

IBM[®] DB2[®] Universal Database and DB2 Connect[™]



Installation and Configuration Supplement

Version 7

IBM[®] DB2[®] Universal Database and DB2 Connect[™]



Installation and Configuration Supplement

Version 7

Before using this information and the product it supports, be sure to read the general information under "Appendix G. Notices" on page 513.

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

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

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

© **Copyright International Business Machines Corporation 1993, 2000. All rights reserved.**

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

Contents

Welcome to DB2 Universal Database! . . .	xi
Conventions	xi

Part 1. Installing and Configuring DB2 Clients 1

Chapter 1. Planning for Installation	3
Memory Requirements	3
Disk Requirements	3
DB2 Clients	3
Software Requirements	4
Client Product Requirements	4
Possible Client-to-Server Connectivity Scenarios	9
NetQuestion Search System	10
Your Next Step	10
Chapter 2. Installing DB2 Clients	11
DB2 Run-Time Client	11
DB2 Administration Client	12
DB2 Application Development Client	12
Distributed Installation	12
DB2 Thin Client	13
Chapter 3. Installing DB2 Clients on Windows 32-Bit Operating Systems	15
Before You Begin Installing	15
Installing Without Administrator Authority	15
Installation Steps	16
Chapter 4. Installing DB2 Clients on OS/2 Operating Systems	19
Before You Begin Installing	19
Installation Steps	19
Chapter 5. Installing DB2 Clients on UNIX Operating Systems	23
Before You Begin	23
About the db2setup utility	23
Updating Kernel Configuration Parameters	24
HP-UX Kernel Configuration Parameters	24
NUMA-Q/PTX Kernel Configuration Parameters	25
Solaris Kernel Configuration Parameters	26

Installing Your DB2 Client	27
Your next step	28

Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant	29
LDAP Directory Support Considerations	29
Before You Begin	29
Configuration Steps	30
Adding a Database Using a Profile	30
Adding a Database Using Discovery	32
Adding a Database Manually	34
Creating and Using Profiles	36
Server Profiles	36
Client Profiles	36

Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor	41
Configuring Named Pipes on the Client	42
Step 1. Identify and Record Parameter Values	42
Step 2. Configure the Client	43
Step 3. Test the Client-to-Server Connection	45
Configuring TCP/IP on the Client	47
Step 1. Identify and Record Parameter Values	48
Step 2. Configure the Client	49
Step 3. Test the Client-to-Server Connection	54
Configuring NetBIOS on the Client	56
Step 1. Identify and Record Parameter Values	56
Step 2. Configure the Client	57
Step 3. Test the Client-to-Server Connection	61
Configuring IPX/SPX on the Client	63
Step 1. Identify and Record Parameter Values	64
Step 2. Configure the Client	65
Step 3. Test the Client-to-Server Connection	69
Configuring APPC on the Client	71

Step 1. Identify and Record Parameter Values	71
Step 2. Update the APPC Profiles	73
Step 3. Catalog the APPC or APPN Node	109
Step 4. Catalog the Database	110
Step 5. Test the Client-to-Server Connection	112
Troubleshooting the Client-to-Server Connection	113

Chapter 8. Control Center Installation and Configuration. 115

Application versus Applet	115
Machine Configurations.	116
Supported Java Virtual Machines for the Control Center	117
Setting Up and Working with the Control Center	118
Control Center Services Setup (Applet Mode only)	118
Working with the Control Center	120
Functional Considerations	122
Installation Tips for Control Center Help on UNIX Operating Systems	122
Configuring TCP/IP on OS/2.	123
Enabling Local Loopback	123
Enabling Localhost	123
Verifying TCP/IP Configuration on OS/2	124
Troubleshooting Information	124
Administering DB2 for OS/390 and DB2 Connect Enterprise Edition servers with the Control Center	125
Preparing DB2 for OS/390 Servers for the Control Center.	126
Working with the Control Center	126
Other Sources of Information	127

Chapter 9. Configuring Stored Procedure Builder 129

Configuring Stored Procedure Builder to run as an add-in with Microsoft Visual Basic	129
Configuring Stored Procedure Builder to run as an add-in with Microsoft Visual C++	129
Configuring Stored Procedure Builder on AIX and Solaris	130

Part 2. Setting Up Server Communications. 131

Chapter 10. Using the Control Center to Configure Server Communications 133

Before You Begin	133
About the Control Center and Communications Protocols.	133
Configuring DB2 Communications for Local Instances.	134
Configuring DB2 Communications for Remote Instances	135

Chapter 11. Using the Command Line Processor to Configure Server Communications 137

