IBM Database Server for OS/2 Warp Up and Running! Version 4

IBM Database Server for OS/2 Warp

Up and Running!

Version 4

IBM

IBM Database Server for OS/2 Warp

Up and Running!

Version 4

> **Note**
>
> Before using this information and the product it supports, be sure to read the general information under Appendix D, "Notices" on page 159.

Contents

Welcome to Database Server

This book provides an **easy start** for users who want to install and use the basic functions of Database Server in an uncomplicated environment.

Users who need more advanced instructions or who are interested in topics such as migration, incompatibilities, or a complete listing of the features available, should read the *Planning Guide* and *Installation and Operation Guide* manuals. The road map that follows makes it easy to find the specific information you might need.

Part 1 gives a brief overview of the IBM software servers and a more extensive overview of Database Server.

Part 2 provides planning and installing information for Database Server. Information is provided for server and client components.

Part 3 provides information on using Database Server and viewing, printing, and ordering additional information.

Road Map

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

If you want to ..	Refer to...
Use a fast path to install and use all components.	Chapter 8, "Getting Started with the Database Server" on page 91
Read about DB2 and related products.	Chapter 2, "More about Database Server" on page 5
Install Database Server components from a CD-ROM drive.	Chapter 4, "Installing Servers" on page 31
Install the OS/2 and Windows client products.	Chapter 5, "Installing Clients" on page 39
Install the Database Server products using SystemView LAN.	Appendix B, "Remote CID Installation Using SystemView" on page 153
Install the client products for Macintosh, AIX, HP-UX, Solaris, and DOS.	The *Installing and Using Clients* book for each of these platforms. These books are included on the CD-ROM in softcopy format.
Use the Database Server components.	Chapter 8, "Getting Started with the Database Server" on page 91
Use the clients.	"Running Your Own Applications" on page 95
View or print online documentation.	Chapter 12, "Obtaining Additional Information" on page 139
Set up a DB2 Call Level Interface or ODBC environment to enable ODBC applications such as Lotus Approach to access DB2 data.	"Running ODBC-Enabled Applications" on page 99
Bind applications and utilities to a DRDA server to give applications access to host data.	"Completing the Configuration" on page 85
Control access, authentication, password and user ID management.	"Chapter 9" of the *DB2 Installation and Operation Guide* or "Chapter 6" of the *Installing and Using Clients* manual
Use the command line processor to enter SQL statements interactively or to enter DB2 commands.	"Using the Command Line Processor" on page 122
Migrate from a previous version of DB2.	"Chapter 2" of the *DB2 Installation and Operation Guide* or "Chapter 3" of the *Installing and Using Clients* manual
Set up and use the DRDA Application Server feature to have host applications access LAN data.	"Chapter 7" of the *DB2 Installation and Operation Guide*
Print or view any one of the books included with this product.	Chapter 12, "Obtaining Additional Information" on page 139

If you want to ..	Refer to...
Look up a product, task, or topic in the index.	*DB2 Master Index*
Learn about last-minute changes to the product.	The README files in the CD-ROM directories

To locate information on other topics, see Chapter 12, "Obtaining Additional Information" on page 139 for a complete description of the DB2 library.

Conventions

You will find this book easier to use if you look for these simple conventions:

- **Boldface type** indicates the name of an item you need to select.

- *Italics type* indicates new terms or book titles.

- `Monospace type` indicates an example or text that is displayed on the screen.

- UPPERCASE TYPE indicates a file name, command name, or acronym.

This book also uses *icons* (pictures) in the margins and text to help you find different types of information. Icons represent objects that you can work with. In this book, a GUI icon in the margin marks a place in the text where that icon (or something related to it) is being discussed. Here are some examples of icons you will see:

This icon marks a Note— important information that can affect the operation of the product or the completion of a task.

This icon marks a Tip— additional information that can help you complete a task with fewer steps.

This is a fast path icon. A fast path section is provided in the beginning of many procedures. Follow fast paths when you want to get started quickly.

This is the command line processor icon. It represents actions that are to be performed with the command line processor. All Database Server components have the command line processor built-in.

This icon marks Database Director tasks. Use the Database Director to perform the

actions listed. DB2 Server, DB2 Software Developer's Kit, and Administrator's Toolkit
components can use this tool.

This is the Visual Explain icon. This icon represents actions that are to be performed with
the Visual Explain tool. DB2 Server, DB2 Software Developer's Kit, and Administrator's
Toolkit components can use this tool.

This icon marks DB2 Client Setup tasks. Use the DB2 Client Setup to perform the actions
listed. DB2 Client Setup is available for all OS/2 and Windows 3.1 clients.

Part 1. The IBM Software Servers

Chapter 1. Introducing IBM's Software Servers

Congratulations on choosing the IBM Database Server for OS/2. This is one of the most significant enhancements you can make to your network this year! You are now on the easy path to using versatile client-server networks.

This exciting new server and client software is part of a family of IBM and Lotus products. It will help you to quickly install and make use of multi-functional, compatible servers. These servers cover a broad range of needs: from an entry-level database server to a server for comprehensive systems management of sophisticated networks. Other technology at your fingertips enables you to quickly get up and running as a server on the Internet, doing guaranteed transaction processing, work group functions with Lotus Notes, and many other functions.

This family of products has been designed to satisfy customers' needs for:

- Adding value to existing environments in a complementary manner with software servers that were designed to work together

- Simpler, more reliable client/server networks

- Working with all of the popular clients (such as Windows, Windows 95, Windows NT, Macintosh, OS/2, and UNIX) and servers (such as OS/2, Windows NT, OS/400, AIX and other UNIX systems, and MVS)

- Built-in system services to enable mobile and remote users to securely access and update your business data, regardless of where they are or when they do it

- Easy growth from the smallest departmental servers to the largest systems of a modern enterprise

- Comprehensive systems management that is easy to use and powerful enough to look after work groups and interconnected networks

- A reliable environment on which to build business-critical solutions with the performance and availability you require

When you combine IBM Database Server with other software in this family from IBM and Lotus, you get an integrated software environment that will support your applications today and lay the foundation for your applications in the future. Other software suppliers have expressed interest in or have already built integrated solutions using members of this IBM software family.

This manual describes the IBM Database Server for OS/2. The Database Server is the client-server database software you'll want for today's networks. DB2 is based on IBM's industry-leading relational database technology, giving you integrated software that's easy to use and install.

You get support for all types of data—text, graphics, audio, and video. You will also get support for hundreds of third-party applications and tools like Lotus Approach, Lotus Notes, Microsoft Access, and PowerBuilder.

Database Server can be installed with currently supported releases of OS/2. (See "Software Requirements" on page 27 for a complete list of software requirements.) OS/2 Warp Server is the preferred network operating system base, providing simplified installation and the most complete application server environment where personal computer technology is deployed.

To learn how to install IBM's Database Server, refer to Chapter 4, "Installing Servers" on page 31.

Chapter 2. More about Database Server

This chapter describes the products and the functions that are part of the Database Server.

Go directly to Chapter 4, "Installing Servers" on page 31 if you want to get started installing Database Server right away!

Database Server Version 4 consists of some of the already existing DB2 Version 2 products and books, and some new ones. It includes:

- The **DB2 Server** which is a *relational* database management system containing functions and tools that enable users to create, update, control, and manage relational databases using Structured Query Language (SQL).

- The **DB2 Software Developer's Kit** which provides an OS/2 development environment for application developers.

- The **DB2 World Wide Web Connection** which enables application developers to create Internet applications that access data from DB2 databases.

In addition to these products, you get:

- A copy of the *Client Pack* CD-ROM which contains the **DB2 Client Application Enabler** and the **Administrator's Toolkit** programs. The DB2 Client Application Enabler enables client workstations from a variety of platforms to connect to any DB2 Server while the Administrator's Toolkit provides graphical database administration tools for OS/2 and AIX client workstations.

- An evaluation copy of the **DDCS Multi-User Gateway** product which provides access to DB2 databases residing on host systems such as MVS/ESA, VM, VSE, and OS/400. Refer to "License Information" on page 29 for licensing information.

The documentation included on the Database Server Version 4 CD-ROM consists of the DB2 Version 2 documentation. Any references to hardcopy or softcopy books refer to the DB2 books.

These products are described in the following sections.

Working with Data Using the DB2 Server

The **DB2 Server** is a collection of products that provide powerful database management on a variety of platforms, graphical tools for tuning performance, the ability to manage all servers from a single site, powerful application development, and SQL query processing.

Data managed by DB2 Servers can be accessed and manipulated by applications on client workstations on a variety of platforms. See "Accessing Data Through DB2 Server" for information about which platforms are supported.

DB2 Server supports a variety of communication protocols and includes the **DB2 database engine** as well as the following client components:

- The **DB2 Client Application Enabler** component for accessing remote DB2 servers

- The **Database Director** tool for administrative tasks such as configuring the system, managing directories, backing up and recovering the system, and managing media

- The **Performance Monitor** tool for monitoring the performance of DB2 for tuning purposes

- The **Visual Explain** tool for analyzing query access plans. (Visual Explain is installed as part of the Database Director.)

These components are described in the following sections.

Accessing Data Through DB2 Server

The DB2 Client Application Enabler provides a run-time environment for client workstations that want to access local and remote databases. DB2 Client Application Enabler supports clients on the following platforms:

- OS/2
- Windows (Windows 95, Windows 3.1, Windows NT)
- UNIX (AIX, HP-UX, Sun Solaris)
- Macintosh
- DOS

The support for all these platforms is available on the *Client Pack* CD-ROM. The Client Pack also includes the **Administrator's Toolkit** which contains a suite of tools to help manage databases from OS/2 or AIX client workstations.

In addition to the clients listed above, VM, MVS, OS/400, and any other DRDA clients are supported through the DRDA Application Server feature. For more information, see the *Installation and Operation Guide*.

Accessing Data Through DB2 Client Application Enabler

Through the DB2 Client Application Enabler, you can access data from multiple local or remote databases.

Accessing Data From Remote Clients: The DB2 Server can be accessed by local and remote clients. Remote clients must have the DB2 Client Application Enabler component installed to access a database server. Figure 1 shows an example of the DB2 server environment.
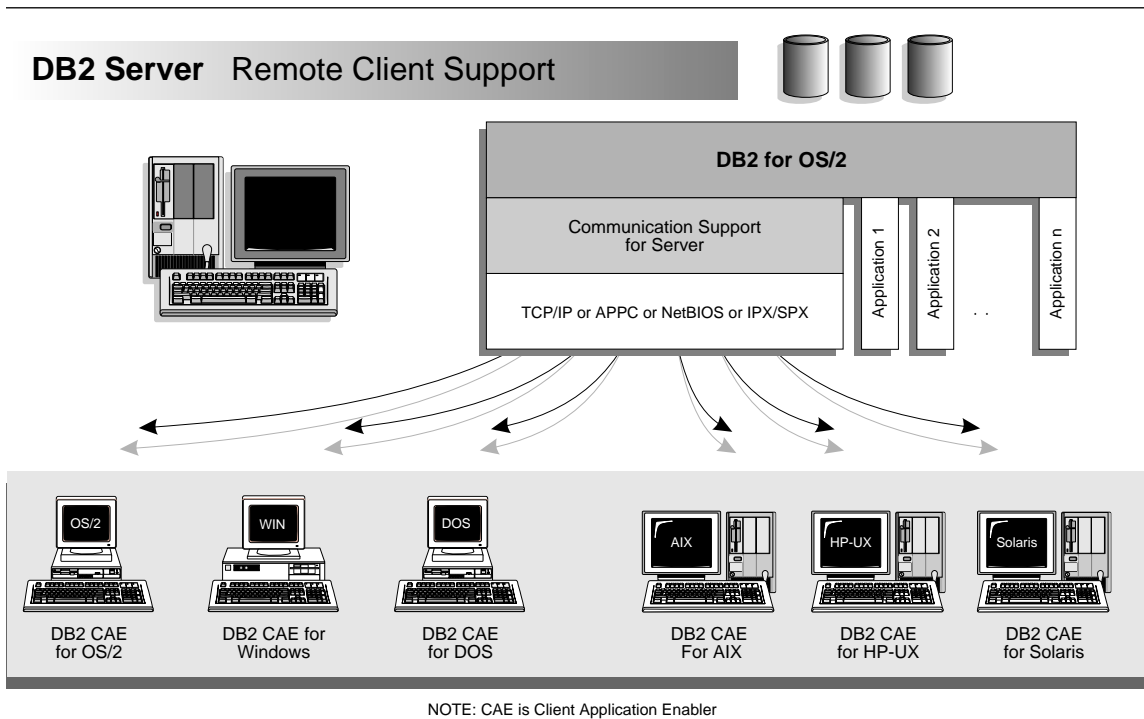
Figure 1. IBM DB2 Server

Accessing Multiple Database Servers: No additional software is required to enable LAN-to-LAN connections between DB2 clients and DB2 servers. Suppose you have a DB2 server on an AIX LAN located in Montreal and a DB2 server on an OS/2 LAN located in Toronto. As long as there is a connection between the two LANs, clients on either network can access either server. See Figure 2.

Communicating between Common Servers
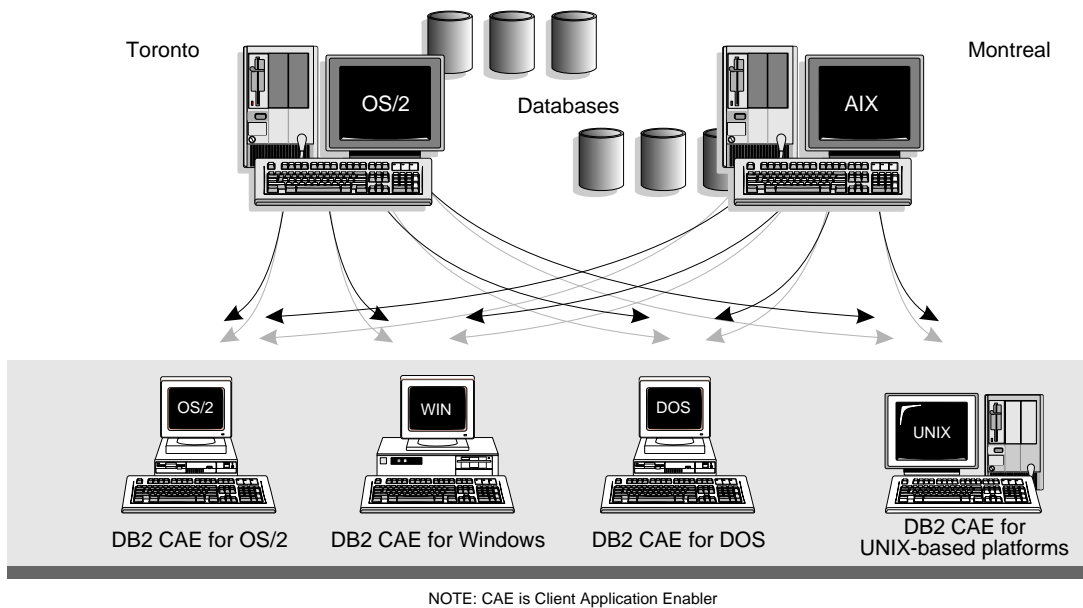
Figure 2. Accessing Data on Multiple Servers

Two-phase commit, or distributed unit of work access, is also supported. This means that within a single transaction, databases on both servers can be accessed and updated with full integrity. See the *DB2 Information and Concepts Guide* for more information on two-phase commits.

Administrator's Toolkit for OS/2 and AIX

Database administration capabilities can be performed from OS/2 and AIX workstations with the Administrator's Toolkit. The Administrator's Toolkit is designed to meet the needs of database administrators (DBAs). It is a collection of tools that helps manage and administer databases, including:

- The **Command Line Processor** to access and manipulate databases from the system command prompt
- The **Database Director** for configuration, backup and recovery, and directory management
- **Visual Explain** to graphically view and navigate complex SQL access plans
- The **Performance Monitor** to monitor the performance of your DB2 system for tuning purposes.

These tools listed are described in the following sections.

Managing Databases Using Database Director

The Database Director is an easy-to-use graphical interface that displays database objects (such as databases, tables, and packages) and their relationship to each other. Using the Database Director, you can select one or more database objects to perform the following tasks:

- Configure databases and database manager instances
- Manage the directories necessary for accessing local and remote databases
- Back up and recover databases or table spaces
- Manage table spaces
- Launch Visual Explain (see "Viewing SQL Access Plans Using Visual Explain" for information)
- Launch Performance Monitor (see "Monitoring Databases Using Performance Monitor" for information).

Chapter 6, "Setting Up Communications for the Client and Server" on page 45 shows step-by-step how to use Database Director to manage your databases and directories. Additional information can also be found in the *Getting Started* help panels.

Viewing SQL Access Plans Using Visual Explain

Visual Explain is a tool for analyzing and tuning SQL statements. This tool helps database administrators and application developers to:

- View the access plan chosen by the database manager's optimizer for a given SQL statement
- View the details of the access plan including statistics in the system catalogs
- Determine the source of problems in SQL statements.

"Using Visual Explain" on page 123 provides an example of using Visual Explain to analyze an SQL query. More information about Visual Explain can be found in the *Getting Started* help panels.

Monitoring Databases Using Performance Monitor

The Performance Monitor is a tool to help you monitor the performance of your DB2 system for tuning purposes. With the Performance Monitor, you can:

- Determine and analyze performance problems in the database manager or database applications
- Tune SQL statements for better performance
- Identify exception conditions based on thresholds you define
- Define your own statistics, in addition to the default set provided.

You can choose to monitor snapshots or events. Snapshot monitoring allows you to capture point-in-time information at specified intervals. Event monitoring allows you to record performance information over the duration of an event such as a connection.

"Using Performance Monitor" on page 124 provides examples of using Performance Monitor to analyze performance statistics. More information about Performance Monitor can be found in the *Getting Started* help panels.

Developing Applications Using the DB2 Software Developer's Kit

DB2 Software Developer's Kit can be installed with the DB2 Server or on a separate machine to provide an application development environment from both server and client platforms. When installed on the server, it provide access to both remote and local databases. When installed on the client, it provides access to remote databases.

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

There is a platform-specific version of the DB2 Software Developer's Kit available for each of the supported client and server environments. Applications developed with it will run on any client platform where the equivalent DB2 Client Application Enabler component is installed, and can access all DB2 servers as well as any other database server that implements the DRDA protocol.

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

- Embedded SQL
- Call level interface development environment (compatible with Microsoft's ODBC)
- Application programming interfaces (APIs) to access database utilities
- A prototyping environment using the command line processor's interactive SQL.

Programming libraries, header files, code samples, and online documentation are provided for developing applications with embedded SQL and the DB2 Call Level Interface (DB2 CLI). Several programming languages (including COBOL, C, and C++) are supported for application development, and precompilers for the supported languages are provided.

See the appropriate *DB2 SDK Building Your Applications* for your platform for complete information on using the DB2 Software Developer's Kit, or for a list of the supported compilers.

Connecting to the Internet Using DB2 World Wide Web Connection

The fast-growing popularity of the Internet and the World Wide Web has created a demand for Web access to enterprise data. DB2 World Wide Web Connection is a Web server gateway to IBM's DB2 family. It enables an application developer to build Web applications that access DB2 databases by using HTML forms and dynamic SQL. These applications are then stored on the Web server, and the end user sees only the HTML forms and the reports returning DB2 data.

A DB2 application built with DB2 WWW Connection displays a typical Web page form (an HTML document) to the end user who then can select values from a list panel or type in values to define the query. The user then clicks on a pushbutton to submit the query to the DB2 WWW Connection runtime engine on the Web server. At the Web server, the complete SQL statement is dynamically built with the user inputs and then sent to the database. If user input is not required for the query, a link in the HTML document can trigger a pre-written SQL query and display the resulting report. This type of automated query could be used for repeated access to current DB2 data.

A live demonstration of several DB2 WWW Connection applications is available on the Internet at URL `http://www.software.ibm.com`. (This address brings you to the IBM Software Page. From here, select the Data and DB2 links to find the DB2 WWW information.) The applications for this demo allow customers to query the CelDial Company database for the status of their accounts, to update account information, and to retrieve information about pending orders.

DB2 World Wide Web Connection can access databases locally if it is installed with DB2 Server and IBM Internet Connection Server. To run DB2 World Wide Web Connection as a remote client to DB2 Server, you need to install DB2 World Wide Web Connection and IBM Internet Connection Server on a client workstation. See Figure 3 for an example of this scenario.

DB2 World Wide Web Connection

DB2 for MVS,
DB2 for VSE, or
DB2 for VM

World Wide Web
(Internet)

UNIX

Web Browser

DB2 for OS/400

OS/2

WIN

Web Browser

Web Browser

DB2 World Wide Web Gateway

Internet Server,
DB2 Client Application Enabler,
DB2 World Wide Web Connection

DB2 Server
DDCS Gateway
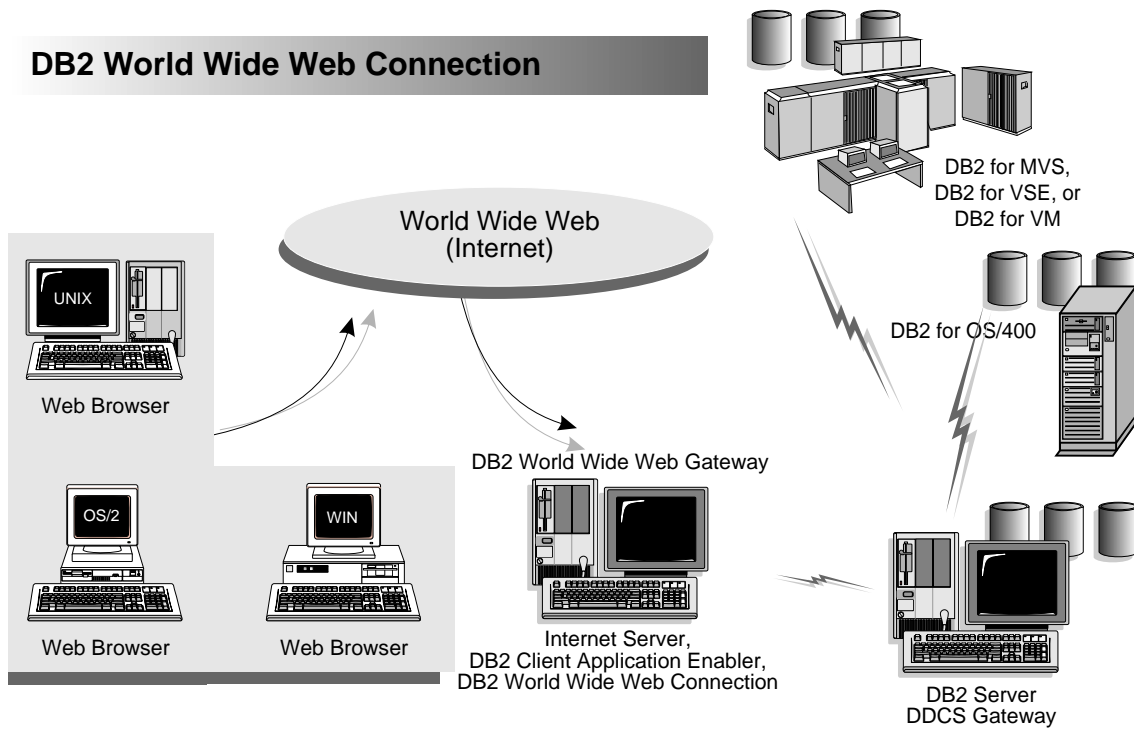
Figure 3. IBM DB2 World Wide Web Connection

If access to MVS, VM, VSE, or OS/400 servers is needed, DDCS is required on the
gateway machine.

Accessing Host Data from the Desktop Using DDCS

DDCS gives clients on your LAN access to data stored in host systems. A great deal of organizational data is managed by DB2 on systems such as DB2 for MVS/ESA, DB2 for VSE and VM, and DB2 for OS/400. Applications running on any of the supported client platforms can work with this data transparently as if it were managed by a local database server.

Distributed Database Connection Services for OS/2 (DDCS) provides applications with transparent online access to data by implementing a standard architecture for managing distributed data. This standard is known as Distributed Relational Database Architecture (DRDA). For example, you can use DDCS with:

Spreadsheets, such as Lotus 1-2-3 and Microsoft Excel, to analyze real-time data without the cost and complexity of data extract and import procedures

Decision support tools, such as IBM Visualizer, 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 IBM VisualAge, PowerBuilder, Microsoft VisualBasic, WATCOM VX-REXX, and VisPro/REXX, to create client/server solutions.

DDCS Multi-User Gateway provides the ability for multiple clients to connect to host data. By concentrating definitions and configurations for host access, this configuration can significantly reduce the effort required for implementing and administering access to enterprise data. See Figure 4.

DDCS Multi-User Gateway

application server

DB2 for
MVS

DB2 for
VSE

DB2 for
VM

S/390, S/370

DRDA   **APPC**

DB2 for
OS/400

AS/400

APPC   **DRDA**

application requesters

OS/2

APPC

DDCS

Communication Support

APPC, IPX/SPX,
NetBIOS, or TCP/IP

Application 1

Application 2

. . .

Application n

OS/2

WIN

DOS

UNIX

DB2 CAE for OS/2   DB2 CAE for Windows   DB2 CAE for DOS   DB2 CAE for
UNIX-based platforms

NOTE: CAE is Client Application Enabler

Figure 4. IBM DDCS Multi-User Gateway Version

DDCS allows DB2 systems on the host to assume the role of an enterprise super
server. This solution allows for established data management procedures to be
maintained, while applications are distributed to PC and workstation platforms to take
advantage of graphical user interfaces, distributed processing power, and excellent
development tools.

Part 2. Planning for and Installing Database Server

Chapter 3. Planning

Go directly to if you want to get started installing Database Server right away!

Database Server provides many components that you might want to use in your environment. This chapter provides product and planning information for Database Server.

Typical System Configurations

This section shows diagrams of the three main configurations that make use of the Database Server components.

