the following example:

   ```
   mount -F cdfs -o ro /dev/cdrom/c0b0t510 /cdrom
   ```

   where `c0b0t510` is the device name found under the `/dev/cdrom` directory and varies with the particular hardware on the system.
3. Run the **db2_install** command as follows:

   ```
   /cdrom/db2_install
   ```

   The **db2_install** command prompts for one or more of the following to be installed, and for the base directory where the product files are to be installed. The products are listed by keyword and product description.

DB2.CAE	DB2 Universal Database Client Application Enabler
DB2.WKGP	DB2 Universal Database Enterprise Edition for SCO UnixWare 7
DB2.ENTP	DB2 Universal Database Enterprise Edition for SCO UnixWare 7
DB2.CONN	DB2 Connect Enterprise Edition for SCO UnixWare 7
DB2.SDK	DB2 Universal Database Software Developer's Kit
NetQ	IBM NetQuestion

   The command displays the following prompt: `Specify one or more of the keywords separated by spaces.`
4. Type the keyword of the products to be installed when prompted.
5. Type the name of the base directory when prompted. The default base directory is `/opt`.

   If the default base directory is used, all files will be installed in the `/opt/IBMdb2/V5.0` directory.
6. Enter `Yes` to start the DB2 product installation.

Installing the DB2 Product Library (Optional)

To view DB2 product documents online, HTML versions of these documents are provided for installation. These documents are translated into several languages. However, not every manual in the DB2 product library is translated into multiple languages. The English version of every manual is available in HTML format. When you select a non-English HTML fileset without selecting the English version, the English HTML fileset is also installed.

A separate package exists for every language in which DB2 documents are translated. Packages for DB2 documentation are given names such as `db2ht%L50`, where `%L` represents two letters for the language and locale identifier.

For example, the package name for the DB2 Product Library in English is `db2hten50`. For a complete list of packages for the DB2 Product Library, see Appendix B, "Contents of the DB2 Products" on page 309.

To install the DB2 Product Library (HTML) package, do the following:

1. Log in as a user with root authority and mount the DB2 product CD-ROM as described in "Step 1. Install DB2" on page 226.
2. Run the **pkgadd** command as follows:

   ```
   pkgadd -d cdrom-dir/db2 pkgname
   ```

   where *cdrom-dir* is the CD-ROM mount directory and *pkgname* is the package name. For example, to install DB2 documentation in English, you can enter the following command:

   ```
   pkgadd -d /cdrom-dir/db2 db2hten50
   ```

   This command installs the compressed-tar files for the DB2 documents in the `/opt/IBMdb2/V5.0`/doc/*lang*/html directory, where *lang* is the language and locale identifier.
3. Run the following command to uncompress and untar the HTML files:

   ```
   /opt/IBMdb2/V5.0/doc/db2insthtml lang
   ```

   where *lang* is the language and locale identifier.

   For example, *lang*=en_US for English. For valid language/locale identifiers, see Table 35 on page 311.
4. From your Web browser, open the following URL to view the DB2 manuals:

   ```
   file:/opt/IBMdb2/V5.0/doc/en_US/html/index.htm
   ```

To recover some disk space, you may want to remove the compressed tar files. It is recommended that you do not just remove the compressed tar HTML files; instead, you should remove the fileset for the DB2 Product Library.

After the fileset is removed, only compressed tar files will be removed. You will still be able to view the online DB2 books.

Installing the DB2 Product Messages (Optional)

A separate package exists for every locale in which DB2 Product Messages are translated. Packages for DB2 Product Messages are given names such as `db2ms%L50`, where `%L` represents two letters for the language name. For example, the package name for the DB2 Product Messages in the `de` (German) locale is `db2msde50`. For a complete list of DB2 Product Messages, see Appendix B, "Contents of the DB2 Products" on page 309.

To install DB2 Product Message packages, do the following:

1. Log in as a user with root authority and mount the DB2 CD-ROM as described in "Step 1. Install DB2" on page 226.
2. Run the **pkgadd** command as follows:

   ```
   pkgadd -d cdrom-dir/db2 pkgname
   ```

   where *cdrom-dir* is the CD-ROM mount directory and *pkgname* is the package name.

   For example, to install the DB2 Product Messages in the `de` (German) locale, enter the following command:

   ```
   pkgadd -d /cdrom/db2 db2msde50
   ```

Step 2. Update the Kernel Configuration Parameters

To run DB2 for SCO UnixWare 7, you may have to update some kernel configuration parameters; the following minimum values are recommended:

Table 26. SCO UnixWare 7 Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Recommended Value
msgmax	65535 (1)
msgmnb	65535 (1)
msgssz	524288
msgmni	256
shmmax	268435456
shmmni	300
shmseg	15
semmi	1024

Notes:

1. Parameters *msgmax* and *msgmnb* should be set at least to 65535.
2. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in the preceding table.
3. Parameter *shmmax* should be set to the suggested value in the above table or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set *shmmax* to 184968806 (176*1024*1024).

To change a value, do the following:

1 Enter the **scoadmin** command to start the System Administration tool.

2 Double-click on the **System** folder.

3 Double-click on the **System Tuner** icon.

4 Click on the drop down box and select the **Inter-Process Communication (IPC) Parameters**.

5 Select the parameter to be changed and enter the new value.

6 Click on **OK** when you have finished changing all the parameters.

7 Click on the **Yes** push button to rebuild the kernel.

8 Reboot the system so that the changes can take effect.

Step 3. Create or Assign Groups and User IDs

If you want to use an existing user or group ID, you do not need to create new ones at this time. Instead, you can proceed to the next step.

If you are using Network Information Services (NIS), NIS+, or any other tools to manage system configuration files, you must perform this step on the master server. After you have completed the following, ensure that the changes have taken effect on all DB2 servers.

You must perform the following as a user with root authority.

1. Create a system administration (SYSADM) group that will be the primary group of the user ID for the instance owner. Any user of the instance owner that belongs to the SYSADM group will have system administrator authority for a given instance.

   Next, create a user ID that will be the instance owner. This user ID will be the name of the instance. Make this user's primary group the SYSADM group you created. In our examples, the instance user ID is `db2inst1` and the SYSADM group is `dbadmin1`.

Dedicate the instance owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

2. Use **admintool** or **groupadd** to create groups, and **admintool** or **useradd** to create users. For example:

   ```
   groupadd -g 999 dbadmin1
   useradd -g dbadmin1 -u 1004 -d /export/home/db2inst1 -m db2inst1
   passwd db2inst1
   ```

3. Similarly, create a group and user for fenced user defined functions and stored procedures.

   For example:

   ```
   groupadd -g 991 db2fadm1
   useradd -g db2fadm1 -u 1001 -d /export/home/db2fenc1 -m db2fenc1
   passwd db2fenc1
   ```

   Fenced user defined functions (UDFs) and stored procedures will execute under this user and group. The group must be the primary group of the user. The user for fenced UDFs and stored procedures is specified as a parameter of the instance creation script. The group for fenced UDFs and stored procedures is implicitly set to the primary group of this specified user (`db2fenc1`).