Setting the DB2COMM Registry Parameter	137
Configuring Named Pipes on the Server	139
Configuring TCP/IP on the Server	139
1. Identify and Record Parameter Values	140
2. Configure the Server	141
Configuring NetBIOS on the Server.	143
1. Identify and Record Parameter Values	143
2. Configure the Server	144
Autostarting DB2 with NetBIOS (for Windows NT only)	147
Configuring IPX/SPX on the Server.	147
1. Identify and Record Parameter Values	148
2. Configure the Server	150
Configuring APPC on the Server.	152
1. Identify and Record Parameter Values	153
2. Configure the Server	154

Part 3. CLI/ODBC Enablement 183

Chapter 12. Running Your Own Applications 185

Binding Database Utilities	185
Running CLI/ODBC Programs	187
Platform Specific Details for CLI/ODBC Access	189
Detailed Configuration Information.	195
Running Java Programs.	198
Configuring the Environment.	199
Java Applications	201
Java Applets	201

Chapter 13. DB2 CLI/ODBC Configuration Keyword Listing. 203

Configuration Keywords by Category	203
CLI/ODBC Settings General	
Configuration Keywords	203

Compatibility Configuration Keywords	203
Data Type Configuration Keywords	203
Enterprise Configuration Keywords	203
Environment Configuration Keywords	204
File DSN Configuration Keywords	204
Optimization Configuration Keywords	204
Service Configuration Keywords	204
Static SQL Configuration Keywords	205
Transaction Configuration Keywords	205
Configuration Keyword Descriptions	205
APPENDAPINAME	205
ASYNCENABLE	206
BITDATA	206
CLIPKG	207
CLISHEMA	208
CONNECTNODE	209
CONNECTTYPE	210
CURRENTFUNCTIONPATH	210
CURRENTPACKAGESET	211
CURRENTREFRESHAGE	212
CURRENTSCHEMA	212
CURRENTSQLID	213
CURSORHOLD	213
DATABASE	214
DB2CONNECTVERSION	215
DB2DEGREE	215
DB2ESTIMATE	216
DB2EXPLAIN	217
DB2OPTIMIZATION	218
DBALIAS	219
DBNAME	219
DEFAULTPROCLIBRARY	220
DEFERREDPREPARE	221
DISABLEMULTITHREAD	222
EARLYCLOSE	222
GRANTEELIST	223
GRANTORLIST	224
GRAPHIC	225
HOSTNAME	226
IGNOREWARNINGS	226
IGNOREWARNLIST	227
KEEPCONNECT	227
KEEPSTATEMENT	228
LOBMAXCOLUMNSIZE	229
LONGDATACOMPAT	229
MAXCONN	230
MODE	231
MULTICONNECT	231
OPTIMIZEFORNROWS	232
OPTIMIZESQLCOLUMNS	233

PATCH1	233
PATCH2	234
POPUPMESSAGE	235
PROTOCOL	235
PWD	236
QUERYTIMEOUTINTERVAL	236
SCHEMALIST	237
SERVICENAME	238
SQLSTATEFILTER	239
STATICCAPFILE	239
STATICLOGFILE	240
STATICMODE	240
STATICPACKAGE	241
SYNCPOINT	242
SYSSHEMA	243
TABLETYPE	244
TEMPDIR	245
TRACE	246
TRACECOMM	246
TRACEFILENAME	247
TRACEFLUSH	248
TRACEPATHNAME	249
TXNISOLATION	250
UID	250
UNDERScore	251
WARNINGLIST	252

Part 4. Configuring DB2 Connect to Host or AS/400 Communications. 253

Chapter 14. Configuring Host Communications Using the Command Line Processor 255

Chapter 15. Manually Configuring TCP/IP Communications on the DB2 Connect Workstation 257

1. Identify and Record Parameter Values	258
2. Configure the DB2 Connect Workstation	259
A. Resolve the Host's IP Address	259
B. Update the Services File	260
3. Catalog the TCP/IP Node	261
4. Catalog the Database as a Database Connection Service (DCS) Database	262
5. Catalog the Database	262
6. Bind Utilities and Applications to the Database Server	264
7. Test the Host or AS/400 Connection	264