Each diagram shows the software components that are required for each system in the network (excluding operating system and communication requirements). The diagrams also recommend the total amount of memory that should be installed on each system assuming a large database and many concurrent clients.

You can find more detailed working-set memory requirements and disk requirements in

Configuration for Accessing LAN Data

This diagram shows a configuration with a database on a LAN. The database server is installed on one machine and various clients connect to this server to access data in the database. Each client must have DB2 Client Application Enabler for its platform installed. Clients can connect to DB2 Server using one of the supported protocols: APPC, NetBIOS, IPX/SPX, or TCP/IP.

DB2 Server

LAN with various clients

UNIX

CAE for UNIX

32 MB

DB2 for OS/2

OS/2

CAE for OS/2

WIN

CAE for Windows
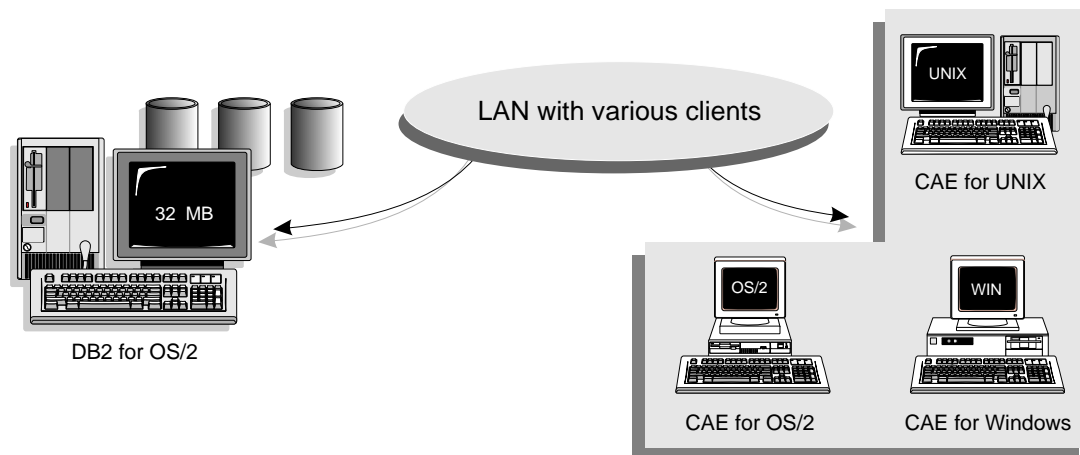
The size of machine you need for your server depends on the intended size of your network and databases. This diagram presumes a large database with approximately 32 concurrent users.

Configuration for Accessing Host Data

This diagram shows a configuration with a database on a host system. DDCS is installed on the gateway machine with the appropriate communications components. Clients can connect to DDCS for OS/2 using one of the supported protocols: APPC, NetBIOS, IPX/SPX, or TCP/IP. DDCS then connects to the appropriate host database management system using DRDA and APPC.

DDCS Gateway

LAN with various clients

UNIX

CAE for UNIX

OS/2

CAE for OS/2

WIN

CAE for Windows

16 MB

DDCS for OS/2

DB2 for MVS,
DB2 for VSE, or
DB2 for VM

DB2 for OS/400
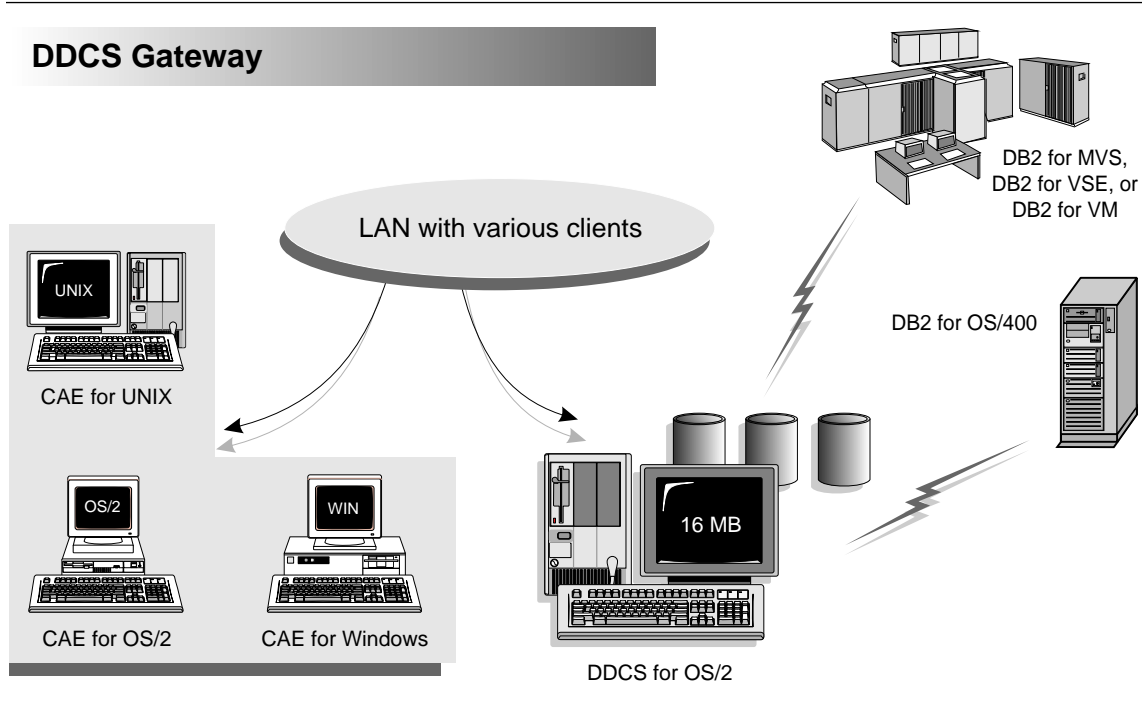
The size of a DDCS gateway machine depends on the number of clients you intend to connect. This diagram presumes a network with approximately 16 concurrent users.

Configuration for Accessing LAN and Host Data

This diagram shows a configuration where you can have databases on both your local LAN and on host management systems. You will need to install both DDCS and DB2 Server. These components do not need to be installed on the same physical machine, but this is one of the likely combinations. Clients can connect to DB2 Server and DDCS for OS/2 using one of the supported protocols: APPC, NetBIOS, IPX/SPX, or TCP/IP. If the request is intended for a host system, DDCS then connects to the appropriate host database management system using DRDA and APPC.

DB2 with DDCS Gateway

DB2 for MVS, DB2 for VSE, or DB2 for VM

DB2 for OS/400

LAN with various clients

UNIX

CAE for UNIX

OS/2

CAE for OS/2

WIN

CAE for Windows

32 MB

DDCS for OS/2
DB2 for OS/2
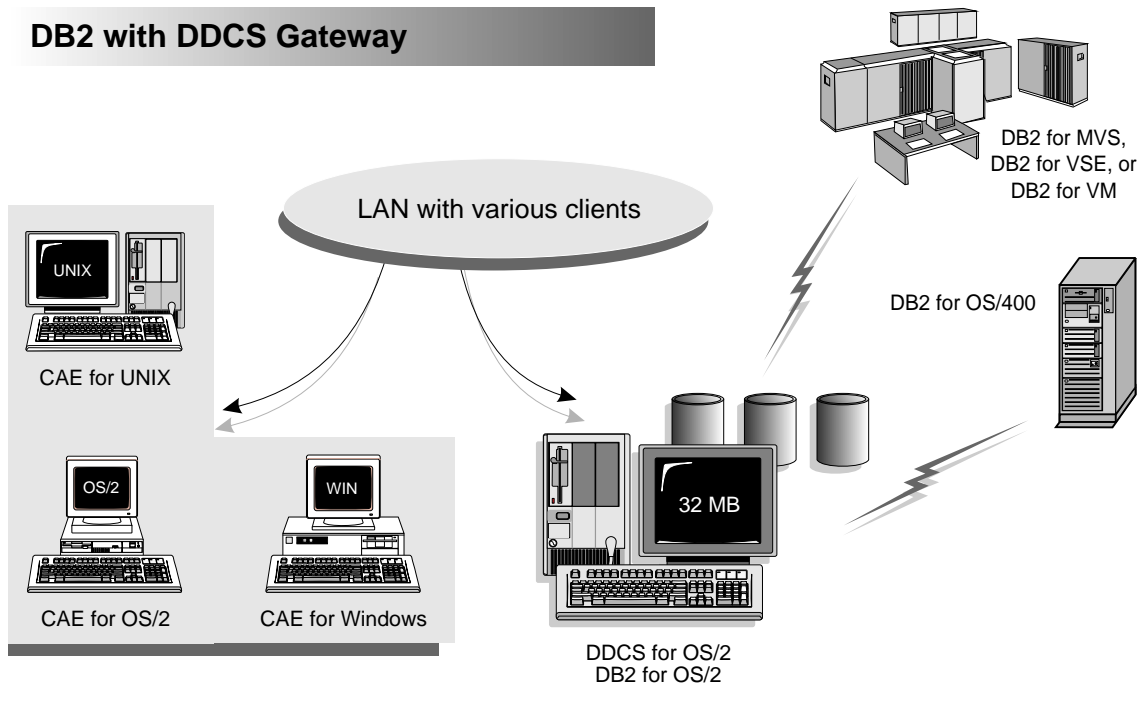
The size of machine with DDCS and DB2 Server installed depends on the size of your network and the intended size of your database. This diagram presumes a network with approximately 16 concurrent users accessing both the databases on DB2 Server and host systems.

Hardware Requirements

The diagrams in Chapter 3, "Planning" on page 19 show some of the many possible configurations with a recommendation for the total amount of memory required to run the installed software. This section shows the actual memory usage of each component that you can install. This is useful to help you plan for more sophisticated systems with a large number of concurrent clients or databases.

To install the Database Server components, you require a system unit capable of reading a CD-ROM. The system must also support the appropriate version of the operating system. See "Software Requirements" on page 27 for details.

The requirements necessary for the operating system, application development tools, and communications products are not included in the amounts listed. Consult the individual product documentation for these amounts. The amount of disk space and memory listed here are averages. The actual amounts that are required depend on the functions you are using.

Memory and Fixed Disk Estimating Worksheets

You can use the following worksheet to estimate the hardware capacity requirements for a particular Database Server configuration.

To estimate the memory and disk requirements for a particular configuration, add the recommended memory size for each installed function and application that is to be run concurrently. Include an allowance for user application data and buffer requirements.

The memory requirement is less if the user does not run all installed functions and applications concurrently. (System memory requirements are based only on applications and operating system functions that run concurrently.)

Using less than the calculated recommended memory size can increase the time required to switch from one function or application to another, reduce keystroke responsiveness, reduce throughput, and increase the size of the virtual memory paging space required.

The numbers shown in Table 1 on page 24 give the recommended memory size in megabytes (MB) for concurrent operation of various functions, and the recommended disk size in megabytes (MB).

Worksheet for Server Components

Use Table 1 to estimate the amount of memory and disk space you need on your server systems.

Table 1 (Page 1 of 2). Memory and Disk Estimating Worksheet for Server Components

Functions	Recommended Working Set Memory (MB)[a]	Recommended Disk (MB)
DB2 Server		
DB2 for OS/2 Server (base)[b]	2.9	26.0
For each local active user or application	0.3	
For each remote client	0.1	
For each active database[c]	1.6	5.5
For each active instance using NetBIOS	1.0	
For each active instance using any other protocol	0.3	
For each active instance	0.2	
Database Director		12.0
Performance Monitor		6.0
Visualizer Flight	2.0	3.0
Windows Support		8.0
Documentation		8.5
Japanese Conversion Tables		2.0
Korean Conversion Tables		0.6
Simplified Chinese Conversion Tables		0.4
Traditional Chinese Conversion Tables		3.0
DB2 Software Developer's Kit (Application Development Environment)		
DB2 Software Developer's Kit	0.7	20.0
For each client application	0.3	
Database Director		12.0
Visualizer Flight	2.0	3.0
Documentation		7.0
DB2 World Wide Web Connection		
IBM DB2 World Wide Web Connection	0.1	4.0
Documentation		1.0
DDCS (Access to Host Databases)		
DDCS for OS/2 (base)	0.8	16.0
For each active connection from remote clients	0.3	

Table 1 (Page 2 of 2). Memory and Disk Estimating Worksheet for Server Components

Functions	Recommended Working Set Memory (MB)[a]	Recommended Disk (MB)
For each active connection from local clients	0.7	
Database Director		6.0
Visualizer Flight	2.0	3.0
Windows Support		8.0
DRDA Syncpoint Manager		22.0
Documentation		4.0
Japanese Conversion Tables		2.0
Korean Conversion Tables		0.6
Simplified Chinese Conversion Tables		0.4
Traditional Chinese Conversion Tables		3.0

a Additional memory for some database configuration parameters, including the database buffer pool or the sort heap, might be required depending on the user's workload.

Working set memory includes the parts of a program's executable code and/or data areas that are being used intensively. Working set memory figures assume default values are used.

b Assumes one remote process using the local database, on one instance.

If only local clients are accessing the database, the base amount of memory required is 2.9 MB.

The base memory requirements for DB2 for OS/2 is 9 MB, but the active (working set) memory is 2.9 MB (as documented in the table). The difference (9 MB – 2.9 MB) is paged out to the swapper file on OS/2.

c The extra disk space for each additional database is for table definitions and internal structures for each database and does not include user data. It is subject to many variables. Actual requirements might differ.

The default configuration allocates 3 log files, each 800 KB in size. These logs can be relocated to another filesystem, and can be made larger or smaller as required. See the *Administration Guide* for more information.

Worksheet for Client Components

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

Table 2. Memory and Disk Estimating Worksheet for Client Components

Functions	Recommended Working Set Memory (MB)	Recommended Disk (MB)[b]
DB2 Client Application Enabler[a] (Remote Client)		
DB2 Client Application Enabler for OS/2 (base)	0.7	14.0
For each client application	0.3	
Visualizer Flight	2.0	3.0
Documentation		3.0
DB2 Client Application Enabler[a] (Remote Client)		
DB2 Client Application Enabler for Windows (base)	0.7	14.0
For each client application	0.3	
Documentation		3.0
Administrator's Toolkit		
Administrator's Toolkit (base)	0.7	32.0
For each client application	0.3	
Visualizer Flight	2.0	3.0
Documentation		4.0

a   The memory and disk estimates listed here are for the DB2 Client Application Enabler for OS/2 component which is meant to be copied to remote workstations on the network.

For memory and disk requirements for the Windows, Macintosh, AIX, DOS, HP-UX, and Solaris client products, see the *Installing and Using Clients* that are available on the Client Pack CD-ROM.

b   An additional 1.6 MB is required on the boot drive for UPM and FFST files if they do not already exist on your system.

Software Requirements

This section outlines the software that is required to run the Database Server components. Earlier versions of Warp, LAN Server, and Communications Manager might be supported. See the *DB2 Planning Guide* for information on any of these earlier versions.

Server Components

Table 3 lists the operating system and communications software required for each of the server components.

Table 3. Software Requirements

Component	Operating System	Communications	Notes
DB2 Server (Database Server)			
DB2 Server	OS/2 Warp Server	APPC, NetBIOS, TCP/IP, or IPX/SPX	• APPC is provided with IBM Communications Server, the other protocols are provided with OS/2 Warp Server. • The DB2 SNMP Subagent requires DPI 2.0 support provided by the OS/2 Warp Server. • Installing clients and servers using SystemView requires *SystemView LAN Client* provided by the OS/2 Warp Server.
DB2 Software Developer's Kit (Application Development Environment)			
DB2 SDK for OS/2	OS/2 Warp Server	APPC, NetBIOS, IPX/SPX, or TCP/IP	• APPC is provided with IBM Communications Server, the other protocols are provided with OS/2 Warp Server.
DB2 World Wide Web Connection (World Wide Web Access)			
DB2 World Wide Web Connection	OS/2 Warp Server	TCP/IP, IPX/SPX, APPC, or NetBIOS	• Also requires a Web Server such as IBM's Internet Connection Server.
DDCS (Access to Host Databases)			
DDCS for OS/2	OS/2 Warp Server	APPC, NetBIOS, TCP/IP, or IPX/SPX for inbound client access.	• APPC support is provided by IBM Communications Server. APPC is required to access DRDA Application Servers. • NetBIOS, TCP/IP, and IPX/SPX are provided by OS/2 Warp Server.

Client Components

Table 4 lists the requirements needed for the client components.

Table 4. Software Requirements

Component	Operating System	Communications	Notes
DATABASE 2 Client Application Enabler (Remote Client for OS/2)			
DB2 Client Application Enabler for OS/2	The client component of OS/2 Warp Server	APPC, NetBIOS, IPX/SPX, or TCP/IP	• The operating system and communications software required for OS/2 clients is provided by IBM Warp Server. • Installing clients and servers using SystemView requires *SystemView LAN client* provided by the OS/2 Warp Server. • APPC is provided with IBM Communications Server, the other protocols are provided with OS/2 Warp Server.
Administrator's Toolkit (Graphical Database Management Tools)			
Administrator's Toolkit	The client component of OS/2 Warp Server	APPC, NetBIOS, IPX/SPX, or TCP/IP	• See notes for DB2 Client Application Enabler for OS/2.
Client Pack (All other remote clients.)			
DB2 Client Application Enabler	Available for Windows 3.1, Windows 95, Windows NT, AIX, Macintosh, HP-UX, and Solaris.	Platform dependent	• See the CD-ROM booklet and the softcopy *Installing and Using Clients* for software requirements for each platform.

License Information

When you purchase Database Server Version 4, you receive a permanent license to use the following products:

- **DB2 Server for OS/2**
- **DB2 Software Developer's Kit for OS/2**
- **IBM DB2 World Wide Web Connection**

In addition, you get to try the **DDCS Multi-User Gateway** for 60 days. Once the 60 days are past, you need to purchase a permanent license and apply the license key to continue to use this product, otherwise it becomes inactive.

You are entitled to one copy of the DB2 Software Developer's Kit for OS/2. You can choose to install it on the server or on a client workstation. Refer to the *License Information* booklet to ensure that you are properly licensed for the components you are using. This booklet is shipped with the product.

Chapter 4. Installing Servers

The following table shows the products and selectable components that you can install on a server. Note that the process to install the client products and components is separate from the installation of the servers. Information is found in Chapter 5, "Installing Clients" on page 39.

Table 5. Products and Components for Database Server

Major Component	Selectable Subcomponents
DB2 Server	
	• Server • Database Director • Performance Monitor • Visualizer Flight • Windows Support • Documentation • Japanese Conversion Tables • Korean Conversion Tables • Simplified Chinese Conversion Tables • Traditional Chinese Conversion Tables
DB2 Software Developer's Kit	
	• DB2 Software Developer's Kit for OS/2 • Database Director • Visualizer Flight • Documentation
DB2 World Wide Web Connection	
	• DB2 World Wide Web Connection • Sample Application • Documentation
DDCS	
	• DDCS Multi-User Gateway • Database Director • Visualizer Flight • Windows Support • DRDA Syncpoint Manager • Documentation • Japanese Conversion Tables • Korean Conversion Tables • Simplified Chinese Conversion Tables • Traditional Chinese Conversion Tables

Database Server Installation

Database Server installation allows you to explicitly choose which products you want to install, update, or delete.

To run the Database Server Installation, perform the following steps:

1 Insert the CD-ROM in the drive.

2 There are two ways to begin the Database Server installation. These are:

- If you have the OS/2 Warp Server installed, double-click on the **Server Installation** icon located in the Server folder on the desktop.

OR

- Open an OS/2 window and type `X:\<LANGUAGE>\INSTALL`.

    Where `X:` is the CD-ROM drive and `<LANGUAGE>` is the language directory from which you want to install the product files. For example, the English files are located in the `X:\EN` directory. See the CD-ROM booklet for a listing of all the language directories.

3 The first install window appears, as follows:

```
┌─────────────────────────────────────────────────────────┐
│ ⌄  IBM Database Server for OS/2 - Installation           │
├─────────────────────────────────────────────────────────┤
│                                                          │
│  ┌─IBM Database Server Products──────────────────────┐   │
│  │ ☑ DB2 - Server                          64 MB     │   │
│  │ ☑ Software Developer's Kit              41 MB     │   │
│  │ ☑ DB2 World Wide Web Connection         18 MB     │   │
│  └───────────────────────────────────────────────────┘   │
│                                                          │
│  ┌─IBM DDCS Product─────────────────────────────────┐   │
│  │ ☐ DDCS Multi-User Gateway *             51 MB     │   │
│  └───────────────────────────────────────────────────┘   │
│                                                          │
│  ┌─Operation Type───────────────────────────────────┐   │
│  │ ● Install       ○ Update         ○ Delete         │   │
│  └───────────────────────────────────────────────────┘   │
│                                                          │
│  * - This product will be installed "Try and Buy".       │
│                                                          │
│                                                          │
│   ┌──────────┐    ┌──────────┐    ┌──────────┐          │
│   │ Continue │    │   Exit   │    │   Help   │          │
│   └──────────┘    └──────────┘    └──────────┘          │
└─────────────────────────────────────────────────────────┘
```

a Click on the product you wish to install, update, or delete.

b Click on the radio button that corresponds to the action you wish to perform: install, update, or delete.

c Click on **Continue** to proceed.

> Steps 4 and 5 are repeated for each of the components selected in the first window.

4 The next window asks you whether to update the CONFIG.SYS file as part of the installation. Here is the window that appears:

```
┌──────────────────────────────────────────────┐
│ ⌄  Install                                     │
│              IBM DB2 for OS/2 - Server         │
│ Product number:  5622-593                      │
│ Version:         02.01.01                      │
│ Feature:         0601                          │
│ ┌Options──────────────────────────────────┐   │
│ │                                          │   │
│ │ ☑ Update CONFIG.SYS                      │   │
│ │                                          │   │
│ └──────────────────────────────────────────┘   │
│ Install this product?                          │
│ ┌────────┐   ┌────────┐   ┌────────┐           │
│ │   OK   │   │ Cancel │   │  Help  │           │
│ └────────┘   └────────┘   └────────┘           │
└──────────────────────────────────────────────┘
```
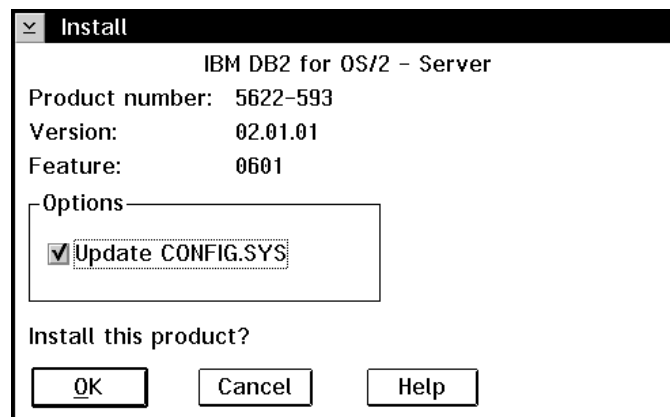
Click on **OK** to have the CONFIG.SYS file updated. A backup copy of your existing file is kept.

To have the changes made to the CONFIG.ADD file, click on **Update CONFIG.SYS** to de-select it and then, click on **OK**. You will need to manually add these values to your CONFIG.SYS file.

5 The Install - directories window provides a list of the product's components. This gives you the opportunity to decide which of the components are needed on your system. Here is the window that appears for the Database Server components:
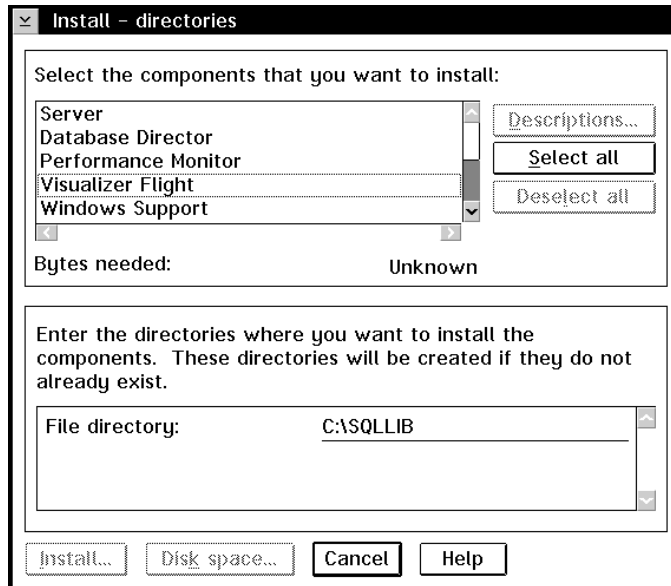
For the Install - directories window, complete the entries as follows:

a Click on each of the components you want to install.

b Click on **Descriptions** for a brief description of the selected component.

c Enter the **File directory** where you want the products installed. The default is C:\SQLLIB.

d Click on **Disk Space** to see how much space you have on each of the drives on your machine.

e Click on **Install** to continue.

f Enter your **Product License Key** to have permanent access to DB2 Software Developer's Kit or DB2 Server. You can find the key on a label shipped with the product. The product license key can also be entered manually following the installation. See "Entering License Keys" on page 35 for further information. Click on **OK** to continue.

The products you have chosen are installed on your system. A progress window appears until the installation is complete.

6 While the product is installing, an electronic registration tool is invoked to allow you to register as a licensed user of DB2 Server. Fill in the panels as directed on the screen and send the registration to IBM. A registration number will be sent to you to you in order to complete the registration. See Appendix C, "Product Registration" on page 157 for more detailed information.

7 Once you have completed the installation of all the products you want on this system, shutdown and reboot your system:

    a Click on **Exit** to complete the installation.

    b Click on **Shutdown** from the LaunchPad Palette.

    c Wait for the `Shutdown complete` message.

    d Press the Ctrl, Alt, Delete keys simultaneously to restart your system.

> Before you reboot, consider completing the "Setting the Environment Variable DB2COMM on the Server" on page 46 step to avoid the need to reboot a second time.