For security reasons, we recommend that you do not use the instance name as the FencedID. However, if you are not planning to use fenced user defined functions or stored procedures, you can set the FencedID to the instance name instead of creating another user for the FencedID.

Notes:

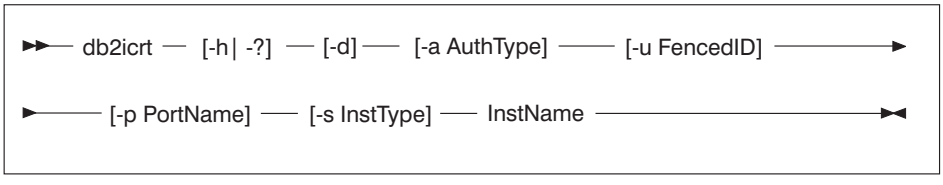
1. You have to repeat Steps 1 and 2 above to create a user name and group name for the Administration Server. You must use different user IDs for the Administration Server and a DB2 instance (for example, `db2as` and `db2inst`). For security reasons, you should not use the primary group of a DB2 instance as the primary group for the user ID for the Administration Server. It is recommended that you create a different group ID (for example, `db2asgrp`).
2. In addition to the rules imposed by the operating system for login names and groups, you must also adhere to the rules described in Appendix D, “Naming Rules” on page 317.

Step 4. Create a DB2 Instance

A DB2 instance is an environment where you store data and run applications. Use the **db2icrt** command to create an instance. You must have *root* authority to enter this command. For more information about database instances, see Chapter 16, “Working with Instances” on page 147.

The **db2icrt** command is located in the `DB2DIR/instance` directory, where `DB2DIR` is /opt/IBMdb2/V5.0.

The syntax of the **db2icrt** command is:



where:

-h | -? Display a help menu for this command.

-d Sets the debug mode that you can use for problem determination.

-a *AuthType* Is an optional parameter that specifies the authentication type for the instance. Valid authentication types are (SERVER), (CLIENT), and (DCS). If the *-a* parameter is not specified, the authentication type defaults to (SERVER), if a DB2 server is installed. Otherwise, the *AuthType* is set to (CLIENT).

Notes:

1. The authentication type of the instance applies to all databases owned by the instance.
2. While authentication type (DCE) is an optional parameter, it is not valid to choose (DCE) for this command.

-u *FencedID* Is the user under which the fenced UDFs and stored procedures will execute. This is not required if you install the DB2 Client Application Enabler or the DB2 Software Developer's Kit. For other products, this is a required parameter.

Note: *FencedID* may not be *root* or *bin*.

-p *PortName* Is an optional parameter that specifies the TCP/IP service name or port number to be used. This value will then be set in the instance's database configuration file.

-s *InstType* Is an optional parameter that allows different types of instances to be created. Valid instance types are *ee*, *eee* and *client*.

instance_name Is the login name of the instance owner.

If a server product is installed, the syntax is:

```
db2icrt -u FencedID instance_name
```

Examples:

- To create an instance for a DB2 server, you can use the following command:

  ```
  db2icrt -u db2fenc1 db2inst1
  ```

- If you installed the DB2 Connect Enterprise Edition only, you can use the instance name as the Fenced ID also:

  ```
  db2icrt -u db2inst1 db2inst1
  ```

- To create an instance for a DB2 client, you can use the following command:

  ```
  db2icrt db2inst1
  ```

When an instance is created, its name is also added to the list of instances on the system.

Step 5. Create an Administration Server

The Administration Server provides services to support client tools that:

- Automate the configuration of connections to DB2 databases.
- Administer DB2 from your server system or a remote client using the Command Center.

To create an Administration Server, see the procedures described in "Creating the Administration Server" on page 157.

Step 6. Install the License Key

The following steps describe how to upgrade the license:

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

1. Log in as a user with root authority.
2. The DB2 product license key can be found in the `nodelock` file:

   ```
   /var/netls/nodelock
   ```

3 Update your DB2 product license with the following command:

```
/opt/IBMdb2/V5.0/cfg/db2licinst license_filename
```

where `license_filename` is the full pathname and filename for the license file that corresponds to the product you have purchased. The name of the license file for these products is:

db2entr.lic DB2 Universal Database Enterprise Edition

For example, if the CD-ROM is mounted in the `/cdrom` directory and the name of the license file is `db2entr.lic`, the command should be as follows:

```
/opt/IBMdb2/V5.0/cfg/db2licinst /cdrom_unnamed/db2/license/db2entr.lic
```

Step 7. Create Links for DB2 Files (Optional)

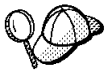
You can create links for the DB2 files to the `/usr/lib` directory, and for the include files to the `/usr/include` directory for a particular version and release level of the product.

You may want to create these links if you are developing or running applications and want to avoid having to specify the full path to the product libraries and include files.

Log in as a user with root authority and invoke the **db2ln** command as follows:

```
/opt/IBMdb2/V5.0/db2ln
```

If there are existing links to the `/usr/lib` and `/usr/include` directories from previous versions of DB2, they will automatically be removed by entering the **db2ln** command to create links for this version of DB2. If you want to re-establish the links to the libraries or previous versions, then you must execute the **db2rmln** command from Version 5 before you execute the **db2ln** command from the previous versions of DB2.



Links can be established for only one version of DB2 on a given system.

Chapter 25. Using the Control Center to Configure Server Communications

The information in this section describes how to use the Control Center to configure communications on the server.

To configure an Administration Server for communications, go to Chapter 26, “Setting up Communications on the Server Using the Command Line Processor” on page 251.

The Control Center is a graphical tool available on OS/2 and Windows 32-bit operating systems. Use the Control Center's setup communications function to configure communications on the server. The Control Center allows you to:

- Display the protocols and configuration parameters that a server instance is configured to use.
- Maintain the configured protocols:
  - You can modify the parameter values of a configured protocol.
  - You can add or delete a protocol.

    When you add support for a new protocol to the server system, the setup communications function detects and generates server instance parameter values for the new protocol; you can accept or modify them before use. When you remove support for an existing protocol from the server system, the setup communications function detects the protocol that has been removed and disables its use by the server instance.

    You can add a protocol that has not been detected, however, you must supply all parameter values before you proceed.

The setup communications function can be used to maintain communications of both local and remote server instances.

Configuring DB2 Communications for Local Instances

1. To start the Control Center, do the following:

   On an OS/2 system, double-click on the **Control Center** icon in the **Administration Tools** folder. (The **Administration Tools** folder is located in the DB2 folder.)

   On a Windows NT system, click on **Start** and select **Programs**->**DB2 for Windows NT**->**Administration Tools**->**Control Center**.

2. Click on the **[+]** sign beside the **Systems** icon in the Control Center to get a list of systems.

3. Click on the **[+]** sign beside the system name to get a list of that system's database instances.

4. Select the instance you want to configure and click on the right mouse button.

5. Select the **Setup communications** option from the pop-up menu. The Setup Communications window opens.

6. Use the Setup Communications window to configure communication protocols for the instance. Click on the **Help** push button for more information.

7. You must stop and start the instance for these changes to take effect.

   a. To stop the database manager instance, select the instance, click with the right mouse button and select the **Stop** option from the pop-up menu.

   b. To start the database manager instance, select the instance, click with the right mouse button and select the **Start** option from the pop-up menu.