Test the Host Connection	265
Chapter 16. Manually Configuring APPC Communications on the DB2 Connect Workstation	267
1. Identify and Record Parameter Values	268
2. Update the APPC Profiles on the DB2 Connect Workstation.	271
Configuring IBM eNetwork Communications Server for OS/2	271
Configuring IBM Personal Communications for Windows	284
Configuring IBM eNetwork Communications Server for Windows	295
Configuring IBM eNetwork Communications Server for Windows NT SNA API Client	300
Configuring Microsoft SNA Server for Windows	303
Configuring Microsoft SNA Client	312
Configuring IBM eNetwork Communication Server for AIX	315
Configuring Bull SNA for AIX	324
Configuring SNAPPlus2 for HP-UX	327
Configuring SNAP-IX Version 6.0.1 for SPARC Solaris	336
Configuring SunLink 9.1 for Solaris.	344
3. Catalog the APPC or APPN Node	347
4. Catalog the Database as a Database Connection Service (DCS) Database.	348
5. Catalog the Database.	349
6. Bind Utilities and Applications to the Database Server	350
7. Test the Host or AS/400 Connection.	351

Chapter 17. Enabling Multisite Updates (Two-Phase Commit)	353
Host and AS/400 Multisite Update Scenarios that Require SPM.	354

Part 5. Configuring DB2 UDB as a DRDA Application Server 359

Chapter 18. Accessing DB2 Universal Database Servers from Host and AS/400 Applications	361
Supported Clients.	361
PTFs Required	362

Configuration Steps for DB2 Universal Database Server	362
Configuring DB2 Universal Database Servers for Host or AS/400 Client Access.	363
Using the DB2 Universal Database Server from Host or AS/400 Clients	364
Authentication	364
Troubleshooting	365
DRDA Functions Supported	365
Security and Auditability	370
Configuration Considerations.	371

Part 6. Distributed Installation 373

Chapter 19. An Introduction to Distributed Installation.	375
Types of Distributed Installation	375
Response File	375
Available Sample Response Files.	376
Important Response File Keywords	377
Response File Keywords for OS/2 and Windows 32-Bit Operating Systems	377
Response File Keywords for DB2 Satellite Edition	381
DB2 Control Server Response File Keywords for Windows NT and Windows 2000	383
Killing DB2 Processes During Interactive and Response File Installations	384
Response File Generator	384
Where Do You Go From Here?	386

Chapter 20. Distributed DB2 Installation on Windows 32-Bit Operating Systems.	387
Before You Begin	387
Make DB2 Files Available for Installation	387
Set up Shared Access	388
Create a Response File	388
Run Setup with the Response File from the Client Workstation	390
DB2 Product Installation Using SMS	392
SMS Requirements	392
Import the DB2 Install File into SMS on the SMS Server	392
Create the SMS Package on the SMS Server.	393
Distribute the DB2 Installation Package from the SMS Server.	394
Configuring Client Settings	396

Configuring Remote Access to a Server Database	396
Configuring db2cli.ini	397
Exporting and Importing a Profile	398

Chapter 21. Distributed DB2 Installation on UNIX Operating Systems	399
Before You Begin	399
Limitations on Response File Install.	399
Step 1. Mount the CD-ROM	399
Step 2. Create a Response File	400
Step 3. Start an Unattended Installation with a Response File	401

Chapter 22. Distributed DB2 Installation on OS/2 Operating Systems	403
Installing DB2 Products from a Hard Disk or CD-ROM	403
Before You Begin	403
Make DB2 Files Available for Installation	403
Create a Response File for Distributed Installation	404
Run the CMD File from the Remote Workstation.	405

Part 7. Thin Client and Thin Connect Architecture 407

Chapter 23. Installing and Configuring Thin-Workstations	409
The Benefits and Disadvantages in a Thin Environment	411
Performing the Installation of a DB2 Thin-Client or DB2 Thin-Connect Environment	412
Step 1. Install a DB2 Administration Client or DB2 Connect Personal Edition With The Code Server Component	412
Step 2. Setup Cross-Platform Support on the Code Server (Optional).	413
Step 3. Share the Code Server Directory where the DB2 Client or DB2 Connect Personal Edition Code is Installed	414
Step 4. Create a Response File for the Target Thin-Workstation.	415
Step 5. Make the Code Server Accessible to the Target Thin-Workstation.	417
Step 6. Create the Target Thin-Workstation	418

Your Next Step	419
--------------------------	-----

Part 8. Configuring a Federated Database System 421

Chapter 24. Creating and Configuring a Federated Database System	423
Supported Data Sources.	424

Chapter 25. Setting Up a Federated System to Access DB2 Family Data Sources	427
Enabling Federated Database Functionality	427
Adding DB2 Family Data Sources to a Federated System.	428
Verifying Connections to DB2 Family Data Sources	431

Chapter 26. Setting Up a Federated System to Access Oracle Data Sources	433
Installing DB2 Relational Connect	433
Installing DB2 Relational Connect on Windows Systems.	433
Installing DB2 Relational Connect on AIX Systems	434
Adding Oracle Data Sources to a Federated System	435
Oracle Code Page Options	441
Verifying Connections to Oracle Data Sources	441