This completes the installation. When your system restarts, a DB2 application called *First Steps* executes automatically (the first time only) to let you log on and create the sample database. This application is also available from the IBM DATABASE 2 folder. See Chapter 8, "Getting Started with the Database Server" on page 91 for more information.

To install the client components, proceed to Chapter 5, "Installing Clients" on page 39. To configure your server to accept requests from remote clients, proceed to Chapter 6, "Setting Up Communications for the Client and Server" on page 45. To create the sample database to verify the installation, proceed to "Creating the Sample Database" on page 95.

Entering License Keys

Before you can use the DB2 Server or DB2 Software Developer's Kit products, you must enter their pre-generated license keys. The Database Server is packaged with a license key for each of them; ensure that you enter the key appropriate for the product you are installing. You do not require a license key for the DB2 Client Application Enabler or Administrator's Toolkit. Refer to "License Information" on page 29 for the Database Server licensing information.

1 If you do not enter the license keys in the nodelock file, the products will become inactive after 60 days.

2 To use DDCS permanently, you need to purchase a DDCS license which includes a license key. Use the following steps to apply it.

Product License Key: The format of a license key is as follows:

P/N: The product part number
Prd: The product name
S/N: The label serial number
Key: The license key

Enter the license keys in the following nodelock file on the machine where the product is installed:

```
\SQLLIB\NODELOCK
```

If this file does not exist, then create it.

Entering the Key During Installation: You are prompted for a license key during installation of DB2 Server and DB2 Software Developer's Kit. The license key can be found on a label inside the product box. If you enter the key at installation time, an entry is automatically made in the nodelock file.

Entering the Key After Installation: To enter the license keys, edit the nodelock file using any text editor, and follow these steps for each key:

1 Add a comment line indicating what the key is used for. Comments are preceded by the # character. This step is recommended, but optional.

2 On one line, type the vendor id, followed by a space, followed by a license key.

The DB2 vendor ID is: "`5fbee0ee6feb.02.09.15.0f.48.00.00.00`".

For example, the license key entry for both DB2 for OS/2 and DB2 Software Developer's Kit might look like:

```
# DB2 for OS/2 Version 2.1 License Key (S/N: ssssssss)
5fbee0ee6feb.02.09.15.0f.48.00.00.00 kkkkkkkkkkkkkkkkkkkkkkkk
# DB2 Software Developer's Kit Version 2.1 License Key (S/N: ssssssss)
5fbee0ee6feb.02.09.15.0f.48.00.00.00 kkkkkkkkkkkkkkkkkkkkkkkk
```

where
"`ssssssss`" is the serial number on the label,
"`5fbee0ee6feb.02.09.15.0f.48.00.00.00`" is the vendor id, and
"`kkkkkkkkkkkkkkkkkkkkkkkk`" is the license key on the label.

Installing Servers Remotely

CID Response Files: You can install the server components through the Configuration, Installation, and Distribution (CID) response files. Use the instructions for the CID process in "Installing Clients Remotely" on page 41 with the following changes.

- The products you are able to install include:

  - IBM DB2 for OS/2 - Server
  - IBM DB2 Software Developer's Kit for OS/2
  - IBM DB2 World Wide Web Connection for OS/2
  - IBM DDCS Multi-User Gateway for OS/2

  Use one of these product names with the `/P` parameter.

- The files are located in the <LANGUAGE> directory on the CD-ROM. For example, the English files are located in the `X:\EN` directory.

- The command files available to simplify the CID process are found in the X:\<LANGUAGE> directory. The command files available include:

  ```
  DB2SRVR.CMD        (IBM DB2 Server)
  DB2SDK.CMD         (IBM DB2 Software Developer's Kit)
  DB2WWW.CMD         (IBM DB2 World Wide Web Connection)
  DDCSMU.CMD         (IBM DDCS Multi-User Gateway)
  ```

- An additional keyword is available for installing remote servers. You can suppress the automatic product registration with the keyword `DB2NOREG=YES`. See Appendix C, "Product Registration" on page 157 for further information.

SystemView: You can install the server components through the SystemView support included in OS/2 Warp Server. SystemView allows you to distribute software to other workstations electronically. This type of installation is referred to as a remote, unattended installation. You need to have a network connection established between the code server and the target workstation. You also need to have the SystemView client installed on the target workstation. Refer to Appendix B, "Remote CID Installation Using SystemView" on page 153 for instructions on how to use SystemView.

Chapter 5. Installing Clients

Clients on the LAN that will connect to the DB2 Server must have the DB2 Client Application Enabler or Administrator's Toolkit installed. DB2 Client Application Enablers are available for several operating system platforms, including DOS, Windows 3.1, 95, NT, UNIX, Macintosh and OS/2. Administrator's Toolkits are available for OS/2 and AIX clients. All available DB2 Client Application Enablers and Administrator's Toolkits are packaged on the *Client Pack*. You are able to copy these components to the appropriate platform on any number of workstations.

As an example, the following table shows the list of components that you can install on remote OS/2 clients. It shows the DB2 component name and the subcomponents that you are able to select.

Major Component	Selectable Subcomponents
DB2 Client Application Enabler	
	Client Application Enabler
	Visualizer Flight
	Documentation
DB2 Administrator's Toolkit	
	Administrator's Toolkit
	Visualizer Flight
	Documentation

The components available to install on the Windows platforms include the DB2 Client Application Enabler and documentation.

Use the following instructions to install and configure OS/2 and Windows clients. See the CD-ROM booklet and the appropriate manual for all other clients. See Chapter 12, "Obtaining Additional Information" on page 139 for details.

Installing OS/2 or Windows Clients

You can install an OS/2 or Windows client:

- Interactively from CD-ROM or from an OS/2 LAN redirected drive
- Unattended or remote using a response file.

The methods are described in the following sections.

Installing Clients from the CD-ROM

To run the client installation, perform the following steps:

1 Insert the *Client Pack* CD-ROM in the drive.

2 Type `X:\OS2\<LANGUAGE>\INSTALL` to begin the OS/2 installation program,
`X:\WINDOWS\<LANGUAGE>\CLIENT\INSTALL` to begin the Windows installation, and
`X:\WIN32\<LANGUAGE>\DISK1\SETUP` to begin the Windows 95 installation.

Where `X:` is the CD-ROM drive and `<LANGUAGE>` is the language directory from
which you want to install the product files. For example, the English files are
located in the `EN` directory. See the CD-ROM booklet for a listing of all the
language directories. (Specific installation instructions for all other platforms can
be found in the X:\<PLATFORM>\<LANGUAGE>\README file).

The first install window appears, as follows:
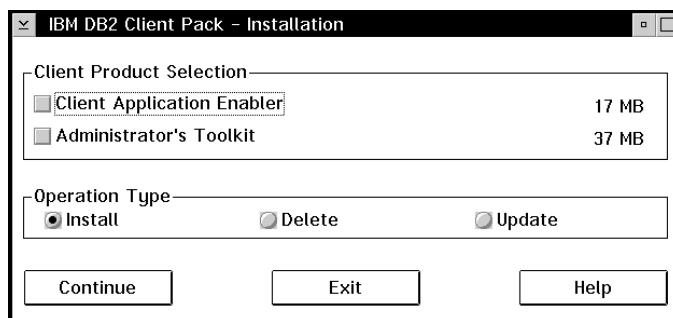
```
┌─────────────────────────────────────────────────────────┐
│ ⌄  IBM DB2 Client Pack – Installation              ▫ □  │
├─────────────────────────────────────────────────────────┤
│  ┌─Client Product Selection──────────────────────────┐  │
│  │  ▢ Client Application Enabler          17 MB      │  │
│  │  ▢ Administrator's Toolkit             37 MB      │  │
│  └───────────────────────────────────────────────────┘  │
│                                                          │
│  ┌─Operation Type────────────────────────────────────┐  │
│  │   ◉ Install        ○ Delete        ○ Update       │  │
│  └───────────────────────────────────────────────────┘  │
│                                                          │
│  ┌──────────────┐   ┌──────────────┐   ┌──────────────┐ │
│  │   Continue   │   │     Exit     │   │     Help     │ │
│  └──────────────┘   └──────────────┘   └──────────────┘ │
└─────────────────────────────────────────────────────────┘
```

a Click on the product you wish to install, update, or delete.

b Click on the radio button that corresponds to the action you wish to perform:
install, update, or delete.

c Click on **Continue** to proceed.

3 The rest of the installation instructions are the same as those given for the Server
Installation. Refer to Step 4 on page 33 for the remainder of the instructions.

To configure your clients to access remote servers, go to Chapter 6, "Setting Up Communications for the Client and Server" on page 45.

Installing Clients Remotely

CID Response Files: To install clients from a code server using a response file, you need to follow these steps.

The example shows the directory for an OS/2 client. For Windows 3.1 clients, the directory is `X:\WINDOWS\<LANGUAGE>\CLIENT`. See the CD-ROM booklet for information on Windows 95 or NT clients.

1 Copy the DB2 product files to your *code server*. Use XCOPY with the `/S` option to copy all the required files from the CD-ROM directories to the hard drive. For example:

```
XCOPY X:\OS2\EN\*.* E:\CLIENTS /S
```

2 Response files are used to install products remotely or to provide a customized install. The DB2RESP command is available for OS/2 clients and triggers the **Create Installation Response File** tool to help you create a response file with all the keywords necessary for a basic install. For Windows 3.1 clients, a sample response file called `CAEWIN.RSP` is provided. Customize this file with the necessary values.

Create a DB2 response file to be used with CID or by individual users by executing the DB2RESP command from the product directory on the CD-ROM.

```
X:\OS2\EN\DB2RESP
```

Use the prompts and the online help to complete the required information. The response file is a plain ASCII file that is made up of a list of keywords and keyword values. For example:

```
CFGUPDATE = AUTO
```

The complete list of keywords are inserted in the file as comment lines. To customize the response file, you can edit the resulting file using any text editor.

See the *Installation and Operation Guide* or *Installing and Using Clients* for a list of supported keywords.

3 Install DB2 with a response file by issuing the remote install command. To simplify the entering of the command, you can use one of the CMD files that contains an example of the command. Edit the file to make any changes you might need, and run the file.

The following CMD files are available in the X:\OS2\<LANGUAGE> directory:

```
DB2CAE.CMD          (IBM DB2 Client Application Enabler)
DB2ADMIN.CMD        (IBM DB2 Administrator's Kit)
```

The syntax for the remote installation command follows:

```
►►─┬──────────┬───install──/A:action──/P:product_name───────────►
   └─d:\path\─┘

►──/R:respfile──/X──┬────────────────────┬─────────────────────►
                    └─/L1:error_logfile──┘

►──┬──────────────────────┬───────────────────────────────────►◄
   └─/L2:history_logfile──┘
```
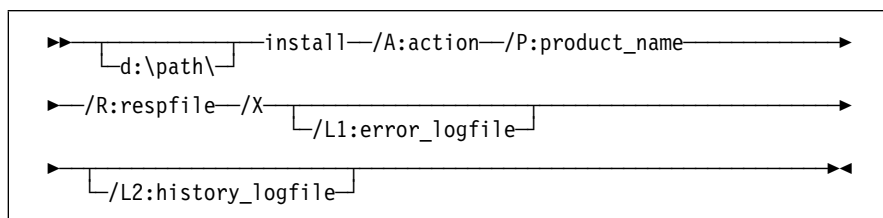
Specify the fully qualified path to invoke the program, followed by the required /R
parameter to identify the response file to be used.

For example:

```
X:\SQLLIB\EN\INSTALL /A:I /P:"IBM DB2 Client Application Enabler for OS/2"
/R:D:\resp.rsp /X /L1:D:\err.log
```

/A   (**Required** for unattended installation)

    D Delete
    I   Install
    R Restore
    U Update

/P   (**Only for OS/2**) Specifies the product to be installed. Enter the product
          name exactly as shown.

    • "IBM DB2 Client Application Enabler for OS/2"
    • "IBM DB2 Administrator's Toolkit for OS/2"

/R   (**Required**) Specifies the fully qualified response file name.

/X   (**Required**) Specifies unattended execution.

/L1   Specifies the error log file name.

          The error log identifies all errors that occur during the execution of the
          installation program, and records where in the program the errors
          occurred.

          Make sure the file name is fully qualified otherwise the file
          might be placed somewhere you may not expect.

/L2 Specifies the history log file name.

The history log provides a list of all the files that were processed by the installation program. The history log should be checked for each workstation after a CID install to ensure the desired components were handled properly.

Make sure the file name is fully qualified, otherwise the file might be placed somewhere you may not expect.

SystemView: Using the SystemView support included in OS/2 Warp Server, you can distribute OS/2 clients to other workstations electronically. This type of installation is referred to as a remote, unattended installation. You need to have a network connection established between the code server and the target workstation. You also need to have the SystemView client installed on the target workstation. Refer to Appendix B, "Remote CID Installation Using SystemView" on page 153 for instructions on how to use SystemView.

Chapter 6. Setting Up Communications for the Client and Server

If you do not have a database of your own to catalog, go to "Creating the Sample Database" on page 95 to create the sample database.

This chapter provides information on setting up the client and server workstations using any one of the following supported communication protocols:

- IPX/SPX - see "Configuring IPX/SPX" on page 48
- NetBIOS - see "Configuring NetBIOS" on page 54
- TCP/IP - see "Configuring TCP/IP" on page 59

APPC is also supported for LAN communications. See the *DB2 Installation and Operation Guide* and *DB2 Installing and Using Clients* for information.

Once communication protocols are set up and nodes and databases are cataloged, a connection between an application on the client workstation and a target database is established automatically when the application issues a CONNECT statement. The location of the target database (local or remote), and the communication protocol to be used (if remote) are transparent to the user application.

Client or Server Tasks: To achieve a connection between a database client and server, you must set up the communications at each workstation. The communication tasks listed here specify whether the task must be performed at the client, server, or both.

The tasks listed for the server need only be done once for each server on your LAN. The tasks listed for the clients need to be done for each client on the network requiring connection to the DB2 Server.

How to Set Up Communications: There are various ways to configure the communications for your servers and clients. This section documents the most common methods, as follows:

This is the command line processor icon. It represents actions that are to be performed with the command line processor. All Database Server components have the command line processor built-in.

This is the Database Director icon. This icon represents actions that are to be performed with the Database Director tool. DB2 Server, DB2 Software Developer's Kit, and Administrator's Toolkit components can use this tool.

This is the DB2 Client Setup icon. This icon represents actions that are to be performed with the DB2 Client Setup tool. DB2 Client Setup is available for all OS/2 and Windows 3.1 clients.

Setting the Environment Variable DB2COMM on the Server

DB2 Server or DDCS for OS/2 Multi-User Gateway can support multiple communication protocols concurrently; however, not all of them must be enabled. The DB2COMM environment variable controls which communication protocol is enabled when a DB2START command is issued.

You can set the DB2COMM environment variable permanently by placing a statement in your CONFIG.SYS file or temporarily for an individual session by entering a command.

To have a protocol initialized each time your system boots up, you must add a statement in the CONFIG.SYS. For example, if your server uses local Windows applications and accepts requests from TCP/IP, enter the following statement in your CONFIG.SYS file:

```
SET DB2COMM=NPIPE,TCPIP
```

Any combination of the following keywords, separated by commas, is valid for DB2COMM:

APPC   initialize APPC support

IPXSPX   initialize IPX/SPX support

NETBIOS initialize NetBIOS support

TCPIP   initialize TCP/IP support

NPIPE   initialize NAMEPIPE support for local Windows applications.

During installation, the `SET DB2COMM=NPIPE` statement is placed in your CONFIG.SYS file if you chose to install Windows support. If present, modify this statement to include the protocols you want to enable. If not, add a SET DB2COMM statement to enable the desired protocols.

Reboot your system after making changes to the CONFIG.SYS file.

To have a protocol initialized for an individual session, enter the `SET DB2COMM=TCPIP` command for the protocol you want to use, before starting DB2 (DB2START). The protocol selected, in this case TCP/IP, is in effect until the next DB2START command is issued. DB2COMM can be set and changed any time, and the new value is effective with the next DB2START command.

When DB2START is issued, the protocol specified at the DB2COMM environment variable is initialized. If there are problems initializing this protocol, a warning message is written to the screen and the specific error messages are logged in the SQLLIB\DB2\DB2DIAG.LOG file.

- If DB2COMM is undefined or set to null, no communication support is initialized when the database manager is started.

- The keywords for DB2COMM can be entered in upper or lower case.

- See the *Installation and Operation Guide* or *Installing and Using Clients* manual for more information on how to set DB2COMM.

Configuring IPX/SPX

For DB2 Client Application Enabler for OS/2 to access a remote database server through IPX/SPX, you must have installed and configured the required NetWare File and Print Gateway Service component of IBM Warp Server on both the client and server workstations. See the "Software Requirements" on page 27 for details.

See your IBM Warp Server documentation for detailed information on the installation and configuration of NetWare File and Print Gateway Service.

File Server Addressing and Direct Addressing: A client can access the server using the IPX/SPX protocol via *File Server Addressing* or *Direct Addressing*: the instructions here assume File Server Addressing. If you decide to use Direct Addressing, see the *Installation and Operation Guide* and *Installing and Using Clients* manuals.

The following steps include the configuration tasks necessary to set up IPX/SPX at the client and server:

1 Complete the IPX/SPX worksheet to determine the values you need to enter in the following steps.

2 Set the environment variable DB2COMM on the server

3 Update the database manager configuration file on the server

4 Register the DB2 Server on the NetWare file server on the server

5 Catalog the IPX/SPX node on the client

6 Catalog the database on the client.

Step 1. Assigning Values to the Parameters (Worksheet)

Complete the "Your Value" column in the following worksheet to understand the purpose of the parameters and to assign values of your own to the various parameters. By assigning these values, you will have all the information you need to perform the rest of the steps in this section.

Parameters	Our Example	Your Value	Explanation
IPX/SPX-specific Values			
DB2NODE	db2ipx1		Alias or nickname for the database node to be cataloged on the client. Choose a name that makes sense to you. This value must be unique within the node directory.
FILESERVER	netwsrv		The name of the NetWare file server where the database server is registered. This parameter must be entered in uppercase.
OBJECTNAME	db2inst1		The name of the database manager instance in an IPX/SPX network. This name must be unique for all database server instances registered at the same NetWare file server.
IPX_SOCKET	879F		Represents the connection end point in a DB2 Server's internetwork address. It must be unique for each DB2 server instance and unique among all IPX/SPX applications running on one DB2 Server workstation. You may select values in the range of 879E to 879A2. Do not use the value 0x0000 or any number in the range 0x5000 to 0x7FFF.

> The following characters are not valid for the fileserver or objectname parameters: / \ : ; , * ?

In the steps that follow, substitute your values with the ones provided in the examples.

Step 2. Setting DB2COMM on the Server

Add or update the environment variable `SET DB2COMM=NPIPE,IPXSPX` in your
CONFIG.SYS file and reboot your system. See "Setting the Environment Variable
DB2COMM on the Server" on page 46 for details.

Step 3. Updating the Database Manager Configuration File on the Server

The database manager configuration file at the server contains three parameters
(fileserver, objectname, and ipx_socket) that are specific for IPX/SPX support.

Ensure you are logged on as an administrator to the server that contains the database.
Then add the fileserver, objectname, and ipx_socket to the database manager
configuration file. This can be performed using either the Database Director or
command line processor, as follows:

Using the DB2 Database Director:

 1 Open the **IBM DATABASE 2** folder.
 2 Double-click on the **Database Director** icon. The Database Director window
    opens.
 3 Click on the database manager that you want to update from the main Database
    Director window. (The default is **DB2**.)
 4 Click on the **Selected** pulldown and choose **Configure**. The DB2 - Configure
    Notebook opens.
 5 Select the **Protocols** tab and advance to the second page of this section.
 6 Configure the IPX/SPX-related information as follows:
    fileserver   netwsrv
    objectname   db2inst1
    ipx_socket   879F
 7 Click on **OK**.
 8 Go back to the **IBM DATABASE 2** folder.
 9 Double-click on the **Stop DB2** icon to stop the database manager. Wait for the
    message `successful` before continuing.
10 Double-click on the **Start DB2** icon to make the configuration file changes
    effective.
11 Proceed to "Step 4. Registering the DB2 Server on the NetWare File Server on
    the Server" on page 51.

Using the Command Line Processor: Enter the following commands at the command line processor to update the database manager configuration file:

```
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING FILESERVER netwsrv
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING OBJECTNAME db2inst1
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING IPX_SOCKET 879F
DB2STOP
DB2START
```

To view the database manager configuration file, issue the following command:

```
DB2 GET DATABASE MANAGER CONFIGURATION
```

Step 4. Registering the DB2 Server on the NetWare File Server on the Server

Register the DB2 Server on the NetWare file server. You must do this after the database manager configuration has been updated with the IPX/SPX parameters. These commands are executed from the DB2 Server:

```
DB2 REGISTER DB2 SERVER IN NWBINDERY USER userid PASSWORD password
```

See the *Command Reference* for details on all commands.

1 You must specify the following in upper case: file server name, user ID, and password.

2 The user ID and password are used to log on to the NetWare file server and must have *Supervisor* or *Workgroup Manager* security equivalence.

3 If you are registering at a NetWare 4.x fileserver (which uses directory services and provides bindery emulation capability), the user ID used to REGISTER/DEREGISTER (besides having supervisory or equivalent authority) 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).

4 If you modify the IPX/SPX configuration parameters or change the DB2 Server network address, you will need to deregister and re-register your DB2 Server.

Step 5. Cataloging the IPX/SPX Node on the Client

You must catalog the db2node, fileserver, and objectname of the remote server node. This adds an entry in the client's node directory that points to the remote database server.

You can catalog an IPX/SPX node with the Database Director, the DB2 Client Setup, or by issuing the CATALOG IPXSPX NODE command on the client workstation. Ensure you are logged on as an administrator to perform this step.

> If you are having trouble logging on to the client, see "Logging On to the System" on page 93 for information that may help.

Using the DB2 Database Director:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window opens.

3 Click on the [+] for the **database manager** where the database resides. (The default is **DB2**.)

4 Click on the [+] for **Directories**.

5 Double-click on the **Node Directory**. The panels for the node directory open.

6 Click on the **Directory Entry** pulldown and choose **Catalog**.

7 Configure the IPX/SPX-related information as follows:

    a Fill in the field for **Node** as follows:
        Node   db2ipx1

        Then, click on the radio button for the IPX/SPX protocol.

    b Enter the information for **File server** and **Object name** as follows:
        File server   netwsrv
        Object name db2inst1

8 Click on **OK**.

9 Proceed to "Step 6. Cataloging the Database on the Server and the Client" on page 53.

Using the DB2 Client Setup:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **DB2 Client Setup** icon. The DB2 Client Setup window opens.

3 Select the **Node** menu.

4 Select the **New** menu item. The New Node window opens.

5 Select the **IPX/SPX** radio button.

6 Configure the IPX/SPX-related information as follows: the values can be:
   Name   db2ipx1
   File Server netwsrv
   Object Name db2inst1

7 Click on **OK**.

8 Proceed to "Step 6. Cataloging the Database on the Server and the Client" on page 53.

Using the CATALOG IPXSPX NODE Command: To catalog the remote database server using the command line processor, execute the following commands from your remote client:

```
DB2 CATALOG IPXSPX NODE db2ipx1 REMOTE netwsrv SERVER db2inst1
DB2 TERMINATE
```

When you catalog a database that resides in the server instance represented by this node, use the value db2ipx1 in your CATALOG DATABASE command.

Step 6. Cataloging the Database on the Server and the Client

Next, you must update the database directory on the client and server. See "Cataloging Databases" on page 66 for information.

Configuring NetBIOS

For DB2 Client Application Enabler for OS/2 to access a remote database server
through NetBIOS, you must first have installed and configured the Adapters and
Protocol Services components of IBM Warp Server on both the client and server
workstations. See the "Software Requirements" on page 27 for details.

See your IBM Warp Server documentation for detailed information on the installation
and configuration of Adapters and Protocol Services.

The following steps include the configuration tasks necessary to set up NetBIOS at the
client and server:

1 Complete the NetBIOS worksheet to determine the values you need to enter in
   the following steps.
2 Set the environment variable DB2COMM on the server
3 Update the database manager configuration file on the server
4 Update the database manager configuration file on the client
5 Catalog the NetBIOS node on the client
6 Catalog the database on the client

Step 1. Assigning Values to the Parameters (Worksheet)

Complete the "Your Value" column in the following worksheet to understand the
purpose of the parameters and to assign values of your own to the various parameters.
By assigning these values, you will have all the information you need to perform the
rest of the steps in this section.

Parameters	Our Example	Your Value	Explanation
NetBIOS-specific Values			
Node Name	• Client: db2netb1 • Server: N/A		Alias or nickname for the database node to be cataloged on the client. Choose a name that makes sense to you. This value does NOT have to the same as the NNAME of the client or server. This value must be unique within node directory.
NNAME	• Client: client1 • Server: db2inst1		The NNAME of the workstation is an arbitrary name known only to DB2. This parameter must be unique among all NetBIOS nodes in the network.

Parameters	Our Example	Your Value	Explanation
Adapter	• Client: 0 • Server: N/A		The adapter refers to the logical network adapter that is used for the NetBIOS connection. See the *Installing and Using Clients* for information on determining the correct adapter to use for Windows NT clients.

In the steps that follow, substitute your values with the ones provided in the examples.

Step 2. Setting DB2COMM on the Server

Add or update the environment variable `SET DB2COMM=NPIPE,NETBIOS` in your CONFIG.SYS file and reboot your system. See "Setting the Environment Variable DB2COMM on the Server" on page 46 for details.

Step 3. Updating the Database Manager Configuration File on the Server

Ensure you are logged on as an administrator to the server that contains the database. Then add NNAME for the server to the database manager configuration file. This can be performed using the Database Director or command line processor commands, as follows:

Using the DB2 Database Director:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window opens.