Modifying an instance's communications settings may require you to update the database connection catalogs on the client.

You do this as follows:

- Using the command line processor on the client, uncatalog and recatalog the communication node and/or database information, depending on the values changed on the server. For more information on cataloging using the command line processor, see Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.
- Using the Client Configuration Assistant on the client, select the database connection you want to change and then click on the **Properties** button. This will launch a Smartguide that will help you with the changes. For more information on cataloging using the Client Configuration Assistant, see Chapter 13, “Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 111.

Configuring DB2 Communications for Remote Instances

1 To start the Control Center, do the following:

On an OS/2 system, double-click on the **Control Center** icon in the **Administration Tools** folder. (The **Administration Tools** folder is located in the DB2 folder.)

On a Windows NT system, click on **Start** and select **Programs**->**DB2 for Windows NT**->**Administration Tools**->**Control Center**.

2 Click on the **[+]** sign beside the **Systems** icon in the Control Center to get a list of systems.

3 Select the **System** icon, click with the right mouse button and select the **Add** option. The Add System window opens.

4 To add a system to the Control Center, you can do one of the following:

a Search the network for known servers.

- Click on the **Refresh** push button. If the *DISCOVER* configuration parameter is set to *ENABLE*, then the Administration Server will locate all servers on the network. For information on the *DISCOVER* parameter, see “Hiding Server Instances and Databases from Discovery” on page 272.
- Select the system you want to add from the **System name** drop-down list.

or

b Enter the server name.

- Enter the hostname or the IP address of the remote DB2 server in the **Host name** field.
- Click on the **Retrieve** push button to obtain this system's information.

5 Click on the **Apply** push button to add the system to the Control Center window.

6 Click on the **Close** push button.

7 Click on the **[+]** sign beside the system name you just added to get a list of that system's database instances.

8 Select the **Instances** folder for the new system and click on the right mouse button.

9 Select the **Add** option. The Add Instance window opens.

10 Click on the **Refresh** push button to obtain a list of available instances.

11 Select the instance that you want to add from the **Remote instance** drop-down list.

12 Click on the **Apply** push button.

13 Click on the **Close** push button.

14 Select the instance you want to configure and click on the right mouse button.

15 Select the **Setup communications** option from the pop-up menu. The Setup Communications window opens.

16 Use the Setup Communications window to configure communication protocols for the instance. Click on the **Help** push button for more information.

17 You must stop and start the instance for these changes to take effect.

a. To stop the database manager instance, select the instance, click with the right mouse button and select the **Stop** option from the pop-up menu.

b. To start the database manager instance, select the instance, click with the right mouse button and select the **Start** option from the pop-up menu.

Modifying an instance's communications settings may require you to update the database connection catalogs on the client.

You do this as follows:

- Using the command line processor on the client, uncatalog and recatalog the communication node and/or database information, depending on the values changed on the server. For more information on cataloging using the command line processor, see Chapter 14, “Configuring Client-to-Server Communications Using the Command Line Processor” on page 117.
- Using the Client Configuration Assistant on the client, select the database connection you want to change and then clicm on the **Properties** button. This will launch a Smartguide that will help you wiht the changes. For more information on cataloging using the Client Configuration Assistant, see Chapter 13, “Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 111

Chapter 26. 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 19, “Entering DB2 Commands and SQL Statements” on page 181. For a description of database naming rules, see Appendix D, “Naming Rules” on page 317.

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, 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

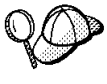
tcpip starts TCP/IP support

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:

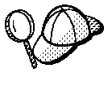
```
db2set DB2COMM=appc,tcpip -i DB2DAS00
db2admin stop
db2admin start
```

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.

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 188.

You are now ready to configure the server workstation to use any of the following communication protocols:

- TCP/IP - see “Configuring TCP/IP on the Server” on page 253
- IPX/SPX - see “Configuring IPX/SPX on the Server” on page 256
- APPC - see “Configuring APPC on the Server” on page 261

Configuring TCP/IP on the Server

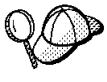
To access a remote server through TCP/IP, you must first have installed and configured communication software for both the client and DB2 server workstations. See “Software Requirements” on page 27 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 34 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 251.

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.

Table 27. TCP/IP Values Required at the Server

Parameter	Explanation	Sample Value	Your Value

Connection Port
• Connection Service name (*svcename*)
• Port number/Protocol (port_number/tcp)
Interrupt Port
• Interrupt Service name
• Port number/Protocol (port_number + 1/tcp)
	Values required in the `services` file.
	The Connection and Interrupt Service names are arbitrary, but each must be unique within the `services` file.
	The Connection and Interrupt Ports must be unique within the `services` file, with the Interrupt Port number equal to the Connection Service Port number plus one.
		`server1`
		`3700/tcp`
		`server1i`
		`3701/tcp`	

Service name (*svcename*)	The name used to update the Service name (*svcename*) parameter in the database manager configuration file at the server. This value must be the same as the Connection Service name specified in the `services` 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 `/etc` directory.

If you are using Network Information Services (NIS) on your network, you must update the `services` file located on your master server.

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:

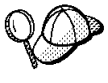
server1	is the Connection Service name
server1i	is the Interrupt Service name
3700 and 3701	are the port numbers for the Connection and Interrupt Ports
`tcp`	is the communication protocol that you are using

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 139.

2. Set up the instance environment and invoke the DB2 command line processor as follows:

   a. Run db2profile or db2cshrc as follows:

      ```
      . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
      source INSTHOME/sqllib/db2cshrc (for C shell)
      ```

      where *INSTHOME* is the home directory of the instance.

   b. Start the DB2 command line processor by entering the **db2** command.

3. 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 7, "Installing DB2 Clients" on page 79 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 27 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 34 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 251.

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 or File Server Addressing.

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.

File Server Addressing:
The DB2 server instance registers its address at the NetWare file server. The client connects to the DB2 server instance through an address stored at a NetWare file server. Configure DB2 servers using this addressing method if they will be accessed by clients using both File Server and Direct Addressing, or just File Server Addressing.

See Table 28 for a list of the supported IPX/SPX addressing methods for your server.

Table 28. IPX/SPX Supported Communication Methods for a DB2 Server

Server Platform	Direct Addressing	File Server Addressing
AIX	√	√
HP-UX	√	√
SCO UnixWare 7	√	
Solaris	√	

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.
   b. Register the server on the NetWare file server (for File Server Addressing only).

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.

Table 29. IPX/SPX Values Required at the Server

Parameter	Explanation	Our Example	Your Value
File server name (*FILESERVER*)	**Direct Addressing** A * value indicates that you are using Direct Addressing. **File Server Addressing** The name of the NetWare file server where the database server instance is registered. This parameter must be entered in UPPERCASE.	**Direct Addressing** * **File Server Addressing** NETWSRV	
DB2 server object name (*OBJECTNAME*)	**Direct Addressing** A * value indicates that you are using Direct Addressing. **File Server Addressing** The database manager server instance, represented as the object *OBJECTNAME* on the NetWare file server. The server's IPX/SPX internetwork address is stored and retrieved from this object. This parameter must be entered in UPPERCASE and be unique on the NetWare file server system.	**Direct Addressing** * **File Server Addressing** DB2INST1	
Socket number (*IPX_SOCKET*)	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	

Naming Restrictions for File Server Addressing

The following characters are not valid for the File server (*FILESERVER*) or the DB2 server object name (*OBJECTNAME*) parameters: / \ : ; , * ?