Chapter 27. Setting Up a Federated System to Access OLE DB Data Sources	443
Enabling Federated Database Functionality	443
Adding OLE DB data sources to a Federated System	443

Part 9. Appendixes 445

Appendix A. Basic Task Knowledge	447
Starting the Client Configuration Assistant	447
Starting the DB2 Control Center	447
Entering Commands Using the Command Center	448
Entering Commands Using the Command Line Processor	449
DB2 Command Window	450
Interactive Input Mode	450

Working with the System Administrative Group	451	Installing NetQuestion on UNIX Systems	475
Granting Advanced User Rights on Windows	452	NetQuestion Installation Workarounds on UNIX Platforms	475
Windows NT	452		
Windows 2000	453		
Working with the Business Intelligence Functions	453		
Mounting CD-ROMs on UNIX Operating Systems	453		
Mounting the CD-ROM on AIX	453		
Mounting the CD-ROM on HP-UX	454		
Mounting the CD-ROM on Linux	455		
Mounting the CD-ROM on PTX	455		
Mounting the CD-ROM on Solaris	455		
Setting the Number of Licensed Processors	456		
Upgrading DB2 from Try and Buy Mode	457		
Appendix B. Working with NetQuestion	459		
NetQuestion Overview	459		
Searching the DB2 Online Information	460		
Solving Search Problems	460		
NetQuestion for Windows 32-bit Operating Systems	464		
Diagnosing NetQuestion Install Errors	464		
TCP/IP Configuration	465		
Changing the Port Number for the Search Server.	465		
Ensuring NetQuestion Environment Variables are set on a LAN-Connected Drive	466		
Locating the NetQuestion Installation Directory.	466		
Searching With Proxies Enabled in Netscape or Internet Explorer.	466		
Searching with a Laptop on Windows 9x	467		
Uninstalling NetQuestion on Windows 32-bit Operating Systems	468		
NetQuestion for OS/2 Operating Systems	470		
Additional NetQuestion Prerequisites	470		
Switching NetQuestion to use TCP/IP	470		
Changing the Port Number for the Search Server (TCP/IP only)	471		
Error Detected When Starting NetQuestion	472		
Searching while Disconnected from the Network	472		
Locating the NetQuestion Directory.	472		
Uninstalling NetQuestion on OS/2	472		
NetQuestion for UNIX Operating Systems	475		
		Appendix C. Setting up DB2 Documentation on a Web Server	479
		Considerations for Using a Web Server	479
		Working with the DB2 Universal Database Documentation Files in a Client/Server Environment	480
		Typical Web Server Scenarios	481
		Scenario 1: Lotus Domino Go! Web Server on OS/2	481
		Scenario 2: Netscape Enterprise Web Server on Windows NT	482
		Scenario 3: Microsoft Internet Information Server on Windows NT	484
		Serving Documentation in Multiple Languages	484
		Serving Documentation for Multiple Platforms	486
		Appendix D. Using the DB2 Library	487
		DB2 PDF Files and Printed Books	487
		DB2 Information	487
		Printing the PDF Books	496
		Ordering the Printed Books	497
		DB2 Online Documentation	498
		Accessing Online Help	498
		Viewing Information Online	500
		Using DB2 Wizards	502
		Setting Up a Document Server	503
		Searching Information Online	504
		Appendix E. National Language Support (NLS)	505
		Language and Codeset Support for UNIX Operating Systems	505
		Code Page and Language Support for OS/2 and Windows Operating Environments	506
		Appendix F. Naming Rules	509
		General Naming Rules	509
		Database, Database Alias, and Catalog Node Name Rules.	509
		Object Name Rules	510
		Username, User ID, Group Name, and Instance Name Rules.	511
		Workstation Name (nname) Rules	512
		DB2SYSTEM Naming Rules	512

Password Rules	512	Index	519
Appendix G. Notices	513	Contacting IBM	527
Trademarks	516	Product Information	527

Welcome to DB2 Universal Database!

The *Installation and Configuration Supplement* guides you through the planning, installation, migration (if necessary), and set up a platform-specific DB2 client. Once the DB2 client is installed, you will then configure communications for both the client and server, using the DB2 GUI tools or the Command Line Processor. This supplement also contains information on binding, setting up communications on the server, the DB2 GUI tools, and DRDA AS.

This supplement also discusses the configuration of distributed requests and access methods to heterogeneous data sources.

The distributed installation section guides you through a network-wide roll out of DB2 products on all supported platforms. This section also details thin client installation and thin connect architecture.