3 Select the database manager that you want to update from the main Database Director window. (The default is **DB2**.)

4 Click on the **Selected** pulldown and choose **Configure**. The DB2 - Configure Notebook opens.

5 Select the **Protocols** tab.

6 Configure the NetBIOS-related information as follows:
nname   db2inst1

7 Click on **OK**.

8 Go back to the **IBM DATABASE 2** folder.

9 Double-click on the **Stop DB2** icon to stop the database manager Wait for the message successful before continuing.

10 Double-click on the **Start DB2** icon to make the changes in the configuration file take effect.

11 Proceed to "Step 4. Updating the Database Manager Configuration File on the Client" on page 56.

Using the Command Line Processor: Update the database manager configuration file with the *workstation name (NNAME)* of the client node. For example, if the NNAME is db2inst1, issue the following commands:

```
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING NNAME db2inst1
DB2STOP
DB2START
```

Step 4. Updating the Database Manager Configuration File on the Client

Ensure you are logged on as an administrator to perform this step. Add the client's NNAME in the database manager configuration file. This can be performed using the DB2 Client Setup, the Database Director, or command line processor commands, as follows:

Using the DB2 Client Setup:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **DB2 Client Setup** icon. The DB2 Client Setup window opens.

3 Select the **Client** menu, then select **Configure**. The Client Configuration window opens.

4 Select **Communication** tab.

5 Configure the NetBIOS-related information as follows:
   Node Name   `client1`

6 Click on **OK**.

7 Proceed to "Step 5. Cataloging the NetBIOS Node on the Client" on page 57.

Using the Database Director:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window opens.

3 Click on the database manager that you want to update from the main Database Director window. (The default is **DB2**.)

4 Click on the **Selected** pulldown and choose **Configure**. The DB2 - Configure Notebook opens.

5 Select the **Protocols** tab and advance to the second page of this section.

6 Configure the NetBIOS-related information as follows:
   nname   `client1`

7 Click on **OK**.

8 Proceed to "Step 5. Cataloging the NetBIOS Node on the Client."

Using the Command Line Processor: Alternately, you can use the following commands:

```
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING NNAME client1
DB2 TERMINATE
```

See the *Command Reference* for more information on all commands.

Step 5. Cataloging the NetBIOS Node on the Client

Cataloging the NetBIOS node adds an entry in the client's node directory that describes the remote database server node and the local adapter number that will be used to access the remote node.

Ensure you are logged on as an administrator to perform this step. You can catalog a
NetBIOS node using the DB2 Client Setup, Database Director, or command line
processor commands.

If you are having trouble logging on to the client, see "Logging On to the
System" on page 93 for information that may help.

Using the DB2 Client Setup:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **DB2 Client Setup** icon. The DB2 Client Setup window
opens.

3 Select the **Node** menu.

4 Select the **New** menu item. The New Node window opens.

5 Select the **NETBIOS** radio button.

6 Configure the NetBIOS-related information as follows:
Name	db2netb1
Workstation Name	db2inst1
Adapter	0

7 Select **OK**.

8 Proceed to "Step 6. Cataloging the Database on the Server and the Client" on
page 59.

Using the Database Director:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window
opens.

3 Click on the [+] for the **database manager** where the database resides.

4 Click on the [+] for **Directories**.

5 Double-click on the **Node Directory**. The panels for the node directory open.

6 Click on the **Directory Entry** pulldown and choose **Catalog**.

7 Configure the NetBIOS-related information as follows:

    a Fill in the field for **Node Name** as follows:
Node	db2netb1

       Then, click on the radio button for the NetBIOS protocol.

    b Configure the NetBIOS-related information as follows:
Server nname	db2inst1
Adapter	0

8 Click on **OK**.

9 Proceed to "Step 6. Cataloging the Database on the Server and the Client" on
page 59.

Using the CATALOG NETBIOS NODE Command: To catalog the remote database server, execute the following commands from your remote client:

```
DB2 CATALOG NETBIOS NODE db2netb1 REMOTE db2inst1 ADAPTER 0
DB2 TERMINATE
```

When you catalog a database that resides in the server instance represented by this node, you must use the value db2netb1 in your CATALOG DATABASE command.

Step 6. Cataloging the Database on the Server and the Client

Next, you must update the database directory on the client and server. See "Cataloging Databases" on page 66 for information.

Configuring TCP/IP

For DB2 Client Application Enabler for OS/2 to access a remote database server through TCP/IP, you must have installed and configured the required TCP/IP Services component of OS/2 Warp Server on both the client and server workstations. See the "Software Requirements" on page 27 for details.

See your IBM Warp Server documentation for detailed information on the installation and configuration of TCP/IP Services.

The following steps include the configuration tasks necessary to set up TCP/IP at the client and server:

1 Complete the TCP/IP worksheet to determine the values you need to enter in the following steps.
2 Set the environment variable DB2COMM on the server
3 Update the SERVICES file on the server
4 Update the database manager configuration file on the server
5 Ensure that the host address can be resolved on the client
6 Update the SERVICES file on the client
7 Catalog the TCP/IP node on the client
8 Catalog the database on the client

Step 1. Assigning Values to the Parameters (Worksheet)

Complete the "Your Value" column in the following worksheet to understand the purpose of the parameters and to assign values of your own to the various parameters. By assigning these values, you will have all the information you need to perform the rest of the steps in this section.

Parameters	Our Example	Your Value	Explanation
TCP/IP-specific Values			
DB2NODE	db2tcp1		Alias or nickname for the database node to be cataloged on the client. Choose a name that makes sense to you. This value does NOT have to be the same as the server HOSTNAME, but must be unique within the node directory.
HOSTNAME	tcphost		The TCP/IP name of the workstation where the database resides. Type hostname at a workstation to determine its hostname.
IP ADDRESS	9.21.15.235		The IP address of the workstation where the database resides. Type ping HOSTNAME to determine the address.
SVCENAME	db2inst1c		The service name is an arbitrary value used by DB2 to connect the client to the server. It must be unique within the services file on the client and on the server, and must be the same in the following entries: • The services file on the server. See Step 3. • The services file on the client. See Step 6. • The node directory for the client. See Step 7.

In the steps that follow, substitute your values with the ones provided in the examples.

Step 2. Setting DB2COMM on the Server

Add or update the environment variable SET DB2COMM=NPIPE,TCPIP in your CONFIG.SYS file and reboot your system. See "Setting the Environment Variable DB2COMM on the Server" on page 46 for details.

Step 3. Updating the Services File on the Server

You must access and edit the SERVICES file. On OS/2 workstations, the hosts file is typically located in either the \MPTN\ETC directory or the \TCPIP\ETC directory on the

drive where TCP/IP or MPTN is installed. (The location depends on the products you have installed on your system. Type SET ETC to determine which directory is used on your system.) Refer to your TCP/IP documentation for further information.

The TCP/IP `SERVICES` file on the server should contain two entries for TCP/IP support for each database manager instance. For example, the entries could be:

```
db2inst1c      3700/tcp       # DB2 connection service port
db2inst1i      3701/tcp       # DB2 interrupt connection service port
```

`db2inst1c` is the value of the SVCENAME parameter and `db2inst1i` is arbitrary. `3700` and `3701` are the port numbers for the connection and interrupt port, and `tcp` is the protocol. The port number `3700` is arbitrary, but must be unique within the file. The second port number must also be unique, and equal to the first number plus one.

1 You must end each line with a comment (`# comment`) or press Enter at the end of each line to insert an end-of-line character.

2 You should refer to the documentation from your TCP/IP product for specific information on host resolution and specifying services.

Step 4. Updating the Database Manager Configuration File on the Server

Ensure you are logged on as an administrator to the server that contains the database. Then add the TCP/IP service port name to the database manager configuration file. This can be performed using either the Database Director or command line processor commands, as follows:

Using the DB2 Database Director:

  1 Open the **IBM DATABASE 2** folder.

  2 Double-click on the **Database Director** icon. The Database Director window opens.

  3 Select the database manager that you want to update from the main Database Director window. (The default is **DB2**.)

  4 Click on the **Selected** pulldown and choose **Configure**. The DB2 - Configure Notebook opens.

  5 Select the **Protocols** tab.

  6 Configure the TCP/IP-related information as follows:
  Service Name   db2inst1c

  7 Click on **OK**.

  8 Go back to the **IBM DATABASE 2** folder.

  9 Double-click on the **Stop DB2** icon to stop the database manager. Wait for the message successful before continuing.

  10 Double-click on the **Start DB2** icon to make the configuration file changes take effect.

  11 Proceed to "Step 5. Resolving the Host Address on the Client" on page 62.

Using the Command Line Processor: Update the database manager configuration file with the following commands:

```
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING SVCENAME db2inst1c
DB2STOP
DB2START
```

where db2inst1c is the Service Name. For more details on all commands, see the *Command Reference*.

> The SVCENAME used must match the connection service port number configured in the SERVICES file on both the client and the server.

Step 5. Resolving the Host Address on the Client

The client workstation must know the address of the host server to which it is attempting to connect. There are two ways to resolve the address of the host:

*   By a name server on your network. This is the recommended approach. See your TCP/IP documentation on how to configure TCP/IP to use a name server.

    If you are using a name server on your network, proceed to "Step 6. Updating the Services File on the Client" on page 63.

- By specifying the host address in the local `hosts` file. On OS/2 or Windows clients, the `hosts` file is typically located in either the `\MPTN\ETC` directory or the `\TCPIP\ETC` directory on the drive where TCP/IP or MPTN is installed. (The location depends on the products you have installed on your system. Type SET ETC to determine which directory is used on your system.)

  Add an entry to the `hosts` file on the client for the server's hostname, as follows:

  ```
  9.21.15.235     tcphost   # host address for tcphost
  ```

  where `9.21.15.235` is the *IP address* and `tcphost` is the HOSTNAME. If the server resides in the same Internet domain as the client, this name can be a flat hostname. If the server is not in the same domain, the name must be a fully specified domain name, such as `TCPHOST.TOROLAB.IBM.COM`, where `TOROLAB.IBM.COM` is an example of a domain name.

  - You must end each line with a comment (`# comment`) or press `Enter` at the end of each line to insert an end-of-line character.
  - You should refer to the documentation from your TCP/IP product for specific information on resolving host addresses.

Step 6. Updating the Services File on the Client

The following information must be added to the `SERVICES` file on the client for TCP/IP support:

```
db2inst1c     3700/tcp     # DB2 connection service port.
                           # Also serves as an interrupt
                           # connection service port.
```

This entry must match the connection service port number that you specified for the server in "Step 3. Updating the Services File on the Server" on page 60.

> The `SERVICES` file is typically located in either the `\MPTN\ETC` directory or the `\TCPIP\ETC` directory on the drive where TCP/IP or MPTN is installed. (The location depends on the products you have installed on your system. Type SET ETC to determine which directory is used on your system.) See your TCP/IP documentation for information.

Step 7. Cataloging the TCP/IP Node on the Client

You must catalog the DB2NODE, HOSTNAME, and SVCENAME information of the remote server node. This adds an entry in the client's node directory that points to the remote DB2 server.

You can catalog a TCP/IP node with the DB2 Client Setup, the Database Director, or the CATALOG TCPIP NODE command using the command line processor. Ensure you are logged on as an administrator to perform this step.

> If you are having trouble logging on to the client, see "Logging On to the System" on page 93 for information that may help.

Using the DB2 Client Setup:

1 Open the **IBM DATABASE 2** folder.
2 Double-click on the **DB2 Client Setup** icon. The DB2 Client Setup window opens.
3 Select the **Node** menu.
4 Select the **New** menu item. The New Node window opens.
5 Select the **TCP/IP** radio button.
6 Configure the TCP/IP-related information as follows:
Name	db2tcp1
Hostname	tcphost
Service Name	db2inst1c
7 Select **OK**.
8 Proceed to "Step 8. Cataloging the Database on the Server and the Client" on page 65.

Using the Database Director:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window opens.

3 Click on the [+] for the database manager instance where the database resides. (The default is **DB2**.)

4 Click on the [+] for **Directories**.

5 Double-click on the **Node Directory**. The panels for the node directory open.

6 Click on the **Directory Entry** pulldown and choose **Catalog**.

7 Configure the TCP/IP-related information as follows:

    a Fill in the field for **db2node** as follows:
      Node   db2tcp1

      Then, click on the radio button for the TCP/IP protocol.

    b Configure the TCP/IP-related information as follows:
      Host name   tcphost
      Service name db2inst1c

8 Click on **OK**.

9 Proceed to "Step 8. Cataloging the Database on the Server and the Client" on page 65.

Using the Command Line Processor: To catalog the remote server, execute the following commands from the client:

```
DB2 CATALOG TCPIP NODE db2tcp1 REMOTE tcphost SERVER db2inst1c
DB2 TERMINATE
```

When you catalog a database that resides in the server instance represented by this node, you must use the value db2tcp1 in your CATALOG DATABASE command.

You can catalog the TCP/IP node by using the *IP address* instead of the HOSTNAME of the remote server node. For example, to catalog the node using the *IP address*, enter (assuming an IP address of 9.21.15.235 for the server):

```
DB2 CATALOG TCPIP NODE db2tcp1 REMOTE 9.21.15.235 SERVER db2inst1c
```

For more details, see the *Command Reference*.

Step 8. Cataloging the Database on the Server and the Client

Next, you must update the database directory on the client and server. See "Cataloging Databases."

Cataloging Databases

Before a client application can access a remote database, the database must be cataloged on the server node and on all client nodes that will connect to the database. The database manager uses the information in the database directory along with the information in the node directory to establish a connection to the database.

When you create a database, it is automatically cataloged on the server.

If you do not have your own database to catalog, go to "Creating the Sample Database" on page 95 to create the sample database.

To catalog databases, do the following steps:

1 Log on as an administrator.

2 Assign values to the worksheet in Table 6.

Complete the "Your Value" column to understand the purpose and to assign values of your own to the various parameters. These values are required before you can catalog any databases.

Table 6 (Page 1 of 2). Worksheet to Use for Cataloging Databases			
Parameters	**Our Example**	**Your Value**	**Explanation**
Local Database - Cataloged at the Server			
Database	sample		Real name of the database on the server.
Alias	sample		Nickname of your choice for the database on the server. By default, this is the same as the database name. To reference the cataloged database, you must use the alias.
Drive	e:		When local, use the drive the database resides on.
Remote Database - Cataloged at the Client			
Database	sample		This is the same name as the alias on the server.
Alias	tor1		Nickname of your choice for the database on the client. If you do not provide an alias, the default is the same as the database name on the client. This is the name that you use when connecting to a database from a client.

Table 6 (Page 2 of 2). Worksheet to Use for Cataloging Databases			
Parameters	**Our Example**	**Your Value**	**Explanation**
Node	db2tcp1, db2netb1, db2ipx1		Use the db2node value that was used when you cataloged the node. This value represents the instance where the database resides.

3 Catalog the database on the server and on the client

You can catalog the database by using the DB2 Client Setup, the Database Director, or by using the command line processor.

Using the DB2 Client Setup: On each client you must catalog the remote databases that can be accessed.

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **DB2 Client Setup** icon. The DB2 Client Setup window opens.

3 Select the node where the database resides to highlight the node. (You can catalog a database without first cataloging the node, but a warning message will be issued.)

4 Select the **Databases** pushbutton. The DB2 Client Setup - Databases window opens.

5 Select the **Database** menu, then select **New**. The New Database window opens.

6 Fill in the remote database information as follow:
Name   sample
Alias   tor1

Optionally you can provide a comment as well.

7 Select **OK**.

8 Proceed to "Testing the Connection" on page 70.

You can test your connection to the database by using the **Test Connection** pushbutton.

Using the Database Director: These instructions show how to catalog a database if it is remote or local. If it is remote, your system is acting as a client. Local databases can only be defined on a server system.

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window opens.

3 Click on the [+] for the database manager instance where the database resides.

4 Click on the [+] for **Directories**.

5 Double-click on the **System database directory**. The panels for the system database directory open.

6 Click on the **Directory Entry** pull-down and choose **Catalog**.

7 Specify the database information as follows:

- For a remote database:

    a Click on the **Remote** radio button.

    b Click on the **Remote** tab.

    c Specify the database information as follows:
Database	sample
Alias	tor1
Node	db2tcp1

- For a local database:

    a Click on the **Local** radio button.

    b Click on the **Local** tab.

    c Specify the database information as follows:
Database	sample
Alias	sample
Drive	e:

8 Click on **OK**.

9 Proceed to "Testing the Connection" on page 70.

Using the Command Line Processor: You can catalog a database on the server with the following command:

```
DB2 CATALOG DATABASE name AS alias ON DRIVE drive
```

You can catalog a database on the client with the following command:

```
DB2 CATALOG DATABASE name AS alias AT NODE db2node
```

name	Is the real name of the database when cataloging on the server and the alias of the remote database when cataloging on the client.
alias	Is the alias you want to use for this database. If you do not provide an alias, the default is the same as the database name.
	The real database name and the alias are stored in the client's database directory. To reference the cataloged database, you must use the alias.
db2node	Is the value you used for *db2node* when you cataloged the node that represents the instance where the database resides. This is entered at the client only.
drive	Is the local drive where the database is located. This is entered at the server only.

Example: A database `sample` resides on the drive `e:` on the server.

To catalog it on the server using the alias name of `sample`, issue this command:

```
DB2 CATALOG DATABASE sample AS sample ON DRIVE e:
```

Now, the client must catalog a node that represents the remote server and the database that resides at the server. The client communicates with this server using the TCP/IP protocol and the remote server node has been cataloged with the command:

```
DB2 CATALOG TCPIP NODE db2tcp1 REMOTE tcphost SERVER db2inst1c
```

To catalog the database using the alias `tor1` at the client, you must use the database alias of the remote database and the remote server node, using this command:

```
DB2 CATALOG DATABASE sample AS tor1 AT NODE db2tcp1
```

> A database can be cataloged before cataloging the node on which it resides; however, a warning that the node has not yet been cataloged will be displayed.

Testing the Connection

Once you have cataloged the node and database at the client workstation and the database at the server workstation, you can test the connection using this command at the client workstation:

```
DB2 CONNECT TO tor1
```

If the connection works, you will get a message showing the database you are connected to.

You must start the database manager on the server before you can connect to a database. Issue the DB2START command at the server.

Now that you have finished cataloging the databases, you are ready to start using the Database Server. See Chapter 8, "Getting Started with the Database Server" on page 91 for details on the many ways you can use Database Server.

Chapter 7. Configuring DDCS

This chapter tells you how to configure APPC communications between your DDCS workstation and DRDA servers. To configure remote OS/2 or Windows clients, see Chapter 6, "Setting Up Communications for the Client and Server" on page 45.

This chapter makes the following assumptions:

- For MVS, VSE, or VM, you have access to a VTAM administrator and a database administrator. For OS/400, you have access to an AS/400 administrator.

- Communications Manager is already installed on the system where DDCS is installed.

- A Token Ring is being used as the communications medium.

- DDCS is already installed.

- DB2 clients have been set up as required.

Configuring Your DRDA Servers

Before you can use DDCS, you must configure both the DRDA servers and the DDCS system. If you decide to configure the DDCS system before you configure the servers, you must at least get information about how they will be configured before you configure the DDCS system.

The following worksheets are used for configuring DRDA servers:

- "MVS, VSE, and VM Server Worksheet" on page 148
- "OS/400 Server Worksheet" on page 150.

For each server that you are connecting to, fill in one copy of the appropriate server worksheet. Each worksheet contains instructions for how to use it.

For a connection to an MVS, VSE, or VM system, you must know the local node name of your machine. You can find this by:

- Reviewing the Local Node Characteristics specification in your existing communications manager configuration

- Referring to your local network administrator.

Figure 5 illustrates the correspondences between information specified on the DRDA server system, information specified in the DDCS configuration files, and information specified in the DDCS directories. The values used in this diagram correspond to the examples shown in the worksheets.
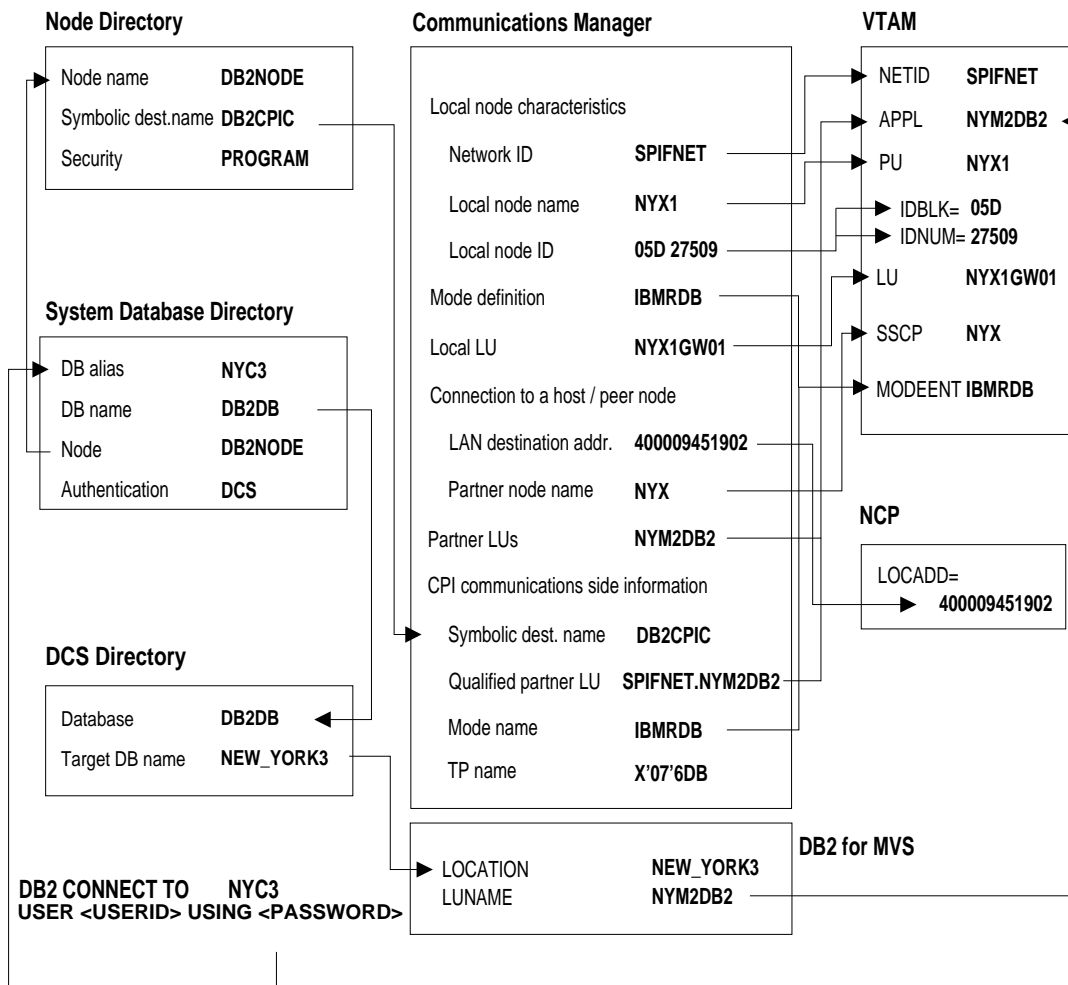
Node Directory

Node name	**DB2NODE**
Symbolic dest.name	**DB2CPIC**
Security	**PROGRAM**

System Database Directory

DB alias	**NYC3**
DB name	**DB2DB**
Node	**DB2NODE**
Authentication	**DCS**

DCS Directory

Database	**DB2DB**
Target DB name	**NEW_YORK3**

DB2 CONNECT TO   NYC3
USER <USERID> USING <PASSWORD>

Communications Manager

Local node characteristics

Network ID	**SPIFNET**
Local node name	**NYX1**
Local node ID	**05D 27509**
Mode definition	**IBMRDB**
Local LU	**NYX1GW01**

Connection to a host / peer node

LAN destination addr.	**400009451902**
Partner node name	**NYX**
Partner LUs	**NYM2DB2**

CPI communications side information

Symbolic dest. name	**DB2CPIC**
Qualified partner LU	**SPIFNET.NYM2DB2**
Mode name	**IBMRDB**
TP name	**X'07'6DB**

DB2 for MVS

LOCATION	**NEW_YORK3**
LUNAME	**NYM2DB2**

VTAM

NETID	**SPIFNET**
APPL	**NYM2DB2**
PU	**NYX1**
IDBLK=	**05D**
IDNUM=	**27509**
LU	**NYX1GW01**
SSCP	**NYX**
MODEENT	**IBMRDB**

NCP

LOCADD=	**400009451902**

Figure 5. Configuration on the DRDA Server and DDCS Workstation

Preparing to Configure Your DDCS System

Use the steps in this section to help you fill in a copy of the "OS/2 Configuration Worksheet" on page 151.

For an APPC connection to an MVS, VSE, or VM connection, perform the following steps:

1 For Network ID, copy the NETID (item **1**) from the server worksheet.
2 For Local node name, copy the PU (item **7**) from the server worksheet.
3 For Local node ID, copy the IDBLK and IDNUM (item **8**) from the server worksheet.
4 For Mode name, copy the MODEENT (item **3**) from the server worksheet.
5 For Local LU name, copy the LU (item **6**) from the server worksheet.
6 For Local LU alias, choose an alias. This can be the same as the LU name or different.

For an APPC connection to an OS/400 system, perform the following steps:

1 For Network ID, copy the local network ID (item **1**) from the server worksheet.
2 For Mode name, copy the mode name (item **3**) from the server worksheet.
3 Leave the other items blank for now.