Step 2. Configure the Server

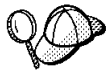
The following steps describe how to configure this protocol on the server. Replace the sample values with your values and record them on your worksheet.

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 139.

2. Set up the instance environment and invoke the DB2 command line processor as follows:

   a. Run db2profile or db2cshrc as follows:

      ```
      . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
      source INSTHOME/sqllib/db2cshrc (for C shell)
      ```

      where *INSTHOME* is the home directory of the instance.

   b. Start the DB2 command line processor by entering the **db2** command.

3. 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
   ```

   Direct Addressing

   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
   ```

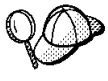
File Server Addressing

For example, if the name of the File server (*FILESERVER*) is NETWSRV, the DB2 server's object name (*OBJECTNAME*) is DB2INST1, and you have chosen a value of 879E for the Socket number (*IPX_SOCKET*) parameter, use:

```
update dbm cfg using fileserver NETWSRV objectname DB2INST1 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`.

Make note of this output on your worksheet for use when you configure an IPX/SPX client using the command line processor.

If you are planning to support clients using only Direct Addressing, you are now finished the configuration and are ready to install a DB2 client. Go to Chapter 7, “Installing DB2 Clients” on page 79 for more information.

B. Register the DB2 Server Instance on the NetWare File Server (for File Server Addressing Only)

The DB2 server must be registered *after* the database manager configuration file has been updated with the IPX/SPX parameters. To register the DB2 server instance at the NetWare File server, enter the following command in the command line processor:

```
register db2 server in nwbindery user USERNAME password PASSWORD
```

Notes:

1. *USERNAME* and *PASSWORD* must be specified in UPPERCASE.
2. The *USERNAME* and *PASSWORD* are used to log on to the NetWare file server and must have Supervisor/Administrator or Workgroup Manager security equivalence.
3. If you want to register at a NetWare 4.x file server (which uses directory services and provides bindery emulation capability), the *USERNAME* used must be created within the same context as the current bindery context used by Directory Services

when it does bindery emulation. The bindery emulation context currently in use can be found by checking the bindery emulation setting on the NetWare 4.x file server (for example, by using the SERVMAN utility). For more information, refer to your IPX/SPX documentation.

Now that you have configured the server, you are ready to install a DB2 client. Go to Chapter 7, "Installing DB2 Clients" on page 79 for more information.

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 27 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 34 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 251.

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.

Table 30. APPC Values Required at the Server

Parameter	Description	Sample Value	Your Value
Transaction program name (*tpname*)	The Transaction program that the server listens on for client requests.	`nyserver`	

Step 2. Configure the Server

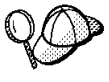
The following steps describe how to configure this protocol on the server. Replace the sample value with your value as recorded on your worksheet.

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 139.

2. Set up the instance environment and invoke the DB2 command line processor as follows:

   a. Run db2profile or db2cshrc as follows:

      ```
      . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
      source INSTHOME/sqllib/db2cshrc (for C shell)
      ```

      where *INSTHOME* is the home directory of the instance.

   b. Start the DB2 command line processor by entering the **db2** command.

3. 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.

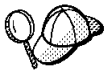
Refer to the *DB2 Connect Enterprise Edition Quick Beginnings* manual for detailed instructions on how to configure your APPC communications subsystem to support communications from the Transaction program that you specified for this server.

If you are using the DB2 Universal Database Workgroup Edition, this manual was not installed on your workstation. Go to the DB2 Product and Service Technical Library located at http://www.software.ibm.com/data/db2/library/ to view this manual.

Chapter 27. Using the Client Configuration Assistant

Use the information in this section to use the Client Configuration Assistant (CCA) to perform administration tasks on OS/2 and Windows 32-bit operating systems.

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.

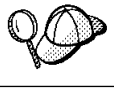


The CCA can be installed on OS/2 and Windows 32-bit operating systems. If it is not installed on your system, you can rerun the setup program and select to install the CCA.

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 266.
- Searching the network for databases, see “Searching the Network for Databases” on page 269.
- Adding database connections manually, see “Manually Configure a Connection to a DB2 Database” on page 274.

Each of these is covered in more detail in the material that follows.

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 273.

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 *admin* parameter 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.

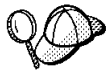
To have the server support Known discovery, set the *discover* parameter in the Administration Server configuration file to *KNOWN*. To have it support Search discovery, set this parameter to *SEARCH* (which also supports all the facilities of Known discovery). To prevent discovery of a server, and all 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).

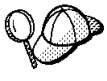
The *discover_comm* parameter is used to control the communication protocols that the server will listen to for search requests from clients, and that clients will use to send out search requests. The *discover_comm* parameter can be set to TCP/IP; this is the only protocol supported by Search discovery.

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 20, “Controlling Your DB2 Environment” on page 187.

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 [ 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.

- 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.

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**.

db2discoverytime and db2nbdiscoverrcvbufs

The *db2discoverytime* 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 *db2discoverytime* registry value to 60 seconds, enter the following command:

  ```
  db2set db2discoverytime=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 28. 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 278.

- 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 278.

- 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 31.

Table 31. 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

HP-UX	TCP/IP
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. Ensure DB2 for MVS/ESA or DB2 for OS/390 is installed and operational on the host.

   Refer to the *DB2 Connectivity Supplement* for information on the configuration required for the DB2 for MVS/ESA or DB2 for OS/390 AR.

2. 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.

3. 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 26, "Setting up Communications on the Server Using the Command Line Processor" on page 251. 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.

4. Update the tables at the DB2 for MVS/ESA or DB2 for OS/390 host.

   Refer to the *DB2 Connectivity Supplement* for more information.