Conventions

This book uses the following highlighting conventions:

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



This icon marks a fast path. A fast path guides you to information specific to your configuration where multiple options are available.



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

For a complete description of the DB2 library, see “Appendix D. Using the DB2 Library” on page 487.



- If you do not follow the documented installation method with the recommended defaults, it may be necessary to refer to the *Administration Guide* and the *Command Reference* to complete the installation and configuration.
- The term *Windows 32-bit operating systems* refers to Windows 95, Windows 98, Windows NT, or Windows 2000.
- The term *Windows 9x* refers to Windows 95 or Windows 98.
- The term *DB2 client* refers to a DB2 Run-Time Client, a DB2 Administration Client, or a DB2 Application Development Client.
- In this book, the term *DB2 Universal Database* refers to DB2 Universal Database on OS/2, UNIX, and Windows 32-bit operating systems, unless otherwise stated.

Part 1. Installing and Configuring DB2 Clients

Chapter 1. Planning for Installation

Before you install DB2, you should make sure that your system meets DB2's hardware and software requirements. If you are migrating from a previous version of DB2, there are also pre-installation migration tasks you should perform to prepare your databases.

This chapter describes the following requirements you should take into consideration before installing DB2:

- "Disk Requirements".
- "Software Requirements" on page 4.
- "Possible Client-to-Server Connectivity Scenarios" on page 9.

If you know that your system meets all the hardware and software requirements, skip this section and proceed to the appropriate DB2 client installation section:

- "Chapter 3. Installing DB2 Clients on Windows 32-Bit Operating Systems" on page 15.
- "Chapter 4. Installing DB2 Clients on OS/2 Operating Systems" on page 19.
- "Chapter 5. Installing DB2 Clients on UNIX Operating Systems" on page 23.

Memory Requirements

To run a DB2 Run-Time Client or an DB2 Application Development Client, you need a minimum of 16 MB of RAM. If you are planning to run a DB2 Administration Client, you need a minimum of 32 MB of RAM.

Disk Requirements

The actual fixed disk requirements of your installation may vary depending on your file system and the components you install. Ensure that you have included a disk space allowance for your operating system, application development tools, application data, and communications products. For more information about space requirements for data, refer to the *Administration Guide*.

DB2 Clients

Use Table 1 on page 4 to estimate the amount of disk space you need on each of your client workstations. You may require additional amounts of disk space depending on your file system.

Table 1. Disk Requirements for Client Components

Client Components	Recommended Minimum Disk (MB)
OS/2	
DB2 Run-Time Client	30 MB
DB2 Application Development Client	125 MB, not including the Java Development Kit (JDK)
DB2 Administration Client	95 MB
UNIX platforms	
DB2 Run-Time Client	30 to 40 MB (70 MB for Silicon Graphics IRIX)
DB2 Application Development Client	90 to 120 MB, not including the JDK (40 MB for NUMA-Q)
DB2 Administration Client	80 to 110 MB
Note: PTX/NUMA-Q and Silicon Graphics IRIX operating systems do not support the DB2 Administration Client.	
Windows 32-bit Operating Systems	
DB2 Run-Time Client	25 MB
DB2 Application Development Client	325 MB, including the JDK
DB2 Administration Client	125 MB

The DB2 Application Development Client and DB2 Administration Client include tools and documentation, except on NUMA-Q systems.

Software Requirements

This section outlines the software required to run DB2 products.

Client Product Requirements

Table 2 on page 5 lists the software requirements needed for a DB2 Administration Client, DB2 Run-Time Client, or a DB2 Application Development Client.

On all platforms you will need a Java Runtime Environment (JRE) Version 1.1.8 to run the DB2 tools, such as the Control Center. If you intend to run the Control Center as an applet on Windows 32-bit or OS/2 systems, you will need a Java-enabled browser. See “Chapter 8. Control Center Installation and Configuration” on page 115 for more information.