Preparing for Each Server Connection

For each server that you are connecting to, fill in a copy of the "OS/2 Connection to Server Worksheet" on page 152 as follows:

1 For Link name, choose a value.
2 For LAN destination address, copy item **2** (controller address or local adapter address) from the server worksheet.
3 For Symbolic destination name, choose a value.
4 For Partner CP name, copy item **4** from the server worksheet. For MVS, VSE, or VM it is an SSCP value; for OS/400 it is a local control point name.
5 For Partner LU name for MVS, VSE, or VM, copy item **5** (APPL) from the server worksheet. For OS/400, copy the Partner CP name that you just wrote on this worksheet.
6 For Remote transaction program, copy the value from the server worksheet. The default is X'07'6DB for MVS and OS/400 and the RDB_NAME for VSE and VM.
7 For Target database name, copy the last item from the server worksheet. This is a LOCATION value (MVS), RDB_NAME (VSE or VM), or relational database name (OS/400).

Configuring APPC for DDCS

Before using this section, be sure that you have completed the "OS/2 Configuration Worksheet" on page 151.

1 The following steps have been documented using Communications Server Version 4 and are intended to show you what information is required. You must prepare a configuration containing profiles similar to those shown in the examples, but specifying your own unique values from the worksheets.

2 In general, these instructions describe how to create new profiles within a new configuration. If you are modifying an existing configuration, then some profiles may need to be deleted before the configuration can be verified successfully.

3 The boxed numerics correspond with the items located on your worksheet; for example, the symbol **8** would be used to represent the LAN Destination Address.

Begin to configure your DDCS system with the following steps:

1 Double-click on the Communications Server setup icon.

2 On the Communications Server Setup panel, click on the **Setup** pushbutton.

3 On the Open Configuration panel:

    a Specify the name of an existing configuration file if you have one, or provide a name for a new file to be created.

    b Click on the **OK** pushbutton to proceed to the Communications Manager Configuration Definition panel.

4 The Configuration Definition panel appears as follows:
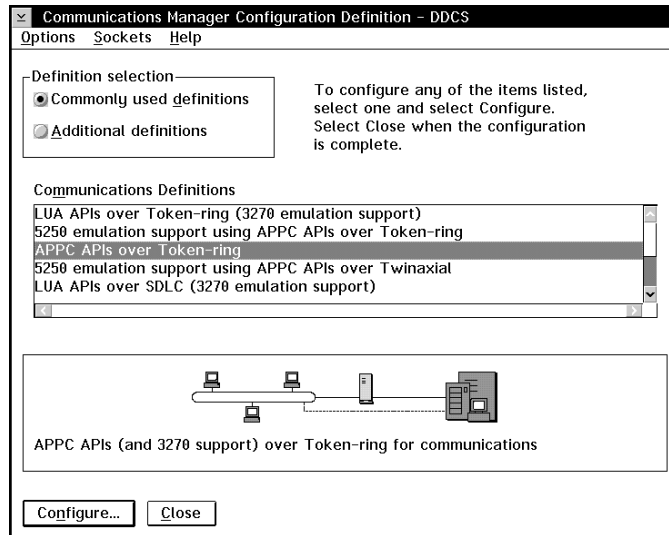
a Click on the **Commonly Used Definitions** radio button.

b In the Communications Definitions selection window, select your choice of protocol. These examples use **APPC APIs over Token-Ring**.

c Click once on the **Configure** pushbutton to proceed.

Alternatively, a double click on the selection of your choice will take you directly to the next panel.

If this is a new configuration file the next panel displayed will be the APPC APIs over Token-Ring panel. Otherwise the next panel displayed will be the Communication Manager Profile List panel.

5 The APPC APIs over Token-Ring panel appears as follows:
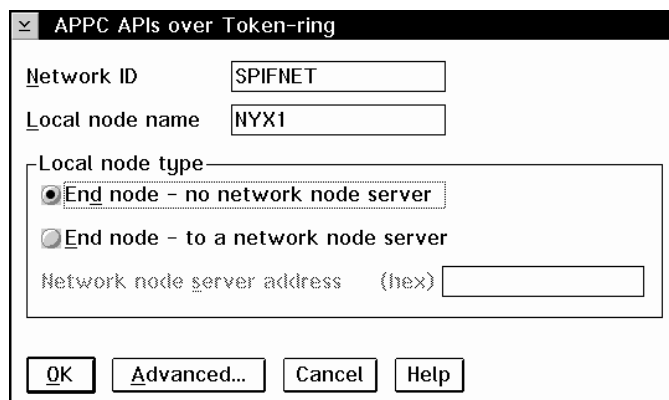
a Type in your values for:

    1 Network ID

    2 Local node name

b Click on the **End node** radio button

You have a choice between **End node - to a network node server** and **End node - no network node server**. Your network administrator should advise you on which to use. Typically, a network node server is used when many users are routed through the same connection. The example here presumes no network node server is used.

c Click on the **Advanced** pushbutton to proceed to the Communication Manager Profile List panel, as shown:
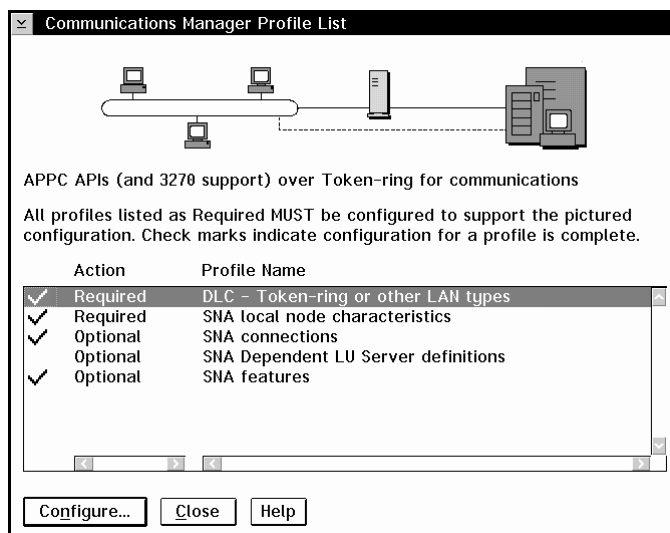
6 Subsequent steps begin from this panel. You will return to this panel when each step is complete, and before saving the configuration.

LAN DLC Profile

From the Communication Manager Profile List panel, prepare a LAN DLC profile as follows:

1 Select **DLC - Token ring or other LAN types**.

2 Select the **Configure** pushbutton.

3 For **C&SM LAN ID**, type the value that you wrote in item **1** of the worksheet.

4 Check that the other values are appropriate for your environment. The values in the example are default values.

5 Click on **OK** to return to the Communication Manager Profile List panel.

SNA Local Node Characteristics

Update SNA Local Node Characteristics as follows:

1 On the Communications Manager Profile List panel, select **SNA local node characteristics** and click on the **Configure** pushbutton. The Local Node Characteristic panel appears, as follows:

2 For **Network ID**, type the value that you wrote in item **1** of the worksheet.

3 The local node name was probably set when Communications Manager was installed. For MVS, VSE, and VM, check that this value matches item **2** on the worksheet. For OS/400, write the local node name in the worksheet now; if no name exists, talk to your LAN administrator.

4 For **Local node ID (hex)**, type the value that you wrote in item **3** of the worksheet. For OS/400, you can leave the second part of the node ID blank or specify `00000`; the first 3 characters are automatically filled in.

5 Select the **Options** pushbutton. The Local Node Options panel appears, as follows:

a Verify that the **Activate Attach Manager at Startup** check box is selected.

b Click on **OK** to return to the Local Node Characteristics panel.

6 Click on **OK** to return to the Communication Manager Profile List panel.

SNA Connections

Before following the steps in this section, be sure you have completed the "OS/2 Connection to Server Worksheet" on page 152.

Prepare the connection profiles as follows:

1 From the Communications Manager Profile List panel, select **SNA Connections**, and click on the **Configure** pushbutton.

2 On the Connections List panel, for **Partner Type** select either the **To peer node** radio button (normally used for OS/400 connections), or the **To host** radio button, (normally used for MVS, VSE, and VM connections), and click on **Create**.

3 The Adapter List panel appears, as follows:
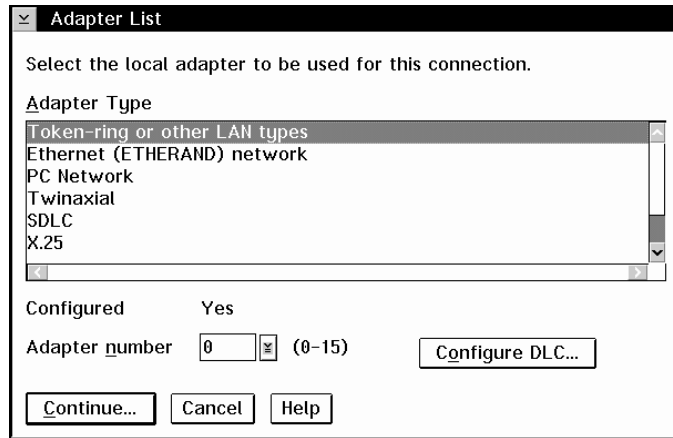
4 Select adapter type **Token-ring, or other LAN types**, and specify the same **adapter number** that you specified in the DLC profile.

5 Select the **Continue** pushbutton to proceed to the Connection to a Peer Node panel or the Connection to Host panel. Steps to configure these panels are in the next section.

Connection to a Peer or Host Node:

At the Connection to a Peer Node panel or the Connection to Host panel:

1. Specify the **Link name** value that you wrote in item **7** of the worksheet.

2. On the Connection to Host panel, change the **Local PU name** to the value that you wrote in item **2** of the worksheet and change the node ID to the value that you wrote in item **3** of the worksheet.

3. Change **LAN destination address** to the value that you wrote in item **8** of the worksheet.

4. Change **Partner network ID** to the value that you wrote in item **1** of the worksheet.

5. Change **Partner node name** to the value that you wrote in item **10** of the worksheet.

6. Select **Define Partner LUs** on either the Connection to a Peer Node panel or the Connection to Host panel to proceed.

Partner LUs:

Proceed as follows:

1 For **Network ID** type the value that you wrote in item **1** of the worksheet.

2 For **LU name** and **Alias**, type the value that you wrote in item **11** of the worksheet.

3 Select the **Add** pushbutton to add the partner LU profile to the connection profile.

4 Click on **OK** to return to the previous panel.

5 On the Connection to Host panel, click on the **Additional Parameters** pushbutton.

Additional Connection Parameters:

If you have selected to configure connections to a host, the following panel appears:

1 Verify that the Multiple PU Parameters are filled in. This value is the local node ID in hex, item **3** in the worksheet.
Local   05D 27509

2 Click on **OK** to return to the Connection to a Host panel.

3 Click on **OK** to return to the Connections List panel.

4 Click on **Close** to return to the Communications Manager Profile List panel.

SNA Features

On the Communications Manager Profile List panel, select **SNA features**, and click on the **Configure** pushbutton. Subsequent steps begin from the SNA Features List panel that appears, as follows:
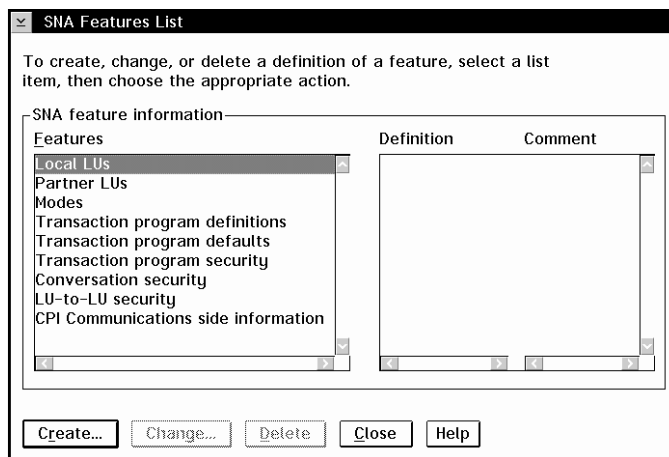
Local LU Profile:

If the DDCS workstation is defined as an independent LU, prepare a Local LU Profile as follows. In most cases, a VTAM administrator on MVS, VSE, or VM would define an independent LU for DDCS. For OS/400, the control point would normally be used instead of an independent LU.

1. On the *SNA Features List* panel, click on **Local LUs** and click on **Create**.
2. For **LU name**, type the value that you wrote in item **5** of the worksheet. For **alias**, type the value that you wrote in item **6** of the worksheet. For **NAU address**, select the **Independent LU** radio button.
3. Click on **OK** to return to the SNA Features List panel.
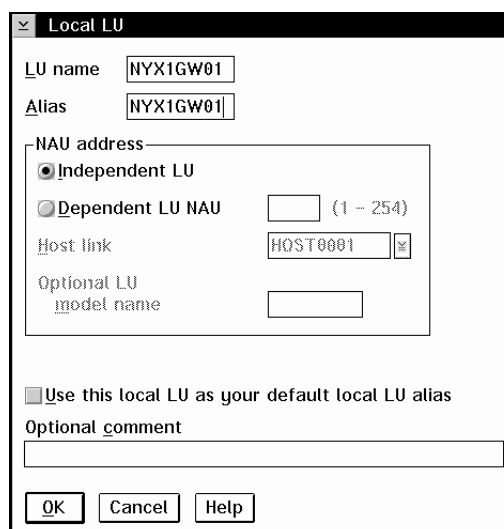
In order to use this local LU when the DDCS workstation starts the APPC connection, check off the box on the Local LU Profile that says "Use this local LU as your default local LU alias". By default, all APPC connections that are started from this DDCS workstation will use this local LU.

Partner LU Profile:

The partner LU profile was configured in a previous step. To verify, click on **Partner LU Profile** from the SNA Features List panel to see if a partner LU is defined.
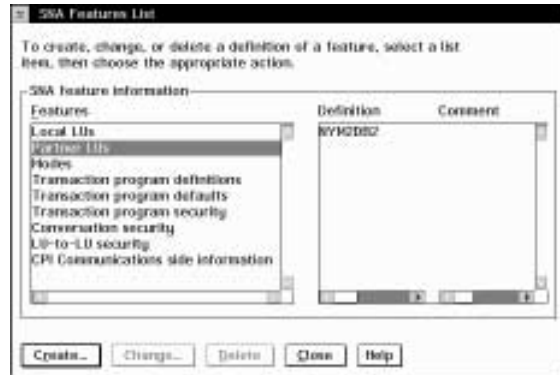
Mode Definition:

From the SNA Features List panel, prepare a Mode Definition as follows:

1 Select **Modes** and click on **Create**.

2 Define your mode profile as follows:

    a For **mode name**, use the value that you wrote in item **4** of the worksheet.

    b On the other lines, specify values that match the mode profile defined on your DRDA server systems.

3 Click on **OK** to finish the creation of the mode. and to return to the SNA Features List panel.

CPIC Side Information:

On the SNA Features List panel, select **CPI Communications Side Information**, and click on **Create**. The CPI Communications Side Information panel appears.

Complete the CPI Communications Side Information panel as follows:

1 For **Symbolic destination name**, type the value that you wrote in item **9** of the worksheet.

2 Select the **Alias** radio button. Select the arrow to view a list of all defined aliases and choose the alias that corresponds to **11** in the worksheet.

3 In the **Partner TP** box, specify the remote **Transaction Program (TP)** name (item **12** of the worksheet).

Select **None** as the **Security type** (this does not mean that you will have no security, you will specify the security type later, in "Completing the Configuration" on page 85).

4 For the mode name, select the value that you wrote in item **4** of the worksheet.

5 Click on **OK** to save the CPI side information profile and return to the SNA Features List panel.

6 Click on **Close** to return to the Communications Manager Profile List panel.

Saving Your Configuration

You are now ready to save your Communications Manager configuration file:

1 Click on **Close** to return to Communications Manager Configuration Definition Panel.

2 Click on **Close** to automatically verify and save the new configuration file, and leave the configuration panels.

Completing the Configuration

The steps you have just completed show how to set up the SNA configuration. Once this is complete, you must configure DDCS as described in the following steps:

1 Stop and start Communications Manager. Click on the **Stop Communications Normally** icon and the **Start Communications** icon.

2 Update the node directory, system database directory, and DCS directory using the Database Director or command line processor. (Instructions for each are described "Updating Directories Using the Database Director" and "Updating Directories Using Commands.")

3 Configure each remote client as described in Chapter 6, "Setting Up Communications for the Client and Server" on page 45.

4 Connect to the DRDA Server and bind the utilities and applications to the DRDA server using commands similar to the following:

```
DB2 CONNECT TO dbalias USER userid USING password
DB2 BIND path@ddcsmvs.lst BLOCKING ALL
    SQLERROR CONTINUE MESSAGES ddcsmvs.msg GRANT PUBLIC
DB2 CONNECT RESET
```

These commands are described in detail in the *DDCS User's Guide*.

Once you have completed these steps, see Chapter 8, "Getting Started with the Database Server" on page 91 for details on the many functions that are available to use with Database Server.

Updating Directories Using the Database Director

The following directories need to be updated:

- Node directory
- System database directory
- Database connection services directory

The following steps show how to update them using the Database Director.

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window opens.

3 Click on the [+] for the database manager instance that you want to work with.

4 Click on the [+] for **Directories**.

5 Double-click on **System Database Directory**. The window for the system
database directory opens.

6 Click on the **Directory Entry** pull-down and choose **Catalog**.

7 Catalog the database as follows:

    a Click on the **Remote** radio button.

    b Click on the **Remote** tab.

    c Fill in the remote database information as follows:

Database	db2db
Alias	nyc3
Node	db2node

    d Click on the box for **DDCS or backlevel database**.

    e For authentication, click on the **DCS** radio button.

8 Click on **Catalog node** to display the Node Directory Entry - Catalog window with
the name of the node already specified.

9 Click on the **APPC** radio button and enter the protocol information as follows:

Destination Name	DB2CPIC
Security	**Program** radio button.

10 Click on **OK** to save the changes and return to the System Database Directory
Entry - Catalog window. (Messages may appear to indicate that the directory
changes may not be effective until the directory cache is refreshed. Click on **OK**
to continue.)

11 Click on **Catalog database** to display the Database Connection Services
Directory Entry - Catalog window with the name of the database already specified.

12 In **Target database**, type new_york3.

13 Click on **OK** to save the changes and return to the System Database Directory
Entry - Catalog window. (Messages may appear to indicate that the directory
changes may not be effective until the directory cache is refreshed. Click on **OK**
to continue.)

14 Click on **OK** to save the changes and close the window. (Messages may appear
to indicate that the directory changes may not be effective until the directory
cache is refreshed. Click on **OK** to continue.)

Updating Directories Using Commands

Type the following command line processor commands at a command prompt:

```
DB2 CATALOG APPC NODE db2node REMOTE db2cpic SECURITY program
DB2 CATALOG DATABASE db2db AS nyc3 AT NODE db2node AUTHENTICATION DCS
DB2 CATALOG DCS DATABASE db2db AS new_york3
```

To do this, you will need to know the symbolic destination name (item **9**) and the target database name (item **13**) from the worksheet.

Part 3. Using the Database Server

Chapter 8. Getting Started with the Database Server

This section provides instructions on using the Database Server in a variety of ways. The topics covered include:

- Performing general tasks such as logging on, starting and stopping the database manager, and running applications

- Creating queries to have data presented in a meaningful way using the command line processor

- Performing administration tasks such as managing database directories, tuning SQL statements and system performance, and setting configuration parameters

- Creating applications that contain voice, photos, or movies or that are accessible through the world wide web.

To get started FAST with Database Server, follow these steps.

The following list shows you the steps to follow to install, set up, and use the Database Server products.

1 Decide how you want to set up your configuration — a single super server or several servers dedicated to specific functions. See Chapter 3, "Planning" on page 19 for guidance.

2 Ensure that the system you are about to install on has the required hardware and software. See "Hardware Requirements" on page 23 and "Software Requirements" on page 27 for information.

3 Install the server components.

Insert the CD-ROM in the CD-ROM drive. Switch to the drive letter of the CD-ROM drive. Type: `X:\<LANGUAGE>\INSTALL`. Where X is the CD-ROM drive and <LANGUAGE> is the language directory from which you with to run the installation images. Reboot your system once you have installed all the products you want to install.

See Chapter 4, "Installing Servers" on page 31 for details.

4 Log on to the system with a user who has administrative authority. If a user does not already exist, you might have to create users. See "Logging On to the System" on page 93 for information.

5 Start the database manager. See "Starting and Stopping the Database Manager" on page 94 for information.

6 Create the sample databases (if necessary). See "Creating the Sample Database" on page 95 for information.

The application "First Steps" combines the above three steps. "First Steps" is automatically started the first time you restart your system after installing Database Server. This application is also available in the **IBM DATABASE 2** folder. Double click on the **First Steps** icon to invoke it.

7 Use the Database Director to configure databases. The sections in Chapter 6, "Setting Up Communications for the Client and Server" on page 45 show you how to use Database Director to configure the protocols and databases.

8 You are now ready to use the databases on the server.

9 Install the clients on your LAN.

See "Software Requirements" on page 27 for details on the prerequisites and Chapter 5, "Installing Clients" on page 39 for details on installing clients on your network.

10 On OS/2 and Windows 3.1 clients, use the "Client Setup" window to help with some of the necessary setup tasks. This tool is invoked by double-clicking on the **DB2 Client Setup** icon in the **IBM DATABASE 2** folder located on the client workstation. The sections in Chapter 6, "Setting Up Communications for the Client and Server" on page 45 show you how to use DB2 Client Setup to configure the protocols and databases.

11 You are now ready to use your clients with your choice of:

- SQL statements using query tools or the command line processor. See Chapter 9, "Querying the Database" on page 101 for information.

- Applications such as Lotus Approach through the ODBC interface. See "Running ODBC-Enabled Applications" on page 99 for information.

- The command line processor. See "Using the Command Line Processor" on page 122 for information.

- Your own applications. See "Running Your Own Applications" on page 95 for information.

Logging On to the System

To use DB2, you must be logged on with a valid user ID that has the appropriate authority level for the commands you want to execute.

You need administrative authority to start the database manager or to create the sample database. In other words, for OS/2 and Windows, you must log on using a user ID defined with a user type of administrator.

See the *Administration Guide* for information on planning for and creating a secure database system.

To log on from the server, use the following steps:

1 Click on the **Logon** pushbutton from the First Steps Application. (First Steps starts the first time you restart your system after installing DB2 and can also be found in the DATABASE 2 folder.)

2 Fill the User ID and Password fields and click on **OK**.

3 If you do not have a user ID and password defined, use UPM to create one.

Or

1 Open the **User Profile Management Services** folder.

2 Double-click on **Logon**.

3 Fill the User ID and Password fields and click on **OK**.

- When using an OS/2 or Windows client, if you are not logged on, DB2 will display the local logon window once you attempt to start the database manager.

- If you are using Warp Server, an administrative user ID was created during installation. Use this user ID to perform the steps that require administrative authority or to create additional users with the correct authority. Previous versions of OS/2 are initially set up with a default user ID (USERID) and password (PASSWORD). Try logging on with the defaults if you are unsure of any other administrative user on your system.

Starting and Stopping the Database Manager

You must start the database manager on the server before you can connect to a database, precompile an application, or bind a package to a database. The database manager must also be started to use DDCS for OS/2.

The commands to start and stop the database manager are:

db2start To start the database manager

    1 Open the **IBM DATABASE 2** folder.

    2 Double-click on the **Start DB2** icon.

or

    • Enter the DB2START command at a command prompt.

db2stop To stop the database manager

    1 Open the **IBM DATABASE 2** folder.

    2 Double-click on the **Stop DB2** icon.

or

    • Enter the DB2STOP command at a command prompt.

When you want to terminate DB2, use the DB2STOP command (and, if necessary, the DB2 FORCE command to terminate all active applications connecting to the databases).

See the *Command Reference* for more information about these and other commands.

Creating the Sample Database

You might want to create the sample database that is provided with DB2 Server. The tables in the sample database show many of the features that are available in the product. Many examples are provided in the product documentation that use the sample tables to show you how to use the functions described.

To create the sample databases:

- Click on **Create the sample database** from the **DB2 First Steps** application

OR

- Enter the DB2SAMPL command at a command prompt.

- The database is automatically cataloged with an alias of SAMPLE when it is created.
- The SAMPLE database uses around 8 MB of disk space and takes approximately 5 minutes to create.

Running Your Own Applications

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

To run a database client application that was developed using DB2 Software Developer's Kit, 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, a DB2START command must be issued at the server before starting the application.

2 Ensure that you know the security requirements on the database server, and if you require a password or user ID, that you have the correct ones.

3 Bind the utilities to the database. See "Binding CLI (ODBC) and Database Utilities" for information.

4 Run the application programs.

Binding CLI (ODBC) and Database Utilities

A number of database utilities, like IMPORT, EXPORT, REORG, the command line processor, DB2 CLI, and the binder program, have associated bind files that must be bound to each database before they can be used with that database. The bind files are grouped together in different `.lst` files, each one being specific to a server.

You may bind the database components to your databases using either the DB2 Client Setup or command line processor commands. Each of these methods is described.

Using the DB2 Client Setup: Perform the following steps to bind database components to the sample database:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **DB2 Client Setup** icon. The DB2 Client Setup window opens.

3 Click on the **Assistance** pushbutton. The DB2 Client Setup - Assistance window opens.

4 Click on the **Bind Program to Database** pushbutton.

5 Select a database to work with. Click on the **Next** pushbutton to continue.

6 Specify the bind files to use by filling in the Bind window. To bind the utilities, DB2 CLI, and ODBC to the SAMPLE database, using the DB2UBIND.LST and DB2CLI.LST files located in the E:\SQLLIB\BND directory, fill in the following information:

   Type of file   Bind and list file
   Drive   e:
   Directory   \SQLLIB\BND
   File   DB2UBIND.LST, DB2CLI.LST

7 Click on **Bind** to continue.

8 Enter your **userid** and **password** in the next window. Click on **OK** to bind the files.

9 A progress window appears providing any information or error messages that arise.

10 Click on **Close** to return to the Assistance window.

11 See the notes at the end of this section to determine if any are applicable to your environment.

Using the Command Line Processor: Change directory to the SQLLIB\BND directory and issue the following commands:

```
DB2 CONNECT TO dbname
DB2 BIND @db2ubind.lst MESSAGES bind.msg GRANT PUBLIC
DB2 BIND @db2cli.lst MESSAGES clibind.msg GRANT PUBLIC
DB2 CONNECT RESET
```

where

- DBNAME is the name of the database to which you wish to connect
- BIND.MSG and CLIBIND.MSG are the output message files
- GRANT PUBLIC grants EXECUTE and BIND privileges to PUBLIC.