5. 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
   ```

6. Test the host-server DRDA connection by logging onto TSO and using DB2I/SPUFI.

For information on setting up two-phase commit, refer to the on-line *DB2 Connect Enterprise Edition Quick Beginnings* manual.

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.

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.

The following communications subsystems have this restriction:

AIX Bull SNA

Solaris all

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 32 on page 282 identifies which functions are implemented in the AS on the DB2 Universal Database server. The subsequent table lists the supported bind options.

Table 32. 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 33 (Page 1 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Sup-ported	Not Sup-ported	DB2 for MVS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
Package Version Name	**Null**	X		VERSION			VERSION
	Any other value		X				
Bind Existence Checking	**Object existence optional**		X	VALIDATE (**RUN**)[b]	**NOEXIST**	GENLVL(**10**, 11-40)	VALIDATE **RUN**
	Object existence required	X		VALIDATE (BIND)[b]	EXIST	GENLVL(00-09)	VALIDATE BIND
Package Replacement Option	**Replacement allowed**	X		ACTION (**REPLACE**)	**REPLACE**	REPLACE(***YES**)	ACTION **REPLACE**
	Replacement not allowed		X	ACTION(ADD)	NEW	REPLACE(*NO)	ACTION ADD
Package Authorization Option	**Keep authorizations**	X			**KEEP**		RETAIN **YES**
	Revoke authorizations		X		REVOKE		RETAIN NO
Statement String Delimiter (Note 2)	Apostrophe	X		**APOSTSQL**	**SQLAPOST**	OPTION([...] *APOSTSQL) (Note 3)	STRDEL APOSTROPHE
	Double quote		X	QUOTESQL	SQLQUOTE	OPTION([...] *QUOTESQL) (Note 4)	STRDEL QUOTE
Statement Decimal Delimiter (Note 5)	Period	X		**PERIOD**	**PERIOD**	OPTION([...] *PERIOD) or OPTION([...] ***SYSVAL**) (Note 6)	DECDEL PERIOD
	Comma		X	COMMA	COMMA	OPTION([...] *COMMA) or OPTION([...] ***SYSVAL**) (Note 6)	DECDEL COMMA
Date Format (Note 7)	**ISO**	X		DATE(ISO) (Note 8)	DATE(**ISO**)	DATFMT(*ISO) (Note 8)	DATETIME **ISO** (Note 9)
	USA	X		DATE(USA)	DATE(USA)	DATFMT(*USA)	DATETIME USA
	EUR	X		DATE(EUR)	DATE(EUR)	DATFMT(*EUR)	DATETIME EUR
	JIS	X		DATE(JIS)	DATE(JIS)	DATFMT(*JIS)	DATETIME JIS
Time Format (Note 7)	**ISO**	X		TIME(ISO) (Note 8)	TIME(**ISO**)	TIMFMT(*ISO) (Note 8)	DATETIME **ISO** (Note 9)
	USA	X		TIME(USA)	TIME(USA)	TIMFMT(*USA)	DATETIME USA
	EUR	X		TIME(EUR)	TIME(EUR)	TIMFMT(*EUR)	DATETIME EUR
	JIS	X		TIME(JIS)	TIME(JIS)	TIMFMT(*JIS)	DATETIME JIS
Package Isolation Level (Note 10)	Repeatable read	X		ISOLATION(**RR**)[b]	ISOLATION(**RR**)		ISOLATION RR
	Read Stability (All)	X			ISOLATION(RS)	COMMIT(*ALL)	ISOLATION RS

Table 33 (Page 2 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Sup-ported	Not Sup-ported	DB2 for MVS/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	DFTRDBCOL	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 SBCS	SBCS	X			CHARSUB(SBCS)		CHARSUB SBCS
If Default CCSID is SBCS	MBCS		X		CHARSUB(MBCS)		CHARSUB MBCS
If Default CCSID is MBCS	BIT		X		CHARSUB(BIT)		CHARSUB BIT

Table 33 (Page 3 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Sup-ported	Not Sup-ported	DB2 for MVS/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	**Value specified when DB2 database was created**	X			CCSIDSBCS() CCSIDGRAPHIC() CCSIDMIXED()		CCSIDS CCSIDG CCSIDM
	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	**Null**	X		REPLVER[b]			REPLVER
	Any other value		X				
Generic Bind Option	**Null**		X				GENERIC
	Any other value		X				
Package Authorization Rule	**Requester**	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	**1**		X (Note 13)				DEGREE 1
	n		X				DEGREE n
	ANY		X				DEGREE ANY

Table 33 (Page 4 of 4). Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Sup-ported	Not Sup-ported	DB2 for MVS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option

Note:

(*) Default values are in **bold**.
(1) Most are precompile options. Bind options are indicated by b.
(2) Defaults to what the target database supports. For DB2 the default is apostrophe.
(3) Default for non-COBOL applications.
(4) Default for COBOL applications.
(5) Defaults to what the target database supports. For DB2 the default is period.
(6) Depending on the installation, *SYSVAL is equivalent to *PERIOD or *COMMA.
(7) Date and time formats must be the same for the DB2 DRDA AS.
(8) Default is dependent on the installation.
(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.
(10) Package isolation level has no default because an explicit value is always present in the DRDA datastream.
(11) The isolation level will be escalated to Uncommitted Read (Change).
(12) Defaults to what the target database supports. For DB2 the default is 31.
(13) All variables will default to 1.

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 `INSTHOME/sqllib/misc` where *INSTHOME* is the home directory of the instance owner.

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 pages, 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.

Chapter 29. Removing DB2 Products

This section shows you how to remove DB2 products. You need to perform the following steps:

1. Stop the Administration Server.
2. Stop all DB2 Instances.
3. Remove the Administration Server.
4. Remove DB2 Instances - this step is optional.
5. Remove the DB2 products.

Step 1. Stop the Administration Server

You must stop the Administration Server before you remove DB2 products. To stop the Administration Server, you need to perform the following steps:

1. Log in as user with root authority.
2. Obtain the name of the Administration Server using the following command:

   ```
   DB2DIR/bin/dasilist
   ```

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris

3. Stop the Administration Server.

   To stop the Administration Server, you must perform the following steps:

   a. Log in as the Administration Server owner.

   b. Run the start up script as follows:

      ```
      . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
      source INSTHOME/sqllib/db2cshrc (for C shell)
      ```

      where *INSTHOME* is the home directory of the instance.

   c. Stop the Administration Server using the **db2admin** command as follows:

      ```
      db2admin stop
      ```

where *DB2DIR*	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	on HP-UX, SCO UnixWare 7, or Solaris

Step 2. Stop All DB2 Instances

You must stop all DB2 Instances before you remove DB2 products. To stop DB2 instances, you need to perform the following steps:

1. Obtain a list of the names of all DB2 instances on your system using the following command:

   ```
   DB2DIR/bin/db2ilist
   ```

2. Stop the instance.

   To stop a database instance:

   a. Log in as the instance owner.

   b. Run the start up script as follows:

      ```
      . INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
      source INSTHOME/sqllib/db2cshrc (for C shell)
      ```

      where `INSTHOME` is the home directory of the instance that you want to use.

   c. Stop the DB2 database manager by entering the **db2stop** command from a command line.

3. Repeat steps 2 and 3 for each instance.

Step 3. Remove the Administration Server

You must remove the Administration Server before you remove DB2 products. To remove the Administration Server, you need to perform the following steps:

1. Log in as user with root authority.
2. Obtain the name of the Administration Server using the following command:

   ```
   DB2DIR/bin/dasilist
   ```

3. Log in as the Administration Server owner.
4. Run the start up script as follows:

```
. INSTHOME/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/db2cshrc (for C shell)
```

where *INSTHOME* is the home directory of the instance.

5. Back up the files in the ASHOME/sqllib directory, if needed, where ASHOME is the home directory of the Administrtion Server.
6. Log off.
7. Log in as *root* and remove the Administration Server using the **dasidrop** command as follows:

   ```
   DB2DIR/instance/dasidrop ASName
   ```

   where *ASName* is the name of the instance being removed.

Note: The **dasidrop** command removes the /sqllib directory under the home direcory of the Administration Server.

Step 4. Remove DB2 Instances

You can optionally remove some or all of the DB2 Version 5 instances on your system. Once an instance is removed, all the DB2 databases owned by the instance, if any, will not be usable. Remove DB2 instances only if you are not planning to use DB2 Version 5 products, or if you do not want to migrate existing instances to a later version of DB2. See Chapter 6, “Migrating from Previous Versions and Releases” on page 57 for more information on how to migrate or upgrade to higher versions of DB2.

To remove an instance, perform the following steps:

1. Log in as the instance owner.
2. Stop the instance by issuing the **db2stop** command.
3. Back up files in the INSTHOME/sqllib directory, if needed.

   You might want to save the database manager configuration file, db2systm, the db2nodes.cfg file, or user defined function or fenced stored procedure applications in INSTHOME/sqllib/function, where INSTHOME is the home directory of the instance owner.
4. Log off as the instance owner.
5. Log in as user with root authority.
6. Remove the instance using the **db2idrop** command. You can enter the DB2IDROP command as follows:

   ```
   DB2DIR/instance/db2idrop InstName
   ```

InstName is the login name of the instance.

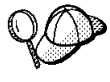
where DB2DIR	= /usr/lpp/db2_05_00	on AIX
	= /opt/IBMdb2/V5.0	HP-UX, SCO UnixWare 7, Solaris, or SGI IRIX
	= /usr/IBMdb2/V2.1	on SCO OpenServer
	= /opt/IBMdb2/V2.1	on SINIX

7. Optionally, as a user with root authority, remove the instance owner's user ID and group (if used only for that instance). Do not remove these if you are planning to re-create the instance.

   This step is optional since the instance owner and the instance owner group may be used for other purposes.

The **db2idrop** command removes the instance entry from the list of instances and removes the `INSTHOME/sqllib` directory.

Step 5. Remove DB2 Products



You must stop all outstanding DB2 processes before removing the DB2 products.

The following steps describe how to remove DB2 products on UNIX operating systems.

Remove DB2 Products on AIX Systems

You can remove the DB2 products on AIX using the **SMIT** interface as follows:

1. Log in as user with root authority.
2. Type **smit install_remove** to proceed directly to the Remove Software Products screen.
3. Press F4 to display a list of the software to remove. Press F7 at some or all of the entries that have a prefix of `db2_05_00`.

   Note: For a detailed description of the software packages, see Appendix B, “Contents of the DB2 Products” on page 309.

4. Press Enter to start removing the DB2 products.

You can also remove all DB2 Version 5 products on AIX, using the **installp** command with the de-install option:

```
installp -u db2_05_00
```

Remove DB2 Products on HP-UX Systems

You can remove the DB2 products on the HP-UX operating system using the **swremove** program as follows:

1. Log in as user with root authority.
2. Use `swremove` to remove some or all of the DB2 Version 5 products.

Remove DB2 Products on SCO UnixWare 7 Systems

You can remove the DB2 products on the SCO UnixWare 7 operating system using the **scoadmin** program as follows:

1. Log in as user with root authority.
2. Use the **scoadmin** command to remove some or all of the DB2 Version 5 products.

Remove DB2 Products on Solaris Systems

You can remove the DB2 products on the Solaris operating system using the **pkgrm** program as follows:

1. Log in as user with root authority.
2. Determine the packages for all DB2 for Solaris related products you have installed on your system by entering:

   ```
   pkginfo | grep -i db2 | grep 50
   ```

3. Remove all packages listed in step 2 with the **pkgrm** command.
4. Select **Yes** at the prompt for each package to be removed.

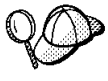
   Before removing a package, all its dependent packages must be removed first. You must remove packages in the following order:

   a. db2ht*%L*50
   b. db2ms*%L*50
   c. db2adts50

d. db2adt50
e. db2cent50
f. db2entp50
g. db2wkgp50
h. db2cnvt50
i. db2cnvc50
j. db2cnvk50
k. db2cnvj50
l. db2smpl50
m. db2repl50
n. db2conn50
o. db2das50
p. db2cipx50
q. db2cdrd50
r. db2csna50
s. db2crte50
t. db2engn50
u. db2rte50
v. db2jdbc50
w. db2odbc50
x. db2cliv50

Where *%L* is the language/locale identifier

Note: See Appendix B, "Contents of the DB2 Products" on page 309 for a list of packages for Solaris.



You can also use the **db2_deinstall** command to remove products as follows:

1. Mount the CD-ROM as described in "Step 3. Mount the CD-ROM" on page 46.
2. Go the appropriate `/cdrom` mount directory.
3. Run the **db2_deinstall** command.

   The command lists the products installed and will de-install them in the correct order.

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 307 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...
Add Database	Catalog a database on a client workstation.	From the Client Configuration Assistant, click on **Add**.
Create Database	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**.
Performance Configuration	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**.
Backup Database	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**.
Restore Database	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...
Create Table	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**.
Create Table Space	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 307.

Type of Help	Contents	How to Access...
Command Help	Explains the syntax of commands in the command line processor.	From the command line processor in interactive mode, enter: **?** *command* where *command* 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.
Control Center Help	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...
Message Help	Describes the cause of a message, and any action you should take.	From the command line processor in interactive mode, enter: **?** *XXXnnnnn* where *XXXnnnnn* is a valid message identifier. For example, **? SQL30081** displays help about the SQL30081 message. To view message help one screen at a time, enter: **?** *XXXnnnnn* **| more** To save message help in a file, enter: **?** *XXXnnnnn* > *filename.ext* where *filename.ext* is the file where you want to save the message help.
SQL Help	Explains the syntax of SQL statements.	From the command line processor in interactive mode, enter: **help** *statement* where *statement* is an SQL statement. For example, **help** *SELECT* displays help about the SELECT statement.
SQLSTATE Help	Explains SQL states and class codes.	From the command line processor in interactive mode, enter: **?** *sqlstate* or **?** *class-code* where *sqlstate* is a valid five-digit SQL state and the *class-code* is first two digits of the SQL state. For example, **? 08003** displays help for the 08003 SQL state, while **? 08** displays help for the 08 class code.

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 304.
Search See "Searching Online Books" on page 305.
Print See "Printing the PostScript Books" on page 305.
Order See "Ordering the Printed DB2 Books" on page 306.

Book Name	Book Description	Form Number File Name
	Cross-Platform Books	
Administration Getting Started	Introduces basic DB2 database administration concepts and tasks, and walks you through the primary administrative tasks.	S10J-8154 db2k0x50
Administration Guide	Contains information required to design, implement, and maintain a database to be accessed either locally or in a client/server environment.	S10J-8157 db2d0x51
API Reference	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
CLI Guide and Reference	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
Command Reference	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
DB2 Connect Enterprise Edition Quick Beginnings	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
DB2 Connect Personal Edition Quick Beginnings	Provides planning, installing, configuring, and using information for DB2 Connect Personal Edition.	S10J-8162 db2c1x51
DB2 Connect User's Guide	Provides concepts, programming and general using information about the DB2 Connect products.	S10J-8163 db2c0x51
DB2 Connectivity Supplement	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
Embedded SQL Programming Guide	Explains how to develop applications that access DB2 databases using embedded SQL, and includes discussions about programming techniques and performance considerations.	S10J-8158 db2a0x50
Glossary	Provides a comprehensive list of all DB2 terms and definitions. **Note:** Available in HTML format only.	No form number db2t0x50
Installing and Configuring DB2 Clients	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
Master Index	Contains a cross reference to the major topics covered in the DB2 library. **Note:** Available in PostScript format and hardcopy only.	S10J-8170 db2w0x50
Messages Reference	Lists messages and codes issued by DB2, and describes the actions you should take.	S10J-8168 db2m0x51
DB2 Replication Guide and Reference	Provides planning, configuring, administering, and using information for the IBM Replication tools supplied with DB2.	S95H-0999 db2e0x52
Road Map to DB2 Programming	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
SQL Getting Started	Introduces SQL concepts, and provides examples for many constructs and tasks.	S10J-8156 db2y0x50
SQL Reference	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
System Monitor Guide and Reference	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
Troubleshooting Guide	Helps you determine the source of errors, recover from problems, and use diagnostic tools in consultation with DB2 Customer Service.	S10J-8169 db2p0x50
What's New	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	
Building Applications for UNIX Environments	Provides environment setup information and step-by-step instructions to compile, link, and run DB2 applications on a UNIX system.	S10J-8161 db2axx51
Building Applications for Windows and OS/2 Environments	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
DB2 Personal Edition Quick Beginnings	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
DB2 SDK for Macintosh Building Your Applications	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
DB2 SDK for SCO OpenServer Building Your Applications	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
DB2 SDK for SINIX Building Your Applications	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
Quick Beginnings for OS/2	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
Quick Beginnings for UNIX	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
Quick Beginnings for Windows NT	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
DB2 Extended Enterprise Edition for UNIX Quick Beginnings	Provides planning, installing, configuring, and using information for DB2 Universal Database Extended Enterprise Edition for UNIX. This book supercedes the *DB2 Extended Enterprise Edition Quick Beginnings for AIX* book, and is suitable for use with all versions of DB2 Extended Enterprise Edition that run on UNIX-based platforms.	S99H-8314 db2v3x51
DB2 Extended Enterprise Edition for Windows NT Quick Beginnings	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 307 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 307 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 300.

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 `SBOF-8915-00`. The form number for the set of books updated for Version 5.2 is `SBOF-8921-00`. The form number for the books listed under the heading "Cross-Platform Books" is `SBOF-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 300. 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. Contents of the DB2 Products

This section lists the contents of various DB2 Universal Database products.

Packaging

The following DB2 Universal Database components, packages, or filesets are available for installation:

Table 34 (Page 1 of 2). DB2 Components, Packages or Filesets

Description	Fileset Name		
	DB2 for AIX	DB2 for HP-UX	DB2 for Solaris / DB2 for SCO UnixWare 7
DB2 Client	db2_05_00.client	DB2V5CAE.client	db2cliv50
Open Database Connectivity (ODBC)	db2_05_00.odbc	DB2V5CAE.odbc	db2odbc50
Java Support (JDBC)	db2_05_00.jdbc	DB2V5CAE.jdbc	db2jdbc50
DB2 Web Control Center	db2_05_00.wcc	DB2V5CAE.wcc	db2wcc50
Administration Server	db2_05_00.das	DB2V5WGRP.das	db2das50
DB2 Run-time Environment	db2_05_00.db2.rte	DB2V5WGRP.db2rte	db2rte50
DB2 Sample Database Source	db2_05_00.db2.samples	DB2V5WGRP.dbsmpl	db2smpl50