Table 2. Software Requirements for Clients

Component	Hardware/Software Requirements	Communications
<ul style="list-style-type: none"> • DB2 Run-Time Client for AIX • DB2 Administration Client for AIX • DB2 Application Development Client for AIX 	<p>RISC System/6000 and the following:</p> <ul style="list-style-type: none"> • AIX Version 4.2 or later • For OLAP Starter Kit, AIX Version 4.3 or later. • For LDAP (Lightweight Directory Access Protocol) support, you require an IBM SecureWay Directory Client V3.1.1 running on AIX V4.3.1 or later. • For Warehouse Agent, bos.iconv.ucs.com and bos.iconv.ucs.pc at AIX Version 4.2 or later. • For Data Links Manager in DCE-DFS environments, you require DCE Version 3.1. <p>Note: When the DB2 Application Development Client is installed, the JDK 1.1.8 is only installed if no other version of the JDK is detected.</p>	<p>APPC or TCP/IP</p> <ul style="list-style-type: none"> • For APPC connectivity, you require IBM eNetwork Communications Server Version 5.0.3 or later for AIX • The AIX base operating system provides TCP/IP connectivity, if selected during install. <p>Note: If you plan to use DCE (Distributed Computing Environment) and you are not using DB2 Data Links Manager, you require a DCE product that is provided by the AIX base operating system.</p>
<ul style="list-style-type: none"> • DB2 Run-Time Client for HP-UX • DB2 Administration Client for HP-UX • DB2 Application Development Client for HP-UX 	<p>HP 9000 Series 700 or 800 system and the following:</p> <ul style="list-style-type: none"> • HP-UX Version 11.00 or later <p>Note: When the DB2 Application Development Client is installed, the JDK is not installed. Contact your operating system vendor for the latest version of the JDK.</p>	<p>APPC or TCP/IP</p> <ul style="list-style-type: none"> • TCP/IP is provided with the HP-UX base operating system. • For APPC connectivity, you require either of the following: <ul style="list-style-type: none"> – SNAplus2 Link R6.11.00.00 – SNAplus2 API R6.11.00.00 <p>Note: If you plan to use DCE (Distributed Computing Environment), you require a DCE product that is provided by the HP-UX Version 11 base operating system.</p>

Table 2. Software Requirements for Clients (continued)

Component	Hardware/Software Requirements	Communications
<ul style="list-style-type: none"> DB2 Run-Time Client for Linux DB2 Administration Client for Linux DB2 Application Development Client for Linux 	<ul style="list-style-type: none"> Linux kernel 2.2.12 or higher; <i>glibc</i> Version 2.1.2 or higher; <i>pdksh</i> package (required to run the DB2 command line processor); and <i>libstdc++</i> Version 2.9.0. <p>To install DB2, you will need <i>rpm</i>.</p> <p>Note: When the DB2 Application Development Client is installed, the JDK is not installed. Contact your operating system vendor for the latest version of the JDK.</p>	<p>TCP/IP</p> <ul style="list-style-type: none"> The Linux base operating system provides TCP/IP connectivity, if selected during installation.
<ul style="list-style-type: none"> DB2 Run-Time Client for OS/2 DB2 Administration Client for OS/2 DB2 Application Development Client for OS/2 	<ul style="list-style-type: none"> OS/2 Warp Version 4 OS/2 Warp Server Version 4 OS/2 Warp Server Advanced V4 OS/2 Warp Server Advanced V4 with SMP Feature OS/2 Warp Server for e-business <p>Note: When the DB2 Application Development Client is installed, the JDK is not installed. You can install the latest version of the JDK from your product CD-ROM.</p>	<p>APPC, IPX/SPX, NetBIOS, or TCP/IP</p> <ul style="list-style-type: none"> For APPC connectivity, you require IBM eNetwork Communications Server for OS/2 Warp Version 5 or IBM eNetwork Personal Communications for OS/2 Warp Version 4.2. For IPX/SPX connectivity, you require the Novell NetWare client for OS/2 Version 2.10 or later. IPX/SPX can only be used to connect to databases. It cannot be used to connect to host or AS/400 databases. The OS/2 base operating system provides NetBIOS and TCP/IP connectivity, if selected during installation. The OS/2 base operating system provides Named Pipes (Local) connectivity. Named Pipes is supported in DOS and WIN-OS/2 sessions. <p>Notes:</p> <ol style="list-style-type: none"> Net.Data requires a Web server such as WebSphere. For DCE Cell Directory Services Support (CDS) for DB2 Clients for OS/2, you must install an IBM Distributed Computing Environment Cell Directory Service client on each client workstation. If you are planning to use Tivoli Storage Manager, PTF 3 for Tivoli Storage Manager Version 3 is required for an OS/2 client.