1 The BIND command must be run separately for each database that you wish to access.

2 In a network environment, if you are using clients that run on different operating systems or are at different versions, then you must bind the utilities once for each operating system / version combination.

3 The utilities only have to be bound to each database once for each operating system / version combination.

4 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.

5 Binding may take a few minutes to complete.

Running ODBC-Enabled Applications

Before you can use ODBC applications such as Lotus Approach to access data in DB2 databases, you must perform the following steps:

These steps show how to set up ODBC for an OS/2 client. The steps vary slightly for Windows clients. See *Installing and Using DB2 Clients* for details.

1 Click on the **ODBC Installer** icon to install the ODBC driver. This does the following:

- Installs the ODBC driver manager, if necessary.

- Adds "IBM DB2 ODBC DRIVER" to the ODBC drivers section in the ODBCINST.INI file.

- Adds a new section named "IBM DB2 ODBC DRIVER" and adds both the "Driver" and "Setup" entries for this new section in the ODBCINST.INI file. This configures the ODBC driver manager to be able to use the DB2 CLI driver.

You may need to shutdown and restart your system to complete this step.

2 Verify that the ODBC driver manager is installed. To do this:

a Open the **ODBC** folder.

b Double-click on the **Administrator** icon.

The Data Sources window appears.

c Select the **Drivers** pushbutton. The Drivers window appears.

d Verify that "IBM DB2 ODBC Driver" is shown.

e Close the Data Sources and Drivers windows.

3 The database (and node if the database is remote) must be cataloged. See "Cataloging Databases" on page 66 for information.

4 The ODBC driver bind files must be bound to the database. See "Binding CLI (ODBC) and Database Utilities" on page 96 for information.

The ODBC driver will autobind on the first connect to the database, provided the user has the appropriate privilege or authorization. The administrator may need to perform the first connect or explicitly bind the required files.

5 Use the DB2 Client Setup to register the database with the ODBC driver manager as a *data source* by following these steps:

    a Open the **IBM DATABASE 2** folder.

    b Double-click on the **DB2 Client Setup** icon. The DB2 Client Setup window opens.

    c Click on the **Assistance** pushbutton. The DB2 Client Setup - Assistance window opens.

    d Click on the **Allow programs access to a DB2 database via ODBC** pushbutton.

    e Select a database to work with. Click on the **Next** pushbutton to continue.

    f In the Configure ODBC Data Source window, select **Data source and description** from the **Settings for** item list. Enter a description for the data source and click on **OK**

    g Click on **OK** to save the changes to the DB2CLI.INI file and to return to the Assistance window.

6 Start an application that supports ODBC, such as Lotus Approach. When you open **ODBC databases** from within the application, you are presented with a list of the databases you can connect to, including the one just registered.

Chapter 9. Querying the Database

To query data in a database, you must use SQL statements. This section shows you how to use SQL to make basic queries against the sample database through the command line processor.

If you are familiar with SQL, you may want to perform the exercises in this chapter using a query tool such as Visualizer Flight or Lotus Approach. (Visualizer Flight is an OS/2 query tool included with Database Server.)

Learning About SQL

This section contains information about basic concepts of the Structured Query Language (SQL) and how to perform common operations to retrieve, enter, and modify data stored in relational databases using SQL statements. For a complete description of SQL, see the *SQL Reference*.

A *relational database* presents data as a collection of tables: a natural and easy to use format. A *table* consists of a defined number of *columns* and any number of *rows*.

```
                              ┌────────┐
                              │ Column │
                              └───┬────┘
                                  │
                                  ▼
        ┌─────┬─────────┬───────┬──────────┐
        │ ID  │ NAME    │ JOB   │ SALARY   │
        ├─────┼─────────┼───────┼──────────┤
        │ 10  │ Sanders │ Mgr   │ 18357.50 │
┌─────┐ │ 20  │ Pernal  │ Sales │ 18171.25 │
│ Row ├─▶ 190 │ Sneider │ Clerk │ 14252.75 │
└─────┘ │ 210 │ Lu      │ Mgr   │ 20010.75 │
        └─────┴─────────┴───────┴──────────┘
```
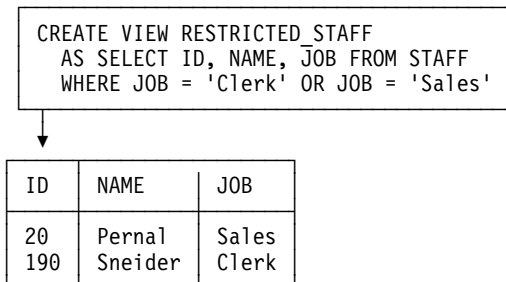
In this example, information about employees is kept in the STAFF table. The last name of each employee is in a column called NAME and employee numbers are in a column called ID. This table has other columns as well. To find just the last names and employee numbers of the staff, just the request:

```
SELECT NAME, ID
   FROM STAFF
```

NAME	ID
Sanders	10
Pernal	20
Sneider	190
Lu	210
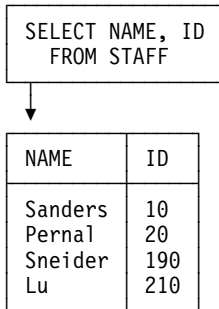
You need not be concerned about other columns in the table, or the locations in the table of columns you do want. You can access many rows of data with a single statement; or access data selectively from one or more tables, rearrange it, and present it in different ways.

You can request data once, or define views for repeated use. A *view* is derived from columns in one or more tables or views. A resulting view looks like a table and is accessed in the same way, but is really a filter that points to data in tables.

Because a view shows selected columns and rows, different people can access different columns or rows in the same tables. Sensitive data can be protected from general users, while individual users all have access to the data they need. For example a view RESTRICTED_STAFF could be based on the table STAFF so that people using that view see only the clerks and sales people and they do not see the SALARY column:

```
CREATE VIEW RESTRICTED_STAFF
   AS SELECT ID, NAME, JOB FROM STAFF
   WHERE JOB = 'Clerk' OR JOB = 'Sales'
```

ID	NAME	JOB
20	Pernal	Sales
190	Sneider	Clerk

Using SQL Statements to Retrieve Data

Use the SELECT statement to retrieve (query) data from your database. Using some basic queries, you can select specific columns or rows from a table to display, prevent duplicate information from being displayed, and use expressions to calculate values to be displayed.

Throughout the rest of this section we will present statements to be keyed and, after most of these, we will show the results that will be displayed when that statement is issued against the sample tables (**assuming that those tables have not been modified**). Note that, although we show the statements in upper case they may be keyed in any mixture of upper and lower case characters (except where they are enclosed in either apostrophes (') or quotes (")).

All of these examples use either the ORG table or the SAMPLE table. Depending on how your database has been set up, it may be necessary to "qualify" those table names by prefixing them with some characters and a dot, for instance PELLOW.ORG and PELLOW.SAMPLE. Ask your administrator whether or not this is necessary and, if it is, for the qualifier to use.

Selecting Columns

To select specific columns from a table, specify a list of column names separated by commas. This list is referred to as a *select-list*. The following statement selects department names and department numbers from the ORG table of the SAMPLE database:

```
SELECT DEPTNAME, DEPTNUMB
   FROM ORG
```

The above statement produces the following result:

```
DEPTNAME           DEPTNUMB
--------------     --------
Head Office              10
New England             15
Mid Atlantic            20
South Atlantic          38
Great Lakes             42
Plains                  51
Pacific                 66
Mountain                84
```

To select all the columns from a table, use an asterisk (*) in the SELECT statement. The following statement selects all columns and all rows from the ORG table:

```
SELECT *
   FROM ORG
```

This statement produces the following result:

```
DEPTNUMB   DEPTNAME        MANAGER   DIVISION    LOCATION
--------   --------------  -------   ---------   -------------
      10   Head Office        160    Corporate   New York
      15   New England         50    Eastern     Boston
      20   Mid Atlantic        10    Eastern     Washington
      38   South Atlantic      30    Eastern     Atlanta
      42   Great Lakes        100    Midwest     Chicago
      51   Plains             140    Midwest     Dallas
      66   Pacific            270    Western     San Francisco
      84   Mountain           290    Western     Denver
```

Selecting Rows

To select specific rows from a table, use the WHERE clause to specify the condition or conditions that a row must meet in order to be selected. The following example selects only those rows for department 20:

```
SELECT DEPT, NAME, JOB
  FROM STAFF
  WHERE DEPT = 20
```

This statement produces the following result:

```
DEPT   NAME       JOB
-----  ---------  -----
   20  Sanders    Mgr
   20  Pernal     Sales
   20  James      Clerk
   20  Sneider    Clerk
```

If you are selecting rows based on a character value, that value must be enclosed in single quotation marks (for example, WHERE JOB = 'Clerk').

Each character value must be typed exactly as it exists in the database. If the data value is lowercase in the database and you type it as uppercase, no rows are selected. If you are selecting rows based on a numeric value, that value must not be enclosed in single quotation marks (for example, WHERE DEPT = 20).

The following example selects clerks in department 20 from the STAFF table:

```
SELECT DEPT, NAME, JOB
  FROM STAFF
  WHERE JOB = 'Clerk' AND DEPT = 20
```

This statement produces the following result:

```
DEPT   NAME       JOB
-----  ---------  -----
   20  James      Clerk
   20  Sneider    Clerk
```

A special value called the *null value* is used when there is no value for a given column in a row. The following WHERE clause selects rows that have null value in a column:

```
WHERE columnname IS NULL
```

A null value occurs where no value is entered and the column does not support a default value. It can also occur where the value is specifically set to NULL. It can only occur in columns which are defined to support null values.

The following statement lists employees whose commission is not known:

```
SELECT ID, NAME
  FROM STAFF
  WHERE COMM IS NULL
```

This statement produces the following result:

```
    ID   NAME
------   ---------
    10   Sanders
    30   Marenghi
    50   Hanes
   100   Plotz
   140   Fraye
   160   Molinare
   210   Lu
   240   Daniels
   260   Jones
   270   Lea
   290   Quill
```

The value zero is not the same as the null value. The following statement selects
everyone in a table whose commission is zero:

```
SELECT ID, NAME
  FROM STAFF
  WHERE COMM = 0
```

Since there are no values of zero in the COMM column in the sample table, the result
set returned is empty.

In the previous examples, the conditions specified in the WHERE clause are based on
either the equal (=) predicate or the NULL predicate. A *predicate* specifies a condition
that is true or false (or unknown) about a row. Many other predicates are available.
For example, the following statement selects all rows where the value of YEARS in the
STAFF table is greater than 9:

```
SELECT NAME, SALARY, YEARS
  FROM STAFF
  WHERE YEARS > 9
```

This statement produces the following result:

```
NAME         SALARY   YEARS
---------   --------  -----
Hanes       20659.80     10
Lu          20010.00     10
Jones       21234.00     12
Quill       19818.00     10
Graham      21000.00     13
```

For more information about predicates, see the *SQL Reference*.

Putting Rows in Order

The order of the information in the result may be important. You can use the ORDER BY clause to put the information in order by the values in one or more columns. The following statement displays the employees in department 84 in order by number of years employed.

```
SELECT NAME, JOB, YEARS
  FROM STAFF
  WHERE DEPT = 84
  ORDER BY YEARS
```

ORDER BY must be the last clause in the entire SELECT statement. Any columns named in the ORDER BY clause must also be named in the select-list.

This statement produces the following result:

```
NAME        JOB     YEARS
---------   -----   -----
Davis       Sales       5
Gafney      Clerk       5
Edwards     Sales       7
Quill       Mgr        10
```

Columns can be ordered in ascending or descending order by explicitly specifying either ASC or DESC within the ORDER BY clause. If neither is specified, the rows are automatically ordered in ascending sequence. The following statement displays the employees in department 84 in descending order by years employed:

```
SELECT NAME, JOB, YEARS
  FROM STAFF
  WHERE DEPT = 84
  ORDER BY YEARS DESC
```

This statement produces the following result:

```
NAME        JOB     YEARS
---------   -----   -----
Quill       Mgr        10
Edwards     Sales       7
Davis       Sales       5
Gafney      Clerk       5
```

Results can be ordered by character values as well as numeric. Character data is sorted in an order determined by the collating sequence defined when the database was created. See your administrator for details.

The following statement displays the employees in department 84 in alphabetical order by name.

```
SELECT NAME, JOB, YEARS
  FROM STAFF
  WHERE DEPT = 84
  ORDER BY NAME
```

This statement produces the following result:

```
NAME        JOB     YEARS
---------   -----   -----
Davis       Sales       5
Edwards     Sales       7
Gafney      Clerk       5
Quill       Mgr        10
```

Removing Duplicate Rows

When using the SELECT statement, you may not want duplicate information to be returned. For example, STAFF has a DEPT column in which several department numbers are listed more than once, and a JOB column in which several job descriptions are listed more than once. The following statement selects all the jobs that in departments whose department number is less than 30:

```
SELECT DEPT, JOB
  FROM STAFF
  WHERE DEPT < 30
  ORDER BY DEPT, JOB
```

This example produces the following result (containing many duplicate rows):

```
DEPT   JOB
------ -----
    10 Mgr
    10 Mgr
    10 Mgr
    10 Mgr
    15 Clerk
    15 Clerk
    15 Mgr
    15 Sales
    20 Clerk
    20 Clerk
    20 Mgr
    20 Sales
```

The DISTINCT option on the SELECT clause can be used to eliminate duplicate rows in a result. For example, if DISTINCT is inserted into the statement each job within a department will be listed only once:

```
SELECT DISTINCT DEPT, JOB
  FROM STAFF
  WHERE DEPT < 30
  ORDER BY DEPT, JOB
```

This example produces the following result:

```
DEPT    JOB
------  -----
    10  Mgr
    15  Clerk
    15  Mgr
    15  Sales
    20  Clerk
    20  Mgr
    20  Sales
```

Notice that DISTINCT eliminates all rows that contain duplicate data in the set of columns specified in the SELECT statement.

Using Expressions to Calculate Values

You can use an expression to produce the result you want to be returned. An *expression* is a calculation or function that you include in a statement. The following statement calculates what the salaries for each employee in department 38 would be if each received a $500 bonus:

```
SELECT DEPT, NAME, SALARY + 500
  FROM STAFF
  WHERE DEPT = 38
```

This statement produces the following result:

```
DEPT    NAME                 3
------  ---------   ------------
    38  Marenghi       18006.75
    38  O'Brien        18506.00
    38  Quigley        17308.30
    38  Naughton       13454.75
    38  Abrahams       12509.75
```

You can form arithmetic expressions using the following operators:

- **+**   for addition
- **–**   for subtraction
- *****   for multiplication
- ***/***   for division.

The operators can operate on values from several different types of operands, some of which are:

Column names (as in RATE * HOURS)
Constant values (as in RATE * 1.07)
Scalar functions (as in LENGTH(NAME) + 1).

A *scalar function* is a special type of expression that performs some operation on a value to return another value. The following statement returns the department names from the ORG table together with the length of each of these names:

```
SELECT DEPTNAME, LENGTH(DEPTNAME)
   FROM ORG
```

This statement produces the following result:

```
DEPTNAME                2
-------------   ------------
Head Office             11
New England             11
Mid Atlantic            12
South Atlantic          14
Great Lakes             11
Plains                   6
Pacific                  7
Mountain                 8
```

DB2 Server provides about 100 scalar functions and, in addition provides a user-defined function capability. Other examples of scalar functions provided by DB2 are:

YEAR   extract the year portion of a DATE value

CONCAT   append one character value to another (e.g.
            CONCAT(FIRST_NAME, LAST_NAME)

ABS   return the absolute value of a number

Naming Expressions

Using the AS clause with an expression, you are able to assign a meaningful name to the expression which makes it easier when referring back to it. The AS clause is optional in select-lists but can be used to provide a name for any item in the select-list.

The expression `SALARY + COMM` is named **PAY** in the following statement that displays all employees whose salary plus commission is known to be less than $13,000.

```
SELECT NAME,
       JOB,
       SALARY + COMM AS PAY
  FROM STAFF
  WHERE (SALARY + COMM) < 13000
  ORDER BY PAY
```

This statement produces the following result:

```
NAME        JOB            PAY
---------   -----   ------------
Yamaguchi   Clerk      10581.50
Burke       Clerk      11043.50
Scoutten    Clerk      11592.80
Abrahms     Clerk      12246.25
Kermisch    Clerk      12368.60
Ngan        Clerk      12714.80
```

Selecting from More Than One Table

You can use the SELECT statement to produce reports that contain information from two or more tables. For example, you can combine (*join*) data from the STAFF and ORG tables to form a new table. To join two tables, specify the table names in a FROM clause and the connection between them in a WHERE clause. The connection is based on data that is the same in both tables. The following example joins the NAME column from the STAFF table and the DEPTNAME column from the ORG table, choosing those rows where the data is the same in the ID column in the STAFF table and the MANAGER column in the ORG table:

```
SELECT NAME, DEPTNAME
  FROM STAFF, ORG
  WHERE MANAGER = ID
```

This statement produces the following result:

```
NAME        DEPTNAME
---------   --------------
Sanders     Mid Atlantic
Marenghi    South Atlantic
Hanes       New England
Plotz       Great Lakes
Fraye       Plains
Molinare    Head Office
Lea         Pacific
Quill       Mountain
```

which lists the name of each manager and their department.

Using a Subquery

When you write a SELECT statement, you can place another SELECT statement within the WHERE clause. Each SELECT after the first SELECT starts a phrase that is called a *subquery*.

You can produce a result in which the condition for selecting data is based on a table other than the one from which the data is selected. This is done by using a subquery in the WHERE clause. The following statement selects the division and location from the ORG table of the employee whose ID in the STAFF table is 280.

```
SELECT DIVISION, LOCATION
   FROM ORG
   WHERE DEPTNUMB = (SELECT DEPT
                        FROM STAFF
                        WHERE ID = 280)
```

When processing this statement, DB2 first determines the result of the subquery. That result is 66. Then the final result is taken from the row of the ORG table whose DEPTNUMB column has the value of 66. The final result is:

```
DIVISION   LOCATION
--------   -------------
Western    San Francisco
```

When you use a subquery, the database manager evaluates it and substitutes the value resulting from the subquery directly into the WHERE clause. The subquery can, in turn, include another subquery whose value is substituted into its WHERE clause. In addition, a WHERE clause can include subqueries in more than one search condition.

Using Column Functions

Column functions operate on many values in a column to derive a single result value. The following column functions are available:

AVG	returns the sum of the values in a set divided by the number of values in that set
COUNT	returns the number of values in a set of values
MAX	return the largest value in a set of values
MIN	return the smallest value in a set of values
SUM	returns the sum of the values in a set of values
STDDEV	returns the standard deviation of the values in a set of values
VARIANCE	returns the variance of the values in a set of values

The following statement selects the maximum salary from the STAFF table.

```
SELECT MAX(SALARY)
  FROM STAFF
```

This statement returns the value 22959.20 from the sample table.

Grouping Rows

You can use the GROUP BY clause to produce a result for each different value of the column in the GROUP BY clause. The following example produces a result that lists the maximum salary for each department number.

```
SELECT DEPT, MAX(SALARY) AS MAXIMUM
  FROM STAFF
  GROUP BY DEPT
```

This statement produces the following result:

```
 DEPT    MAXIMUM
------   --------
    10   22959.20
    15   20659.80
    20   18357.50
    38   18006.00
    42   18352.80
    51   21150.00
    66   21000.00
    84   19818.00
```

Creating Tables and Views

If your administrator has given you the authority to do so, you can create new tables and views.

Creating Tables

New tables are created by issuing an SQL statement of the general form:

```
CREATE TABLE  table-name
  ( column-name  data-type [NOT NULL]
   ,column-name  data-type [NOT NULL]
      ...
   ,column-name  data-type [NOT NULL] )
```

data-type   represents one of the many data types supported by DB2. Some of those types are:

        CHAR(*length*)   A character string whose length is always fixed as *length* characters.

        VARCHAR(*length*)   A character string whose length varies but is never longer than *length* characters.

        SMALLINT   A signed integer whose value may range from -32 768 to 32 767

        DECIMAL(*precision*,*scale*)   A decimal number whose maximum size (*precision*) is 31 digits. The *scale* part is used to specify the number of digits to be retained after the decimal place.

        DATE   A date.

        TIME   A time.

[NOT NULL]   is optional and may be specified to indicate that null values are not allowed in a column.

Refer to the *SQL Reference* for information on additional data types as well as for additional clauses on the CREATE TABLE statement.

The following statement creates a table named PERS which is like the STAFF table but has additional columns for date of birth and phone number.

```
CREATE TABLE PERS
  ( ID            SMALLINT        NOT NULL,
    NAME          VARCHAR(9),
    DEPT          SMALLINT,
    JOB           CHAR(5),
    YEARS         SMALLINT,
    SALARY        DECIMAL(7,2),
    COMM          DECIMAL(7,2),
    BIRTH_DATE    DATE,
    PHONE         SMALLINT )
```

This statement will create an empty table. We will see in the next section how data can be inserted into the new table.

Creating Views

New views are created by issuing an SQL statement of the general form:

```
CREATE VIEW  view-name
  [ ( column-name ,column-name ... ]
  AS fullselect
```

fullselect represents a SELECT with all of the power that we discussed above. The list of columns is optional and, if not used the views inherit the names of the columns in the *fullselect*.

The following statement creates a view of the non-managers in the STAFF table where salary and commission do not show through from the base table to the view.

```
CREATE VIEW STAFF_ONLY
  AS SELECT ID, NAME, DEPT, JOB, YEARS
    FROM STAFF
    WHERE JOB <> 'Mgr'
```

Note that the '<>' operator in the WHERE clause means 'not equal'.

If the following statement is issued against the new view:

```
SELECT * FROM STAFF_ONLY
```

then the results will be:

```
    ID   NAME        DEPT   JOB    YEARS
------   ---------   ------  -----  ------
    20   Pernal        20    Sales      8
    40   O'Brien       38    Sales      6
    60   Quigley       38    Sales      –
    70   Rothman       15    Sales      7
    80   James         20    Clerk      –
    90   Koonitz       42    Sales      6
   110   Ngan          15    Clerk      5
   120   Naughton      38    Clerk      –
   130   Yamaguchi     42    Clerk      6
   150   Williams      51    Sales      6
   170   Kermisch      15    Clerk      4
   180   Abrahams      38    Clerk      3
   190   Sneider       20    Clerk      8
   200   Scoutten      42    Clerk      –
   220   Smith         51    Sales      7
   230   Lundquist     51    Clerk      3
   250   Wheeler       51    Clerk      6
   280   Wilson        66    Sales      9
   300   Davis         84    Sales      5
   310   Graham        66    Sales     13
   320   Gonzales      66    Sales      4
   330   Burke         66    Clerk      1
   340   Edwards       84    Sales      7
   350   Gafney        84    Clerk      5
```

Earlier we joined the STAFF and ORG tables to produce a result that listed the name of each department and the name of the manager of that department. The following statement creates a view that can be repetitively used for the same purpose:

```
CREATE VIEW DEPARTMENT_MGRS
  AS SELECT NAME, DEPTNAME
       FROM STAFF, ORG
       WHERE MANAGER = ID
```

Once this view is created, the following SELECT statement:

```
SELECT * FROM DEPARTMENT_MGRS
```

would produce the result below:

```
NAME        DEPTNAME
---------   --------------
Molinare    Head Office
Hanes       New England
Sanders     Mid Atlantic
Marenghi    South Atlantic
Plotz       Great Lakes
Fraye       Plains
Lea         Pacific
Quill       Mountain
```

Other Data Definition and Data Control statements

In addition to the CREATE TABLE and CREATE VIEW statements there are many SQL statements that support the management of object definitions. A few of these statements are:

ALTER TABLE ...   Modify the definition of a table.

GRANT SELECT ON TABLE ...   Give someone the right to select information from a table.

DROP TABLE ...   Remove a table from the database.

CREATE DISTINCT TYPE   Create a new data type (e.g. MONEY) which can subsequently be used in table definitions.

Refer to the *SQL Reference* for information on these statements as well as for additional statements.

Using SQL Statements to Enter Data

When you define a new table, it does not contain data. The INSERT, UPDATE, and DELETE statements can be used to add, change, or delete data in the table.

Adding Data

To enter new rows into a table, use the INSERT statement. The INSERT statement has two general forms:

- With one form, you use a VALUES clause to specify values for the columns of one or more rows.

- With the other form, you include a SELECT statement to copy values from rows contained in another table or view.

In either case, for every row you insert, you will either supply a value for each column or accept the default value for certain columns.

The following statement uses a VALUES clause to insert one row of data into the STAFF table:

```
INSERT INTO STAFF
  VALUES (400, 'Harris', 20, 'Sales', NULL, 18000.66, 0)
```

The following statement uses a VALUES clause to insert three rows into the STAFF table where only the ids, the names, and the jobs are known. The rest of the values will be set to the null value.

```
INSERT INTO STAFF
  (NAME, JOB, ID)
  VALUES ('Swagerman', 'Prgmr', 500) ,
         ('Limoges', 'Prgmr', 510) ,
         ('Li', 'Prgmr', 520)
```

Notice that, in this case, VALUES were not specified for every column. In order to make this work, a list of column names was first specified and the values provided match the columns in that list. If the list of column names is omitted (as it was in the first example), the list of data values after VALUES must be in the same order as the columns in the table into which they are inserted, and the number of values must equal the number of columns in the table.

Each value must be compatible with the data type of the column into which it is inserted. If a column is defined as nullable and a value for that column is not specified, then the value NULL is given to that column in the inserted row. If a column is defined as NOT NULL and it does not have a default value, you must specify a value for it.