DB2 Engine	db2_05_00.db2.engn	DB2V5WGRP.db2engn	db2engn50
DB2 Replication	db2_05_00.repl	DB2V5WGRP.repl	db2repl50
DB2 Connect	db2_05_00.conn	DB2V5CONN.conn	db2conn50
DB2 Communication Support - TCP/IP	db2_05_00.cs.rte	DB2V5WGRP.csrte	db2crte50
DB2 Communication Support - SNA	db2_05_00.cs.sna	DB2V5WGRP.cssna	db2csna50
DB2 Communication Support - DRDA Application Server	db2_05_00.cs.drda	DB2V5WGRP.csdrda	db2cdrd50
DB2 Communication Support - IPX	db2_05_00.cs.ipx	DB2V5WGRP.csipx	db2cipx50
License Support of DB2 Enterprise Edition	db2_05_00.esrv	DB2V5ENTP.esrv	db2entp50
License Support of DB2 Connect Enterprise Edition	db2_05_00.cent	DB2V5CONN.cent	db2cent50
DB2 Application Development Tools (ADT)	db2_05_00.adt.rte	DB2V5SDK2.adtrte	db2adt50

Table 34 (Page 2 of 2). DB2 Components, Packages or Filesets

Description	Fileset Name		
	DB2 for AIX	DB2 for HP-UX	DB2 for Solaris / DB2 for SCO UnixWare 7
DB2 ADT Sample Programs	db2_05_00.adt.samples	DB2V5SDK.adtsamp	db2adts50
Code Page Conversion Tables - Uni Code Support	db2_05_00.cnvucs	DB2V5CAE.convucs	db2cucs50
Code Page Conversion Tables - Japanese	db2_05_00.conv.jp	DB2V5WGRP.convjp	db2cnvj50
Code Page Conversion Tables - Korean	db2_05_00.conv.kr	DB2V5WGRP.convkr	db2cnvk50
Code Page Conversion Tables - Simplified Chinese	db2_05_00.conv.sch	DB2V5WGRP.convsch	db2cnvc50
Code Page Conversion Tables - Traditional Chinese	db2_05_00.conv.tch	DB2V5WGRP.convtch	db2cnvt50
DB2 Product Messages - %L [1, 2]	db2_05_00.msg.%L[4]	DB2V5MSG.%L	db2ms%L50
DB2 Product Document (HTML) - %L [1]	db2_05_00.html.%L[4]	DB2V5HTML.%L	db2ht%L50

Notes:

1. %L in the fileset name represents the locale name. There is a separate fileset for each locale. For Solaris and HP-UX systems, abbreviated names are used for some of the locales.
2. English messages are always installed.
3. DB2 supports a number of locales. However, DB2 messages are not translated in every supported locale. For a complete list of DB2 supported locales, see the *"National Language Support (NLS)"* appendix in the *Administration Guide*.
4. While DB2 Product Messages and Documentation are translated in several locales, not every message catalog or book is translated in every locale.

Abbreviated names for locale names are given in the following table. %L represents the abbreviated locale name.

Table 35. Abbreviated Names for Locales

DB2 for HP-UX		DB2 for Solaris/ DB2 for SCO UnixWare 7	
Locale	**%L**	**Locale**	**%L**
bg_BG.iso88595	bg_BG_I	-	-
da_DK.iso88591	da_DK_I	da	da
da_DK.roman8	da_DK_R	-	-
de_DE.iso88591	de_DE_I	de	de
de_DE.roman8	de_DE_R	-	-
en_US.iso88591	en_US_I	en_US	en
en_US.roman8	en_US_R	-	-
es_ES.iso88591	es_ES_I	es	es
es_ES.roman8	es_ES_R	-	-
fi_FI.iso88591	fi_FI_I	fi	fi
fi_FI.roman8	fi_FI_R	-	-
fr_FR.iso88591	fr_FR_I	fr	fr
fr_FR.roman8	fr_FR_R	-	-
ja_JP.eucJP	ja_JP	ja	ja
ko_KR.eucKR	ko_KR	ko	ko
no_NO.iso88591	no_NO_I	no	no
no_NO.roman8	no_NO_R	-	-
pt_BR.iso88591	pt_BR_I	pt	pt
pt_BR.roman8	pt_BR_R	-	-
sl_SI.iso88592	sl_SI_I	sl	sl
sv_SE.iso88591	sv_SE_I	sv	sv
sv_SE.roman8	sv_SE_R	-	-
zh_CN.eucCN	zh_CN	zh	zh
zh_TW.eucTW	zh_TW	tw	tw

Products and Selectable Components

Table 36 on page 312 lists the DB2 Universal Database products and selectable components that you can install on a DB2 Server.

Table 36 (Page 1 of 2). Components for DB2 Products

Product / Component Description	DB2 Client Application Enabler	DB2 Universal Database Workgroup Edition	DB2 Universal Database Enterprise Edition	DB2 Connect
DB2 Client Application Enabler	√√	√√	√√	√√
Open Database Connectivity (ODBC) support	√	√	√	√
Java Support (JDBC)	√	√	√	√
DB2 Web Control Center support	√	√	√	√
Administration Server	n/a	√√	√√	√√
DB2 Run-time Environment	n/a	√√	√√	√√
DB2 Sample Database Source	n/a	√√	√√	n/a
DB2 Engine	n/a	√√	√√	n/a
DB2 Connect	n/a	√√	√√	√√
Communication Support for TCP/IP	n/a	√√	√√	√√
Communication Support for IPX/SPX	n/a	√√	√√	√√
Communication Support for SNA	n/a	√√	√√	√√
Communication Support for DRDA Application Server	n/a	√√	√√	√√
Code Page Conversion Tables - Japanese	n/a	√	√	√
Code Page Conversion Tables - Korean	n/a	√	√	√
Code Page Conversion Tables - Simplified Chinese	n/a	√	√	√
Code Page Conversion Tables - Traditional Chinese	n/a	√	√	√
Licensed Support for DB2 Universal Database Workgroup Edition	n/a	√√	n/a	n/a
License Support for DB2 Universal Database Enterprise Edition	n/a	n/a	√√	n/a
Licensed Support for DB2 Connect	n/a	n/a	n/a	√√
DB2 Replication	n/a	√	√	√
DB2 Product Messages (non-English)[1]	o	o	o	o
DB2 Product Library (HTML)[1]	o	o	o	o

Table 36 (Page 2 of 2). Components for DB2 Products

Product / Component Description	DB2 Client Application Enabler	DB2 Universal Database Workgroup Edition	DB2 Universal Database Enterprise Edition	DB2 Connect

√√	This is a required component that must be installed.
√	This is a recommended component that is installed by default. You can choose not to install this component.
o	This is an optional component that is not installed by default. If you want to install it, you must select it.
n/a	This component is not available for installation.

Notes:

1. There is a separate component of the DB2 Product Messages and the DB2 Product Library for each locale.
2. The DB2 SDK is available only with the SDK Pack. Refer to “Developing Applications Using the DB2 Software Developer's Kit” on page 13.

Prerequisites

The DB2 filesets are listed below with their corresponding prerequisites.

Table 37. Prerequisites for DB2 Filesets

Fileset Description	Prerequisite Fileset
DB2 Client Application Enabler	Appropriate version of the operating system
Open Database Connectivity (ODBC) support	DB2 Client Application Enabler
Java Support (JDBC)	DB2 Client Application Enabler
DB2 Web Control Center support	DB2 Client Application Enabler
Administration Server	DB2 Run-time Environment
DB2 Run-time Environment	DB2 Client Application Enabler
DB2 Sample Database Source	DB2 Engine
DB2 Engine	DB2 Run-time Environment
DB2 Replication	DB2 Run-time Environment
DB2 Connect	DB2 Run-time Environment
Communication Support for TCP/IP	DB2 Run-time Environment
Communication Support for IPX/SPX	DB2 Run-time Environment
Communication Support for SNA	Communication Support for TCP/IP
Communication Support for DRDA Application Server	Communication Support for SNA
Licensed Support for DB2 Universal Database: Enterprise Edition	DB2 Engine
Licensed Support for DB2 Connect Enterprise Edition	DB2 Connect
DB2 Application Development Tools (ADT)	DB2 Client Application Enabler
DB2 ADT Sample Programs	DB2 Application Development Tools
DB2 Product Messages	DB2 Client Application Enabler
DB2 Product Library (HTML)	DB2 Client Application Enabler

Appendix C. National Language Support (NLS)

This section contains information about the National Language Support (NLS) provided by DB2, including information about supported locales and code sets. For more detailed information on developing applications, refer to the *Embedded SQL Programming Guide*.

Language and Codeset Support for UNIX Operating Systems

DB2 supports many code sets and locales without translating the messages for the corresponding languages. Supporting a locale means that you can create and use a database in that locale, but you may have to view all panels and messages in a different language, if translated messages are not available in DB2. For a complete list of locales supported, refer to the *Administration Guide* .

If you want to operate in a different language environment, do the following:

1. Ensure that the appropriate message option for the desired language has been installed.
2. Set the *LANG* environment variable to the desired locale.

   For example, to use `fr_FR` messages on DB2 for AIX, you must have the `fr_FR` message option installed and must set *LANG* to `fr_FR`.

The selected message catalog filesets are placed in the following directories on the target workstation:

DB2 for AIX	`/usr/lpp/db2_05_00/msg/%L`
DB2 for HP-UX	`/opt/IBMdb2/V5.0/msg/%L`
DB2 for SCO UnixWare 7	`/opt/IBMdb2/V5.0/msg/%L`
DB2 for Solaris	`/opt/IBMdb2/V5.0/msg/%L`

where *%L* is equal to the locale name of the message catalog.

Code Page and Language Support for OS/2 and Windows Operating Environments

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 38 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*.

Table 38. Languages and Code Pages

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

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 317.

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.

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 317.

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.

Usernames, Group Names, and Instance Names

Usernames are the identifiers assigned to individual users. When naming users, groups, or instances, see "General Naming Rules" on page 317.

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 317.

In addition, the name you specify:

- Can contain 1 to 8 characters;
- Cannot include &, #, and @;
- Must unique within the network.

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 317.

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
ADSTAR
AISPO
AIX
AIXwindows
AnyNet
APPN
AS/400
CICS
C Set++
C/370
DATABASE 2
DatagLANce
DataHub
DataJoiner
DataPropagator
DataRefresher
DB2
Distributed Relational Database Architecture
DRDA
Extended Services
FFST
First Failure Support Technology
IBM
IMS
Lan Distance
MVS/ESA
MVS/XA
NetView
OS/400
OS/390
OS/2
PowerPC
QMF
RACF
RISC System/6000
SAA
SP
SQL/DS
SQL/400
S/370
System/370
System/390
SystemView
VisualAge
VM/ESA
VSE/ESA
VTAM
WIN-OS/2

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.

Index

Special Characters

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

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

Part Number: 04L7807

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.