Table 2. Software Requirements for Clients (continued)

Component	Hardware/Software Requirements	Communications
<ul style="list-style-type: none"> • DB2 Run-Time Client for NUMA-Q • DB2 Application Development Client for NUMA-Q 	<ul style="list-style-type: none"> • NUMA-Q system running PTX Version 4.5 or later. • ptx/EFS v1.4.0 with templog is required. <p>Note: When the DB2 Application Development Client is installed, the JDK is not installed. Contact your operating system vendor for the latest version of the JDK.</p>	<p>TCP/IP</p> <ul style="list-style-type: none"> • For TCP/IP connectivity, no additional software is required.
<ul style="list-style-type: none"> • DB2 Run-Time Client for Silicon Graphics IRIX • DB2 Application Development Client for Silicon Graphics IRIX 	<ul style="list-style-type: none"> • Silicon Graphics IRIX, Version 6.x, and the following filesets: <ul style="list-style-type: none"> – eoe.sw.oampkg – eoe.sw.svr4net <p>The following patches are required for Versions 6.2 and 6.3:</p> <ul style="list-style-type: none"> – 2791.0 – 3778.0 <p>Note: When the DB2 Application Development Client is installed, the JDK is not installed. Contact your operating system vendor for the latest version of the JDK.</p>	<p>TCP/IP</p> <ul style="list-style-type: none"> • The Silicon Graphics IRIX base operating system provides TCP/IP connectivity.
<ul style="list-style-type: none"> • DB2 Run-Time Client for Solaris • DB2 Administration Client for Solaris • DB2 Application Development Client for Solaris 	<p>Solaris SPARC-based computer and the following:</p> <ul style="list-style-type: none"> • Solaris Version 2.6 or later. <p>The following patches are required for Solaris version 2.6:</p> <ul style="list-style-type: none"> • 105568 – 12 or higher • 105210 – 25 or higher • 105181 – 17 or higher <p>Note: When the DB2 Application Development Client is installed, the JDK is not installed. Contact your operating system vendor for the latest version of the JDK.</p>	<p>APPC or TCP/IP</p> <ul style="list-style-type: none"> • For APPC connectivity, you require SunLink SNA 9.1 or later and the following: <ul style="list-style-type: none"> – SunLink P2P LU6.2 9.0 or later – SunLink PU2.1 9.0 or later – SunLink P2P CPI-C 9.0 or later • The Solaris base operating system provides TCP/IP connectivity. • If you plan to use DCE (Distributed Computing Environment), you require Transarc DCE Version 2.0 or later.

Table 2. Software Requirements for Clients (continued)

Component	Hardware/Software Requirements	Communications
<ul style="list-style-type: none"> • DB2 Run-Time Client for Windows 9x • DB2 Administration Client for Windows 9x • DB2 Application Development Client for Windows 9x 	<ul style="list-style-type: none"> • Windows 95 4.00.950 or later • Windows 98 <p>Note: When the DB2 Application Development Client is installed, the JDK 1.1.8 is installed.</p>	<p>IPX/SPX, Named Pipes, NetBIOS, or TCP/IP</p> <ul style="list-style-type: none"> • The Windows 9x base operating system provides NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. Note: IPX/SPX connectivity is only supported to Windows NT and Windows 2000 servers. • If you plan to use LDAP (Lightweight Directory Access Protocol), you require either a Microsoft LDAP client or an IBM SecureWay LDAP client V3.1.1. For more information, refer to the <i>Administration Guide</i>. • If you plan to use the Tivoli Storage Manager facilities for backup and restore of your databases, you require the Tivoli Storage Manager Client Version 3 or later. • If you have the IBM Antivirus program installed on your operating system, it must be disabled or uninstalled to complete a DB2 installation.
<ul style="list-style-type: none"> • DB2 Run-Time Client for Windows • DB2 Administration Client for Windows • DB2 Application Development Client for Windows 	<ul style="list-style-type: none"> • Windows NT Version 4.0 with Service Pack 3 or later • Windows Terminal Server (can only run the DB2 Run-Time Client) • Windows 2000 <p>Note: When the DB2 Application Development Client is installed, the JDK 1.1.8 is installed.</p>	<p>APPC, IPX/SPX, Named Pipes, NetBIOS, or TCP/IP</p> <ul style="list-style-type: none"> • The Windows NT and Windows 2000 base operating systems provide NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. • For APPC connectivity, you require one of the following products: <ul style="list-style-type: none"> – IBM eNetwork Communications Server for Windows V5.01 or later. – Windows 2000: IBM eNetwork Personal Communications for Windows Version 4.3 CSD2 or later – Windows NT: IBM eNetwork Personal Communications for Windows Version 4.2 or later – Microsoft SNA Server Version 3 Service Pack 3 or later – Wall Data Rumba • If you plan to use DCE (Distributed Computing Environment) and connect to a DB2 for OS/390 V5.1 database, that database must be enabled for DCE support using OS/390 DCE Base Services Version 3. • If you plan to use LDAP (Lightweight Directory Access Protocol), you require either a Microsoft LDAP client or an IBM SecureWay LDAP client V3.1.1. For more information, refer to the <i>Administration Guide</i>. • If you plan to use the Tivoli Storage Manager facilities for backup and restore of your databases, you require the Tivoli Storage Manager Client Version 3 or later. • If you have the IBM Antivirus program installed on your operating system, it must be disabled or uninstalled to complete a DB2 installation.