The following example inserts NULL into DEPT, YEARS, and COMM since values have not been specified for those columns in the row.

```
INSERT INTO STAFF (ID, NAME, JOB, SALARY)
  VALUES (410, 'Perna', 'Sales', 11100.11)
```

The NOT NULL clause on a column definition in a CREATE TABLE statement can be extended with the words WITH DEFAULT. If a column is defined as NOT NULL WITH DEFAULT, and that column is not specified in the column list, the default value is inserted to that column in the inserted row. The default values for the various data types are discussed in the *SQL Reference*.

The second form of the INSERT statement is very handy for populating a table with values from rows in another table. Rather than specifying VALUES, you specify a *fullselect* to identify columns from rows contained in other tables and/or views. The following example selects data from the STAFF table for members of the South Atlantic department (38) and inserts it into the PERS table:

```
INSERT INTO PERS (ID, NAME, DEPT, JOB, YEARS, SALARY)
  SELECT ID, NAME, DEPT, JOB, YEARS
    FROM STAFF
    WHERE DEPT=38
```

Following this insertion, the following SELECT statement:

```
SELECT ID, NAME, DEPT, JOB, YEARS, BIRTH_DATE, PHONE
  FROM PERS
```

would produce the result below:

```
    ID  NAME       DEPT  JOB    YEARS  BIRTH_DATE  PHONE   SALARY
------  ---------  ----  -----  -----  ----------  ------  ------
    30  Marenghi     38  Mgr        5        –         –    17506.75
    40  O'Brien      38  Sales      6        –         –    18006.00
    60  Quigley      38  Sales      –        –         –    16808.30
   120  Naughton     38  Clerk      –        –         –    12954.75
   180  Abrahams     38  Clerk      3        –         –    12009.75
```

Changing Data

The UPDATE statement allows you to change the data in a table. With the UPDATE statement, you can change the value of one or more columns in each row that satisfies the search condition of the WHERE clause. The result of the UPDATE statement is one or more modified column values in zero or more rows of a table (depending on how many rows satisfy the search condition specified in the WHERE clause). You can also use the UPDATE statement to delete a value from a nullable column (without removing the row) by changing the column value to NULL.

The following example updates information on the employee whose ID is 270:

```
UPDATE STAFF
  SET JOB='Prgmr',
      SALARY = SALARY + 300
  WHERE ID = 270
```

In this example, first the table (STAFF) was named then a `column-name = value-expression` pair was specified for each column that is to be updated. The employee whose employee number is 270 (Lea) has had their job title changed from Mgr to Prgmr, and the employee's salary has increased by $300.00.

You can change data in more than one row by including a WHERE clause that applies to two or more rows. The following example increases the salary of every salesperson by 15%:

```
UPDATE PERS
  SET SALARY = SALARY * 1.15
  WHERE JOB = 'Sales'
```

The SET clause specifies the columns to be updated and provides the formula for the new values.

The WHERE clause specifies the row or rows to be updated. If the WHERE clause is omitted, the database manager updates each row in the table or view with the values you supply.

Deleting Data

The DELETE statement allows you to delete a row or rows of data from a table. The following example deletes the row in which the employee ID is 320.

```
DELETE FROM STAFF
   WHERE ID=320
```

You can use the DELETE statement to delete more than one row. The following example deletes all rows in which the employee DEPT is 20:

```
DELETE FROM STAFF
   WHERE DEPT=20
```

When you delete a row, you remove the entire row. The DELETE statement does not remove specific columns from a row; it removes zero or more rows of a table (depending on how many rows satisfy the search condition specified in the WHERE clause).

If you omit a WHERE clause from a DELETE statement, the database manager removes all the rows of the table. (To delete the definition of a table as well as its contents, issue the DROP TABLE statement, described in the *SQL Reference*).

Chapter 10. Performing Administration Tasks

This section provides instructions on performing administration tasks. The topics covered include:

- Using the Database Director to manage database directories. See "Using Database Director."

- Using the command line processor to enter SQL statements or DB2 commands interactively. See "Using the Command Line Processor" on page 122.

- Using Visual Explain to tune SQL statements. See "Using Visual Explain" on page 123.

- Using Performance Monitor's Snapshot Monitor and Event Monitor to tune system performance. See "Using Performance Monitor" on page 124.

- Viewing configuration parameters. See "Viewing Configuration Parameters" on page 127.

- Using service tools to perform tasks such as receiving fixes, collecting diagnostic information, sending and receiving information, setting the trace utility, and setting the diagnostic levels. See "Using the Service Tools" on page 127.

- Using the product registration tool to become a registered user of the product. See Appendix C, "Product Registration" on page 157.

Using Database Director

The Database Director provides a graphical interface to DB2 objects and allows you to perform selected tasks such as:

- Configure databases and database manager instances

- Manage the directories necessary for accessing local and remote databases and instances

- Back up and recover databases or table spaces.

Start the Database Director by double clicking on the **Database Director** icon in the IBM DATABASE 2 folder.

Chapter 6, "Setting Up Communications for the Client and Server" on page 45 covers instructions on setting up the protocols using Database Director. More information is covered in the Getting Started windows in the online help. To view the help, follow these steps:

1 Double-click on the **Database Director** icon.

2 Click on the **Help** task.

3 From the help pull-down menu, click on **Getting Started**.

Using the Command Line Processor

The command line processor can operate in the following ways: interactive input mode, command line mode, or file input mode. The interactive and command line mode are covered here. See the *Command Reference* for information on the file input mode.

Interactive Input Mode

In the interactive input mode, you can enter command line processor commands and view the output from these commands.

To invoke the command line processor interactive input mode, click on the **command line processor** icon in the **IBM DATABASE 2** folder, or type db2 at a command prompt. When in interactive input mode, the prompt looks like this:

```
db2 =>
```

To end the command line processor interactive input mode, enter TERMINATE. The system returns to the operating system command prompt. When in this mode, you must NOT prefix commands with **db2**.

To execute operating system commands without quitting interactive mode, you can issue !<operating-system-command>.

Command Line Mode

To invoke the command line processor in command line mode, simply prefix a valid command line processor command with DB2, for example:

```
DB2 LIST NODE DIRECTORY
```

In the command line mode, you can execute individual command line processor commands from an OS/2 command prompt without entering interactive input mode.

To end the command line mode, enter TERMINATE. The system returns to the operating system command prompt.

Using Visual Explain

Visual Explain lets you view the access plan for explained SQL statements in graph format. You can then use the information available from the graph to find ways to tune your SQL queries for better performance.

The following steps enable you to create the examples as presented in the **Getting Started** information in the online help, and to perform basic tasks on the data.

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Start DB2** icon to start the database manager.

3 Enter DB2 CONNECT TO SAMPLE at a command prompt to connect to the sample database. (If you have not yet created the sample database, follow the steps in "Creating the Sample Database" on page 95.)

4 Enter the following command in the SQLLIB\MISC directory:

    DB2 -TF EXPLAIN.DDL

This command creates the explain tables that are based on sample queries.

5 Enter the following command in the SQLLIB\SAMPLES\VE directory:

    DB2 -TF VESAMPL.DDL

This command imports the predefined snapshots that are based on sample queries.

6 Double-click on the **Database Director** icon. The Database Director window opens.

7 Click on the [+] for the **database manager** where the SAMPLE database resides.

8 Click on the [+] for **Databases**.

9 Click on the [+] for **Sample**.

10 Double-click on the **Explained Statements History** icon to view the explained statements.

11 Select an explained statement, click on the **Statement** pull-down and choose **Show access plan** to view the graph of the explained statement. You can double-click on each node in the graph to view statistics about the various components.

12 Click on the **Statement** pull-down and choose **Show SQL text** to view the SQL statement that was explained. From here you can view the optimized SQL text to see how you can improve the performance or your queries.

Knowledge about how to best tune SQL queries is developed through experience over time. This information provides an overview on how to get started analyzing your SQL queries using Visual Explain. For detailed information, see the **Getting Started** information in the online help.

Using Performance Monitor

Performance Monitor provides the ability to report on hundreds of DB2 performance attributes. Some of the common attributes include connections, sort activity and I/O activity. You are able to view graphical summaries of the collected data or the complete details.

You can choose to monitor snapshots or events. "Snapshot Monitoring" allows you to capture point-in-time information at specified intervals. "Event Monitoring" allows you to record performance information over the duration of an event such as a connection.

Snapshot Monitoring: Snapshot monitoring is useful for monitoring database objects such as databases, database manager instances, database connections, tables, and table spaces. For example, using the snapshot monitor can help you find information on the total number of connections.

The following steps enable you to create a snapshot of the SAMPLE database. Information will be captured for each of the performance attributes every 30 seconds. We will look at a graph of the results as well as the details.

1 Double-click on the **Start DB2** icon to start the database manager.

2 At a command line, enter DB2 CONNECT TO SAMPLE. (If you have not yet created the sample database, follow the steps in "Creating the Sample Database" on page 95.)

3 Open the **IBM DATABASE 2** folder.

4 Double-click on the **Database Director** icon. The Database Director window opens.

5 Click on the [+] for the **database manager** where the database resides.

6 Click on the [+] for **Databases**.

7 Select **Sample** by clicking on the icon.

8 Click on the **Selected** pull-down and choose **Start monitoring** to begin monitoring the SAMPLE database. A standard set of information will be collected regarding the sample database.

9 From the Snapshot Monitor - Monitored Objects window, click on the **Snapshot monitor** pull-down. Click on **Open as settings** to change the capture interval time to 30 seconds.

10 Enter the following SQL statement at a command prompt:

```
DB2 SELECT * FROM EMPLOYEE ORDER BY LASTNAME
```

to generate some simple activity against the database.

11 From the Snapshot Monitor - Monitored Objects window, click on SAMPLE, then click on the **Selected** pull-down. Choose **Open as** and then **Performance details** to view the standard information that is collected for a snapshot taken of the database.

12 Scroll through the information until you see the database connection information. Double-click on the **Connection** entry to view detailed information on the database connections that were made. The number for **Total Connections** corresponds to the number of connections you made to the SAMPLE database.

Event Monitoring: Event monitoring gathers information about database events that occur during the running of an application. Event monitoring is useful for detecting deadlocks, overflow events, transaction completion, and application disconnections. For example, using the event monitor can help you find information on a slow-running application.

The following steps enable you to create an event and analyze the resultant data:

1 Double-click on the **Start DB2** icon to start the database manager.

2 At a command line, enter DB2 CONNECT TO SAMPLE. (If you have not yet created the sample database, follow the steps in "Creating the Sample Database" on page 95.)

3 Create an event monitor named test1 which will collect data for statements and connections by typing the following command at a command prompt:

```
DB2 CREATE EVENT MONITOR TEST1 FOR CONNECTIONS, STATEMENTS WRITE TO FILE 'E:\SQLLIB\EVENTS\'
```

4 Make sure that the directory you specify exists. If not, create it before activating the Event Monitor.

5 Activate the event monitor by typing the following command at a command prompt:

```
DB2 SET EVENT MONITOR TEST1 STATE=1
```

6 Generate activity against the database by opening several windows, connecting to the SAMPLE database, and issuing simple queries such as `DB2 SELECT * FROM EMPLOYEE`.

7 Turn off the event monitor by issuing the following command:

```
DB2 SET EVENT MONITOR TEST1 STATE=0
```

This forces the event monitor to write the trace file. If not turned off, a write only occurs once the buffer is full.

8 Activate the Event Analyzer to look at the trace information collected by performing the following steps:

a Open the **IBM DATABASE 2** folder.

b Double-click on the **Event Analyzer** icon. The Specify Parameter window opens.

c Enter the event files directory as specified in the create event monitor command. For our example: `E:\SQLLIB\EVENTS`. The Monitored Periods View opens.

d Select a **monitored period**, click on the **Selected** pull-down. Choose **Open as** and then **Connections**. This shows the number of connections you made.

e Click on the **Selected** pull-down and choose **Open as** and then **Statements**. This provides information on the SQL statements you issued during the connection.

As you go through this exercise, you will see many other options. Feel free to explore any of them. Read the *Getting Started* online help to see detailed information on each of the options. The online help also provides information on monitoring and analyzing your own applications.

Viewing Configuration Parameters

After you have installed the DB2 product, you can change DB2 configuration parameters to customize your node.

Configuration parameters typically govern how your computing resource is being used. Both client and server workstations have configuration parameters that can be changed.

These parameters do not typically need to be changed from the default values. You should work with the database administrator of the database server to determine which changes are necessary for your environment.

See the *Administration Guide* for a list of the configuration parameters.

To view the default configuration parameters, use the following instructions:

1 Open the **IBM DATABASE 2** folder.

2 Double-click on the **Database Director** icon. The Database Director window opens.

3 Click on the **database manager** that you want to work with.

4 Click on the **Selected** pull-down and choose **Configure**. The DB2 - Configure notebook opens.

5 Click on **Defaults** to set the parameters to the default values.

6 Advance through the pages of the notebook to view all the parameters. Click on **Help** to view detailed information about each of the parameters.

7 Click on **Cancel** to exit the notebook and to restore the values to their original values.

Using the Service Tools

Several service-related tools are packaged together in the **Service** folder (found within the IBM DATABASE 2 folder). Several of these tools require that you have an Internet connection established before you can use them. Here is a brief set of instructions to step you through the use of these tools.

Receive Fixes: With the Receive Fixes tool, you can FTP to an IBM Internet site and download available software fixes. Typically an IBM representative will make you aware of any fixes you might need, but you can also choose to browse periodically to see what is available. To use:

1 Double-click on the **IBM DATABASE 2** folder.

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

3 Double-click on the **Receive Fixes** icon.

4 The Internet address and remote directory for the FTP server are specified on the window. You do not need to change these values.

5 Click on the language in which you want to receive the Fixpak.

6 Click on the product for which you want to receive the Fixpak.

7 A second panel appears showing the fixes that are available for the product and language you have selected.

8 Ensure the **Local directory** is set for the location for the downloaded files.

9 Click on the **Fixpak** you need. A list of all files associated with the Fixpak appears. Choose the file you want to download. One of the files you can select is a readme file. It contains instructions for unpacking the files and descriptions of the problems fixed.

10 Click on **Get** to begin the transmission.

Collect Diagnostic Information: A tool to collect the files to send to IBM for diagnosing software problems. You must have a problem record open with IBM to use this function. Ensure that you have your problem record number available. To use:

1 Double-click on the **IBM DATABASE 2** folder.

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

3 Double-click on the **Collect Diagnostic Information** icon.

4 If you have more than one instance, select the instance you want to work with and click on **OK**.

5 Enter your problem record number. This is the number that is assigned to your problem when you open a problem with IBM.

6 Click on **OK** to proceed.

7 A panel with a default list of files appears. Click on **Add** to include more files to the list.

8 Click on **Pack** to collect and compress the files.

9 The next window allows you to send the package of files to IBM. Click on **Send** to send the information to IBM immediately.

Send or Receive Information: This tool allows you to send diagnostic information to IBM or receive information from IBM. To use:

 1 Double-click on the **IBM DATABASE 2** folder.

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

 3 Double-click on the **Send or Receive Information** icon.

 4 The Internet address and remote directory for the IBM FTP server are specified on the panel. You do not need to change these values.

 5 Click on **Send** for a panel where you can enter the name of the file you want to send to IBM. It is recommended that you use the **Collect Diagnostic Information** tool to collect and compress the files to send.

 6 Click on **Find** for a panel to help you find the file you to send.

 7 Find the file you want to send by clicking on the appropriate drive and directory where the file is located. Click on the file you want to send and click on **OK**.

 8 Click on **Send** once you have verified that the file is correct.

 9 From the main window, click on **Get** to view a list of files available from the IBM FTP site. An IBM Service Representative will place files for you to download at this location.

10 Select a file you would like to receive and click on **Get** to initiate the file transfer.

Trace Utility: Use the trace utility to collect trace information about DB2 activities. Once the trace information is collected, you can send it to IBM for analysis. To collect data during problem determination, perform the following steps:

 1 Double-click on the **IBM DATABASE 2** folder.

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

 3 Double-click on the **Trace Utility** icon.

 4 Select **Start** to activate the tracing of DB2 activities. Default trace parameters are already specified.

 5 Recreate the problem you are diagnosing by running through a typical sequence of steps.

 6 When you have finished recreating the problem, transfer the tracing information from memory to the file specified in **Dump to file** by selecting the **Dump** push button. Ensure that the file name for the dump file is your problem record number.

 7 Stop the tracing process by selecting the **Stop** push button.

 8 Select **Send** when you are ready to send the information to IBM. A panel appears with the Internet address and remote directory of IBM's FTP server. You typically do not need to change these values.

Set Diagnostic Levels: Use this tool to set the amount of diagnostic information generated by DB2 events as well as where the information is stored on your workstation. To use:

1. Double-click on the **IBM DATABASE 2** folder.
2. Double-click on the **Service** folder.
3. Double-click on the **Set Diagnostic Levels** icon.
4. For **Select a Diagnostic Level (DIAGLEVEL)**, click on the radio button that corresponds to the level of information you want recorded.
5. For **Diagnostic Path (DIAGPATH)** enter the fully-qualified path name of the location to store the diagnostic information.
6. Click on **OK** to set the level and path.
7. Click on **Exit** to close the panel.

Chapter 11. Creating Applications

This section provides instructions for creating applications that access DB2 Servers. The topics covered include:

- Using the DB2 Software Developer's Kit to build applications. See "Using DB2 Software Developer's Kit."

- Using the DB2 World Wide Web Connection to create macros containing HTML and SQL. See "Using DB2 World Wide Web Connection" on page 133.

Using DB2 Software Developer's Kit

There are two ways you can write applications that access DB2 databases:

- Embed static and dynamic SQL statements in your application.

- Code function calls to the DB2 Call Level Interface to invoke dynamic SQL statements.

DB2 also provides the following programming options:

- Application programming interfaces (APIs)

  Your application can call the DB2 APIs to perform functions such as backing up and restoring databases. Refer to the *API Reference* for more information about APIs.

- Stored procedures

  Your application, running on a client, can call procedures stored at the server using the SQL CALL statement. The procedures access the database and return information to your application. Your client application can be on one platform, while the stored procedure on the server can be on another platform. Refer to the *Application Programming Guide* for more information about stored procedures.

- User-defined Functions (UDFs)

  You can develop your own scalar functions that are specific to your database, and you can store them on the server. Your client application can then call the functions as required. You can use UDFs to enhance the use of User-defined Distinct Types (UDTs). Refer to the *Application Programming Guide* for more information about UDFs and UDTs.

Building Your Embedded SQL Application

When you embed SQL statements in your application, you must precompile and bind your application to a database, following the steps shown.

1 Create one or more source files with SQL statements.

2 Connect to a database, then precompile the source files. For each source file, precompiling produces a modified source file, and an access package in the database and/or a bind file. A bind file contains the data required to create an access package.

The access package contains an access plan for static SQL statements in your application. An access plan contains the information required by the database manager to access and manipulate data to perform an SQL statement.

To view an access plan, you can use Visual Explain from the Database Director. For more information about access plans, refer to the *Administration Guide*.

3 Compile the modified source files (and other files without SQL statements) using the host language compiler.

4 Link the object files with the DB2 and host language libraries, producing an executable program.

5 Bind the bind file to create the access package, if the package was not created at precompile time, or if a different database is going to be accessed.

6 Run the application. The application accesses the database.

Using the DB2 Call Level Interface

This section provides a general overview of DB2 Call Level Interface (DB2 CLI) applications. Refer to the *Call Level Interface Guide and Reference* for more information.

The DB2 CLI provides another way to access a database using SQL that is an alternative to embedded SQL statements.

To access a database, you code function calls to the DB2 CLI to invoke dynamic SQL statements. The DB2 CLI passes the SQL statement to the database manager for processing.

Building Your DB2 CLI Application

You do not need to precompile or bind DB2 CLI applications because they use a common access package provided with DB2. This makes DB2 CLI applications

portable because they do not depend on a particular precompiler, and because they can access any DB2 database using a common package.

You compile and link your application using the compiler for your programming language.

Using DB2 World Wide Web Connection

The information in this section provides an overview of DB2 World Wide Web Connection. For guidance on building applications known as *macro files* that allow access to DB2 data, read the *DB2 World Wide Web Connection Application Developer's Guide*. To view or print this book, see "Viewing .INF Information Files" on page 143 and "Printing the PostScript Books" on page 144 for information.

Before You Begin

It is recommended that you install a Web Server, such as the IBM Internet Connection Server, before installing DB2 World Wide Web Connection. If you have installed DB2 World Wide Web Connection first, consider reinstalling in the recommended order. Otherwise, you must copy several files from DB2 World Wide Web Connection directories to IBM Internet Connection Server directories, as follows:

```
XCOPY c:\sqllib\db2www\cgi-bin\*.* c:\www\cgi-bin\ /s /e
XCOPY c:\sqllib\db2www\html\*.* c:\www\html\ /s /e
XCOPY c:\sqllib\db2www\icons\*.* c:\www\icons\ /s /e
```

Where c:\sqllib\db2www\ is the default DB2 World Wide Web Connection directory and c:\www\ is the default IBM Internet Connection Server directory.

Make sure you use the XCOPY command with the /s and /e options. This will ensure that all the proper files and directories are copied to the correct location.

Starting the Demo

DB2 World Wide Web Connection provides a sample application for you to use in order to familiarize yourself with this type of application. To use the demo, follow these steps:

1 Double-click on the **Install DB2 World Wide Web Connection Demo** icon located in the DATABASE 2 folder. This installs the sample applications. This step can only be performed on a system with DB2 Server installed.

2 Start the Web Server. For example, start the IBM Internet Connection Server by double-clicking on the IBM Internet Connection Server icon.

3 Use a web browser, such as IBM's Web Explorer, to open the URL:

```
http://machine_name/celdemo.htm
```

Where `machine_name` is the fully-qualified name of the system with DB2 World Wide Web Connection installed.

A series of sample applications and macros are available through celdemo for you to try and to learn from.

Invoking DB2 World Wide Web Connection

DB2 WWW Connection is a common gateway interface (CGI) program that you can invoke from an HTML anchor reference or form.

If you plan to run queries that take a long time to complete, consider changing your default timeout for CGI programs. Usually the default timeout is five minutes, which is enough for most queries. See your Web server's documentation for specific instructions.

- Anchor reference

  ```
  <A HREF=http://{web-server}/cgi-bin/db2www(.exe)/{macro-file}/
  {cmd}[?name=val&...]>any text</A>
  ```

- HTML form

  ```
  <FORM METHOD={method} ACTION=http://{web-server}/cgi-bin/
  db2www(.exe)/{macro-file}/{cmd}[?name=val&...]>any text
  ```

{web-server}   This is the name of the web server as defined by the system administrator. For example, *www.ibm.com*. If the macro exists locally, you only need the relative URL, omitting *http://www.ibm.com/*.

{macro-file}   This is the name of the macro file as defined by the DB2 WWW Connection application developer. For example, *salesfrm.d2w*.

{cmd}   You can specify either *input* or *report*. If *input*, an HTML input form is generated. People accessing the form through a Web browser can fill in values or select options. If *report*, the HTML in the report section is displayed and specified SQL queries are called from %EXEC_SQL statements.

{method}   You can specify either *get* or *post*, which are specified in HTML 2.0. Using the get method is not recommended because of its limitations.

?name=val&...   These are optional parameters that may be passed in your applications. For example, you can pass the user ID so it doesn't have to be entered multiple times or you can pass the name of a macro that calls another macro so you can back out. This can also be used to call other DB2 WWW macros based on the results of a query.

A typical URL with optional parameters looks like this:

```
http://www.abc.com/cgi-bin/db2www/stdqry1.d2w/input?field=custno&field=custname
```

Anchor References

The anchor references are commonly called *hyperlinks* or simply *links*. Using links, you specify the macro name and the *cmd* to invoke a SQL query. You can use this method to make queries available that require no HTML form input, sometimes referred to as *canned queries*.

You can place text or graphics in the link. When people click on the linked text or graphic, they invoke the macro. For example:

- Show what's new this season
- Show 10 best sellers
- Show items under $200

This is an example of a canned query that you can invoke from a Web page using a link like this:

```
<A href="http://www.ibm.com/cgi-bin/db2www.exe/monlstA.d2w/report">
List all monitors</A>
```

The link calls this macro:

```
%DEFINE DATABASE="MNS95"

%SQL{
SELECT MODNO, COST, DESCRIP FROM EQPTABLE
WHERE TYPE='MONITOR'
%}

%HTML_REPORT{
%EXEC_SQL
%}
```

This query returns a table of all the monitors sold, including the model number, cost, and description for each one. You can use HTML to format the output as you like in a SQL report subsection. This macro does not need an HTML input section because no variables are used in the query.

HTML Forms

The form method exploits the power of HTML forms so your customers can get exactly the information they want. You create the form with any ASCII editor, although there are several good HTML editors available commercially. A form can be as simple as a short list of items or as complex as you need, including Boolean operators and beginning and ending dates. You can also use your Web browser to look at HTML code in forms you find on the WWW with your Web browser's view feature to see how others use forms.

"Connecting to the Internet Using DB2 World Wide Web Connection" on page 12 lists some places on the WWW where you can learn more about HTML and writing HTML forms.

For most transactions you will find it useful to call an input section before making the query. This example is similar to the monitor list example above, but allows the application user to select which product to view. The macro can be called from a URL like this:

```
<A href="http://www.ibm.com/cgi-bin/db2www.exe/equiplst.d2w/input">
List of hardware</A>
```

Here is the macro the link calls:

```
%DEFINE DATABASE="MNS95"

%HTML_INPUT{
<H1>Hardware Query Form</H1>
<FORM METHOD="POST"
      ACTION="/cgi-bin/db2www.exe/equiplst.d2w/report">
<dt>What hardware do you want to list?
<dd><input type="radio" name="hdware" value="MON" checked>Monitors
<dd><input type="radio" name="hdware" value="PNT">Pointing devices
<dd><input type="radio" name="hdware" value="PRT">Printers
<dd><input type="radio" name="hdware" value="SCN">Scanners
</dl>
<input type=submit value=Submit>
</FORM>
%}
```

```
%SQL{
SELECT MODNO, COST, DESCRIP FROM EQPTABLE
WHERE TYPE=$(hdware)
%SQL_REPORT{
<B>Here is the list you requested:</B><BR>
%ROW{
<HR>
$(N1): $(V1)     $(N2): $(V2)
<P>
$(V3)
%}
%}
%}

%HTML_REPORT{
%EXEC_SQL
%}
```

The form appears on the Web browser as shown in Figure 6.

Figure 6. An HTML Form Using List Variables

After a selection is made and the Submit button selected, the HTML report section is called. The HTML report section calls the SQL section and the query is processed, substituting the value specified in the form for $(hdware). The query results are displayed according to the HTML in the SQL report subsection.

Chapter 12. Obtaining Additional Information

Additional information on the Database Server product is available in several different formats. The following formats are available:

- Online Help
- Interactive Presentation Facility (`.INF`)
- PostScript (`.PS`)

All books described in this chapter are also available in hardcopy by ordering from IBM. To order, 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.

How the Database Server Library Is Structured

The Database Server library consists of online help and manuals. The tables that follow outline the information that is available and how you can access it.

Online Help: Online help is available with all Database Server components. The following table gives details on the various type of help available and how to access each.

Type	Provides...	How to Access...
Online Help		
General Help	A list of the commands that are available.	To get General Help, press F1 or the HELP push button. This will provide general or field help for the component you are currently using.
		To get General Help in the command line processor, type ? from interactive input mode.
SQL Help	Help text for the SQL command syntax.	To get SQL Help in the command line processor, type `help statement` from interactive input mode, where `statement` is an SQL statement such as SELECT.
Syntax Help	Help text for the command line processor command syntax.	To get Syntax Help in the command line processor, type ? `phrase` from interactive input mode, where `phrase` is a command line processor command.

Type	Provides...	How to Access...
SQLSTATE Help	Help text for SQL states and class codes.	To get SQLSTATE help from command line processor interactive mode, issue the commands `? sqlstate` or `? class-code` where `sqlstate` is a valid five digit SQL state and `class-code` is a valid two digit class code. For example, `? 08003` and `? 08`.
Command Help	Help information on specific commands.	To get help on a command, type `? command` where `command` can be the first few keywords. For example, `? catalog database` brings up help for the CATALOG DATABASE command, whereas `? catalog` brings up help for all the CATALOG commands.
Message help	Help in understanding DB2 messages.	Describes the cause of the message and any action that should be taken to solve the problems. To get Message Help from command line processor interactive mode, issue the command: `? message number`, where `message number` is a valid command line processor or SQL message number. For example, `? SQL30081N` provides help on the `SQL30081N` message. To view message help one screen at a time, type the following command: `echo ? XXXnnnnn	db2 +p	more` To save the message help in a file while in this mode, type the following command: `echo ? XXXnnnnn	db2 +p > filename.ext` where `XXX` represents the message prefix and `nnnnn` represents the message number.
Graphical Interface help	Help for those tools that have a graphical user interface.	It provides information specific to the window that you are using. (From the window, press the Help key on your keyboard, or select the **Help** push button if it is available.)
Getting Started Help	Instructions on using the graphical panels.	From a Database Director primary window, select **Getting Started** from its **Help** pull-down. Getting Started Help is provided with Database Director, Visual Explain, and Performance Monitor.

Hardcopy Information: The hardcopy library that is available for Database Server components is listed in the following table. You can view each of these books online, print them on PostScript printers, or order the published book through IBM.

Table 7 (Page 1 of 3). Hardcopy Documentation

Form Number File Name	Book Name	Book Description
		Cross-Platform Books
S20H-4664 SQLG0x02	*Information and Concepts Guide*	Product and conceptual information to anyone who needs a comprehensive overview of the DB2 products. It is useful when deciding which DB2 products suit your environment. It also includes a glossary of terms used in the book.

Table 7 (Page 2 of 3). Hardcopy Documentation

Form Number File Name	Book Name	Book Description
S20H-4580 SQLD0x03	*Administration Guide*	Contains information required to design, implement, and maintain a database to be accessed either locally or in a client/server environment.
S20H-4871 SQLF0x02	*Database System Monitor Guide and Reference*	Includes a description of how to use the Database System Monitor and a description of all the data elements for which information can be collected.
S20H-4645 SQLN0x03	*Command Reference*	Reference information needed to use system commands and the DB2 for OS/2 command line processor to execute database administrative functions. Describes the commands that can be entered at an operating system command prompt or in a shell script to access the database manager. Explains how to invoke and use the command line processor, and describes the command line processor options. Provides a description of all the database manager commands.
S20H-4984 SQLB0x03	*API Reference*	Provides information about the use of application programming interfaces (APIs) to execute database administrative functions. Presents a description of APIs and the data structures used when calling APIs, as well as detailed information on the use of database manager API calls in applications written in the supported programming languages.
S20H-4665 SQLS0x03	*SQL Reference*	Is intended to serve as a reference for syntax and rules governing the use of SQL statements. Syntax diagrams, semantic descriptions, rules and examples are provided for the SQL statements. Catalog views, product maximums, release-to-release incompatibilities, and a glossary are also included in this book.
S20H-4643 SQLA0x03	*Application Programming Guide*	Discusses the application development process and how to code, compile, and execute application programs that use embedded SQL to access the database. It includes discussions on programming techniques and performance considerations for the application programmer.
S20H-4644 SQLL0x03	*Call Level Interface Guide and Reference*	Is a guide and reference manual for programmers using the Call Level Interface. DB2 Call Level Interface is a callable SQL interface based on the X/Open CLI specification and is compatible with Microsoft Corporation's ODBC.
S20H-4808 SQLM0x03	*Messages Reference*	Lists messages and explanations. Each explanation includes the action to be taken when a message or code is issued.
S20H-4779 SQLP0x02	*Problem Determination Guide*	Provides information that helps in determining the source of errors, recovering from problems, and describing and reporting defects.
S20H-4793 SQLC0x02	*DDCS User's Guide*	Provides concepts, programming and general information about the DDCS products.
Platform Specific Books		
S20H-4784 SQLG2x02	*DB2 for OS/2 Planning Guide*	Provides product planning information that is useful when determining which products need to be purchased for a particular environment. Lists all the features that are new to Version 2.
S20H-4785 SQLI2x03	*DB2 for OS/2 Installation and Operation Guide*	Contains information required to install and use the database manager and supplied tools in your operating environment. This manual shows you how to set up your server to support remote clients.

Table 7 (Page 3 of 3). Hardcopy Documentation

Form Number File Name	Book Name	Book Description
S20H-4788 SQLT2x02	*DB2 for OS/2 Master Index*	Lists the combined index entries of the manuals, indicating where the information can be found.
S20H-4782 SQLJ2x03	*Installing and Using OS/2 Clients*	Provides installation and setup information for the DB2 Client Application Enabler and DB2 Software Developer's Kit products. Also includes general usage information for the DB2 Client Application Enablers.
S20H-4789 SQLJWx03	*Installing and Using DB2 Clients for Windows*	Provides installation and setup information for the DB2 Client Application Enabler and DB2 Software Developer's Kit products. Also includes general usage information for the DB2 Client Application Enablers.
SC09-1628 SQLJDx01	*DB2 Client Application Enabler for DOS User's Guide*	Provides installation and setup information for the DB2 Client Application Enabler and DB2 Software Developer's Kit products. Also includes general usage information for the DB2 Client Application Enablers.
S20H-4666 SQLJ3x03	*Installing and Using DB2 Clients for Windows*	Provides installation and setup information for the DB2 Client Application Enabler and DB2 Software Developer's Kit products. Also includes general usage information for the DB2 Client Application Enablers.
S20H-4787 SQLA2x02	*DB2 SDK for OS/2 Building Your Applications*	Provides environment setup information and step-by-step instructions to compile and link DB2 applications on an OS/2 system.
S20H-4795 SQLC2x03	*DDCS for OS/2 Installation and Configuration Guide*	Provides installation and setup information for the DDCS product.
DB2 World Wide Web Connection		
DB2W0x00	*DB2 World Wide Web Connection Application Developer's Guide*	Provides information to help you to build Web applications that access DB2 databases by using HTML forms and dynamic SQL. These applications are then stored on the Web server, and the end user sees only the HTML forms and the reports returning DB2 data.

- The x in the sixth position of the filename corresponds to the language you are using. For example, English books have an E in the sixth position. The English version of the *Information and Concepts Guide* has the filename SQLG0E02.

- For additional information that is not found in the publications, see the **README** file. Each product includes a README file which can be found in the directory in which the program files are installed. For more information, refer to the installation information that comes with the product.

Ordering the Cross-Platform Printed Manuals

You can also order the cross-platform printed manuals. The part number for this set of books is 41H2451. The books included in this package are listed in Table 7 on page 140 under the heading **Cross-Platform Books**.

Viewing .INF Information Files

To view the books that are provided in OS/2 Information Format, follow these steps:

1 Open the **IBM DATABASE 2** folder and click on the **DB2 information** icon.

2 Double-click on the icon for the book you want to view. Each book for the product is represented by a separate icon.

The **Documentation** component must be selected during installation in order for you to view the .INF books.

Once inside a book, you are able to hyperlink to the other books in the library. To search on specific information, click on the **Search** push button and type your search criteria. A panel displays all of the search matches. Click on the one you want to view.

If you want to launch the book from an OS/2 window, type `VIEW` and the book name. The book names have the file extension `.INF`.

If you want to view the books directly from the CD-ROM, the files are located in the `X:\<LANGUAGE>\BOOKS\INF` directory.

See "How the Database Server Library Is Structured" on page 139 for a list of the manuals that are shipped on the Database Server CD-ROM.

Printing the PostScript Books

If you prefer to have printed copies of the manuals, you can decompress and print PostScript versions by following these steps:

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:\<LANGUAGE>\BOOKS\PS` directory.

2 Decompress the file by running the book name. The result from this step is a printable PostScript file with a file extension of `.PS`.

3 Ensure that your default OS/2 printer is a PostScript printer capable of printing Level 1 (or equivalent) files.

4 Enter the following command from an OS/2 command line:

```
PRINT FILENAME.PS
```

The PostScript manuals have a file extension of .EXE (until they are decompressed). See "How the Database Server Library Is Structured" on page 139 for the list of book titles and file names.

Ensure you specify the entire filename of the file you intend to print.

Part 4. Appendixes

Appendix A. Worksheets for Use with DDCS

This appendix contains the following worksheets for use when planning the installation of DDCS on your IBM software servers server.

- "MVS, VSE, and VM Server Worksheet" on page 148
- "OS/400 Server Worksheet" on page 150
- "OS/2 Configuration Worksheet" on page 151
- "OS/2 Connection to Server Worksheet" on page 152

For information about how to use the worksheets, see Chapter 7, "Configuring DDCS" on page 71.

MVS, VSE, and VM Server Worksheet

Item	Parameter	Example	Your value	Note
1	NETID	SPIFNET		
2	Controller address	400009451902		
3	MODEENT	IBMRDB		
4	SSCP name	NYX		
5	APPL	NYM2DB2		
6	LU	NYX1GW01		1
7	PU	NYX1		1
8	IDBLK and IDNUM	05D 27509		1
9	Remote transaction program	(default)		
10	Database location/name	NEW_YORK3		

1 The PU, IDBLK, and IDNUM (items 7 and 8) must be the same for all DRDA server systems that DDCS is connected to. The LU can also be the same.

For the DDCS administrator:

1 If you have already defined a connection to another DRDA server, write the LU, PU, and IDBLK and IDNUM values that you used in the table above.

2 If this is your first DRDA server connection, look at the Local Node Characteristics panel and write the local node name for your machine beside PU in the table above.

For the VTAM administrator:

DDCS gives applications on remote systems (such as OS/2) access to data in your DB2 system. In order to make this possible, you must define DDCS to VTAM as follows:

1 Create an independent LU and a switched major node for the DDCS system. Use the PU value that the DDCS administrator wrote in the table above. If the LU, IDBLK and IDNUM values are already written in the table, use these values also.

2 If the database is not already set up for remote communication in VTAM, set it up.

3 Write the following information for the DDCS administrator in the table:
  - The name of the network (NETID)
  - The locally-administered address or universal address of the communication controller
  - The entry in the mode table (MODEENT) used for communication. We recommend that you use mode IBMRDB, which is described in the *DRDA Connectivity Guide*.
  - The SSCP name
  - For the database, the APPL value
  - For the independent LU corresponding to the DDCS system, the LU name
  - For the switched major node corresponding to the DDCS system, the IDBLK and IDNUM values.

4 Give the worksheet to the database administrator.

For the database administrator:

DDCS gives applications on remote systems (such as OS/2) access to data in your DB2 system. In order to make this possible, you must perform the following steps:

1 Set up the database for remote communication. (For MVS, this includes adding entries to the SYSIBM.SYSLUNAMES and SYSIBM.SYSUSERNAMES tables). As an LU name, use the value written for "LU" in the table.

2 If you are not using the default TP name, write the value that you are using in the table beside Remote transaction program. For MVS systems, the default (defined by DRDA) is X'07'6DB. For VSE and VM systems, the default is the RDB_NAME.

3 For MVS, write the LOCATION value in the last line of the table. For VSE or VM, write the RDB_NAME value in the last line of the table.

4 Give the worksheet to the DDCS administrator.

OS/400 Server Worksheet

Item	Parameter	Example	Your value	Note
1	Local network ID	SPIFNET		1
2	Local adapter address	400009451902		2
3	Mode name	IBMRDB		3
4	Local control point	SYD2101A		4
5	Remote transaction program	(default)		5
6	Relational database name	NEW_YORK3		6

For the AS/400 administrator:

DDCS gives applications on remote systems (such as OS/2) access to data in your DB2 for OS/400 system. In order to make this possible, the DDCS administrator needs the following information:

1 The local network ID. You can get this information by entering `DSPNETA`.

2 The local adapter address. You can get this information by entering `WRKLIND (*TRLAN)`.

3 The mode name. You can get a list of mode names by entering `WRKMODD`. If the mode IBMRDB has been defined on your OS/400 system, we recommend that you use it.

4 The local control point. You can get this information by entering `DSPNETA`.

5 The remote transaction program name. The default (defined by DRDA) is `X'07'6DB`.

6 The relational database name. You can get this information by entering `DSPRDBDIRE`.

OS/2 Configuration Worksheet

Item	Parameter	Example	Your value	Note
1	Network ID	SPIFNET		1
2	Local node name	NYX1		2
3	Local node ID	05D 27509		3
4	Mode name	IBMRDB		4
5	Local LU name	NYX1GW01		5
6	Local LU alias	NYX1GW01		

1 The Network ID matches item 1 on the server worksheet.

2 For MVS, VSE, or VM, the Local node name matches item 7 (PU) on the server worksheet. For OS/400, any unique node name can be used; in most cases, this value has already been set for you.

3 For MVS, VSE, or VM, the Local node ID matches item 8 (IDBLK and IDNUM) on the server worksheet. For OS/400, you can leave this blank.

4 The Mode name matches item 3 on the server worksheet.

5 The Local LU name matches item 6 (LU) on the MVS, VSE, and VM server worksheet.

OS/2 Connection to Server Worksheet

Item	Parameter	Example	Your value	Note
7	Link name	LINKHOST		
8	LAN destination address	400009451902		1
9	Symbolic destination name	DB2CPIC		
10	Partner CP name	NYX		2
11	Partner LU name	NYM2DB2		3
12	Remote transaction program	(default)		4
13	Target database name	NEW_YORK3		5

1 The LAN destination address matches item 2 on the server worksheet.

2 The Partner CP name matches item 4 on the server worksheet.

3 For MVS, VSE, or VM, the Partner LU name matches item 5 (APPL) on the server worksheet. For OS/400, you can use the partner CP name.

4 The Remote transaction program is copied from the server worksheet.

5 The Target database name matches the last item on the server worksheet.

Appendix B. Remote CID Installation Using SystemView

You can use SystemView to install the following Database Server products on unattended workstations:

- DB2 Server
- DB2 Software Developer's Kit
- DB2 World Wide Web Connection
- DDCS
- DB2 Client Application Enabler for OS/2
- Administrator's Toolkit for OS/2

SystemView uses the statements included in two files called a *software profile* and a *response file*. A software profile is used by SystemView to invoke an application installation program. A response file is used by the installation program to get all of the data it needs during the installation process.

Using SystemView, you specify the information for the software profile and response files when you customize the application definition file, which defines how to install and configure Database Server products. This file is shipped with Database Server.

Refer to the SystemView documentation for a complete description of software profiles and response files.

Overview

To install the appropriate Database Server products on remote workstations using SystemView, you need to:

- Have *SystemView LAN Client* installed on your workstations. This is provided by the OS/2 Warp Server product.

- Copy the Database Server product files to an OS/2 code server

- Add Database Server product information to the SystemView software library

- Configure Database Server products in the SystemView software library

- Distribute and install the resulting software object using SystemView on target server workstations.

These tasks are described in the following sections. If you need additional assistance
to perform these tasks, use the online help or refer to *SystemView Up and Running!* for
more information.

Copying Database Server Products to the Code Server

To set up the code server so that you can install Database Server products using
SystemView, enter the XCOPY command.

For example, to copy the files from the Client Pack CD-ROM located in the E: drive to
the D: drive, you would enter this command:

```
XCOPY E:\OS2\EN\*.* D:\CID\IMG\DB2CLI. /S
```

This command copies the files from the \OS2\EN directory on the Client Pack CD-ROM
in the E: drive to the D: drive, putting the files under the directory \CID\IMG\DB2.

The /S indicates that all of the subdirectories and their contents and the contents of the
root directory should be copied.

The files are found in the E:\EN directory on the Database Server CD-ROM
and the copy command would be:

```
XCOPY E:\EN\*.* D:\CID\IMG\DB2SRV. /S
```

Adding Database Server Products to the Software Library

To add Database Server products to the software library, do the following steps:

1 Start SystemView.

2 Double-click on the **SystemView Service Manager** icon.

3 Double-click on the **CID Software Preparation** icon.

4 Double-click on the **Software Library** icon.

5 Select **Software** from the menu bar on the Software Library window.

6 Select **New...**.

7 Enter the name of the product you are adding.

8 Select **Find...**.

9 In the Find definition file window, select:

- The drive letter, where the CD-ROM is located, in the **Drive** list
- EN*.ADF in the **Directory** list
- DB2SRVR.ADF in the **File** list

10 Click on **OK** when finished.

11 Select **Add** to associate the application definition file with Database Server.

This copies the remote installation files (.ADF, .VAR, .CMD and .MRF) from the CD-ROM to the subdirectory where SystemView resides (\SWLIB).

An icon is created in the Software Library folder that represents the product you have just added.

Creating Configurations for Database Server Products

To configure Database Server products, do the following steps:

1 Double-click on the Database Server product icon in the Software Library window.

2 Select **Configuration** from the menu bar on the Software Configuration window and then select **New...** from the pull-down window.

The Settings notebook is displayed. You need to define how the Database Server product will be installed on the target workstation.

3 Enter a unique identifier to define the configuration you are creating.

4 On each page of the notebook, select each of the items in the **Variables** list and choose the **setting**. A description of each variable is displayed in the **Explanation** field of the notebook page.

5 When you have completed all of the pages of the notebook, click on **OK**. If you are ready to distribute the software, select **Yes** when asked **Do you want to catalog the database?** A response file is created based on values in the notebook.

Here is an overview of just a few of the pages you will fill in.

Configuration Page

Use to define both an identifier needed to catalog Database Server uniquely and a name that is a short description for the configuration you are creating. For example, the identifier may be `EUROSERV` and the name may be `Definition for the servers in Europe.`

Code Server Page

Use to associate the configuration with the redirection mechanism to be used during CID installation. Define the code server aliases and the CID directory aliases that let the target workstation (where you want to install Database Server products) access the code server images. You also indicate whether an existing or new response file will be used.

Response File Page

Use to specify the name and directory where the existing response file is located.

The remaining pages of the Settings notebook configure Database Server products at the target workstation. SystemView uses the information from the notebook to create the settings for Database Server variables in the response file. Refer to the *Installation and Operation Guide* for more information about response file keywords.

Installing Database Server Products on a Remote Server Workstation

To install Database Server products on a remote workstation, you can either use the Event Scheduler or the Remote System Manager. You can find information on how to use the Event Scheduler and the Remote System Manager in *SystemView Up and Running!* This book also contains information about changing variables at installation time.

Appendix C. Product Registration

Database Server provides an electronic registration tool which is used to register user information. When you become a registered user, you will:

- Receive notification of product enhancements
- Be in line to take advantage of upgrades and promotions
- Receive a subscription to the IBM DB2 Family Newsletter for ongoing hints and tips for using the DB2 products.

The registration tool is invoked from the "System Setup" inside the OS/2 System folder or will be invoked automatically during installation of the server.

Having filled in the user information and the answers for product specific questions, you will have the choice of sending the registration information electronically using a modem, if present, or by printing to send the information via a fax machine or postal service, or by calling a toll-free number and giving the information to the operator. At the completion of the registration process, you will be issued a Registration Number. If the registration information is provided via a modem, the number will be stored automatically. Otherwise, you need to enter the number through the registration tool to indicate the process is complete.

If the registration is not complete, the registration tool will be automatically invoked during DB2START. You will be reminded on an 8-day interval until the registration information is provided and the registration number is entered. At the first and second anniversaries of the initial registration, the registration tool will be started to remind you to register again so that you will be informed of any updates to the Database Server product.

Considerations for Remote Installations

In a production environment involving remote servers, you may want to suppress the automatic invocation of the registration tool. Examples of when this might be true include:

- The DB2 Server software will be installed on remote systems using the SystemView CID process.

- The DB2START command will be issued by an automation tool and the server console is unattended.

- The DB2START command will be issued by users at the remote site who should not be involved in the registration process.

In these cases you should:

1 Register your central site as a database server user. If you have not already registered, follow the registration instructions which you will be presented with when you install the database server at your central point. Registration of your central point as a database server user will provide your company with all registration benefits.

2 Add a `SET DB2NOREG=YES` statement in the remote server's CONFIG.SYS file. This suppresses the invocation of the automatic registration tool. In the case of the SystemView CID remote installation process, this statement will be added to the CONFIG.SYS when the database server is installed if it is defined in the CID response file. See Appendix B, "Remote CID Installation Using SystemView" on page 153.

Entering Your Product Registration Number

If you have sent your registration to IBM using a fax machine or the postal service, you will be supplied with a registration number once IBM receives your registration. When you receive the number, you can invoke the product registration tool by double-clicking on the elephant icon in OS/2's System Setup folder. Indicate which product the registration number is for by clicking on the server name in the window on the first screen.

Appendix D. Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

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

> IBM Director of Licensing
> IBM Corporation
> 500 Columbus Avenue
> Thornwood, NY 10594
> USA

For license inquiries regarding double-byte character set (DBCS) information, contact:

> IBM World Trade Asia Corporation Commercial Relations
> IBM Kamiya-cho Building, 4-3-9 Toranomon, Minato-ku
> Tokyo, Japan

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.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

Trademarks

The following terms are trademarks or registered trademarks of the IBM Corporation in the United States and/or other countries:

AIX	IBM
AS/400	MVS/ESA
DATABASE 2	Operating System/2
DB2	OS/400
DB2 for AIX	OS/2
DB2 for MVS/ESA	SQL/DS
DB2 for OS/400	SQL/400
DB2 for OS/2	SystemView
DB2 for VSE and VM	VisualAge
Distributed Database Connection Services	Visualizer
Distributed Relational Database Architecture	VM/ESA
DRDA	VSE/ESA
FFST/2	

Windows is a trademark of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Index

W

WHERE clause
worksheets

Contacting IBM

This section lists a number of ways you can reach IBM for various reasons. Depending on the nature of your problem/concern, we will ask you to be prepared to provide us with some information to allow us to serve you better.

If you have a technical problem, please take the time to review and carry out the actions suggested by the *Problem Determination Guide*. To request IBM technical service and support for product defects after reviewing the *Problem Determination Guide*, call one of the following numbers to learn about the free and chargeable service options available:

- 1-800-237-5511 for all IBM products.
- 1-800-992-4777 for OS/2 and DOS products.
- 1-800-CALL-AIX for AIX products.

Both of these numbers are for the United States and Canada only. Outside these countries, contact your local IBM Software Support Center.

Note: In some countries, IBM-authorized dealers should contact their dealer support structure in place of the IBM Support Center.

For availability or to order any of the common server products contact an IBM representative at a local branch office; or, contact any authorized IBM software remarketer; or, contact IBM directly by phoning one of the following numbers:

- 1-800-IBM-CALL or 1-800-3IBM-OS2 in the United States.
- 1-800-IBM-4YOU or 1-800-IBM-CALL in Canada.

To order publications, use one of the following numbers:

- 1-800-879-2755 in the United States
- 1-800-IBM-4YOU in Canada.

If you wish, you can also reach IBM through the World Wide Web, CompuServe**, or Internet. The addresses are as follows:

World Wide Web   **http://www.software.ibm.com**

                                       In the "Data Management" section, current information such as DB2 news, product descriptions, education schedules, frequently asked questions, world-wide phone access for DB2 support, and much more, can be found.

CompuServe   **IBM DB2 Family Forum** or **GO IBMDB2**

                                       All DB2 products are supported through these forums, as well as Visualizer and IMS products.

Internet   **comp.databases**, **comp.databases.ibm-db2**, or **bit.listserv.db2-l** newsgroups.

                                       These newsgroups are available for DB2 users to discuss their experiences with the products.

Anonymous FTP Sites **ps.boulder.ibm.com**

In the directory `/ps/products/db2`, you can find demos, fixes, information, and tools concerning DB2 and many related products.

IBM®

Part Number: 33H4878

Printed in U.S.A.

33H4878