Possible Client-to-Server Connectivity Scenarios

The following table shows the communication protocols that can be used when connecting a specific DB2 client to a specific DB2 server. DB2 Workgroup, DB2 Enterprise, and DB2 Enterprise - Extended Editions can service requests from host or AS/400 clients (DRDA ARs).

Table 3. Possible Client-to-Server Connectivity Scenarios

Client	Server						
	AIX	HP-UX	Linux	OS/2	PTX/NUMA-Q	Solaris	Windows NT/ Windows 2000
AS/400 V4R1	APPC	N/A	N/A	APPC	N/A	APPC	APPC
AS/400 V4R2	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP
AIX	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP
HP-UX	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP
Linux	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
MVS	APPC	N/A	N/A	APPC	N/A	APPC	APPC
OS/2	APPC IPX/SPX(1),(2) TCP/IP	TCP/IP	TCP/IP	APPC IPX/SPX(1),(2) NetBIOS TCP/IP	TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) NetBIOS TCP/IP
OS/390	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP
PTX/NUMA-Q	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
Silicon Graphics IRIX	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
SQL/DS	APPC	N/A	N/A	APPC	N/A	APPC	APPC
Solaris	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP
VSE & VM V5	APPC	N/A	N/A	APPC	N/A	APPC	APPC
VSE V6	APPC	N/A	N/A	APPC	N/A	APPC	APPC
VM V6	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP
Windows 9x	TCP/IP	TCP/IP	TCP/IP	NetBIOS TCP/IP	TCP/IP	TCP/IP	IPX/SPX(1) NPIPE NetBIOS TCP/IP
Windows NT/ Windows 2000	APPC IPX/SPX(1) TCP/IP	TCP/IP	TCP/IP	APPC IPX/SPX(1) NetBIOS TCP/IP	TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) NPIPE NetBIOS TCP/IP

1. Direct Addressing
2. File Server Addressing

NetQuestion Search System

If you installed online product documentation with your previous version of DB2 for Windows, OS/2, AIX, HP-UX or Solaris, or if you installed another IBM product like VisualAge C++ or VisualAge for Java, an online search system called NetQuestion was also installed automatically.

If the version of NetQuestion that comes with DB2 Version 7 is later than the version of NetQuestion currently present on your system, the current version will be upgraded and any existing document indices will be re-registered with NetQuestion. This will be done automatically for you during DB2 installation.

For more information on NetQuestion, see “Appendix B. Working with NetQuestion” on page 459.

Your Next Step

After you have determined that your system meets all hardware and software requirements, and after you have prepared any existing databases and instances for migration, you can now install DB2 using either the interactive or distributed method. For installation procedures, see “Chapter 19. An Introduction to Distributed Installation” on page 375.

Chapter 2. Installing DB2 Clients

This section describes the different DB2 clients, and provides information on distributed installation, and thin client configuration.

Note: DB2 clients can connect to DB2 servers *two* releases later or *one* release earlier than the client's release level, as well as to servers at the same release level. For example, a DB2 Version 5.2 client can connect to DB2 Version 5.0, 5.2, 6.1, and 7.1 servers, while a DB2 Version 7.1 client can connect to DB2 Version 6.1 and 7.1 servers.

You may install a DB2 client on any number of workstations. For information about licensing, refer to your *License Information Booklet*.

You cannot create a database on a DB2 client, only connect to databases that reside on a DB2 server.



To go to the section that provides client installation instructions:

- “Chapter 3. Installing DB2 Clients on Windows 32-Bit Operating Systems” on page 15
- “Chapter 4. Installing DB2 Clients on OS/2 Operating Systems” on page 19
- “Chapter 5. Installing DB2 Clients on UNIX Operating Systems” on page 23

To download installation packages for DB2 clients supported on other platforms, and pre-Version 7 clients, connect to the IBM DB2 Client Application Enabler Web site at
<http://www.ibm.com/software/data/db2/db2tech/clientpak.html>

DB2 Run-Time Client

A DB2 Run-Time Client provides the ability for workstations from a variety of platforms to access DB2 databases.

DB2 Run-Time Clients are available for the following platforms: AIX, HP-UX, Linux, OS/2, NUMA-Q, Silicon Graphics IRIX, the Solaris Operating Environment, and Windows 32-bit operating systems.

DB2 Administration Client

A DB2 Administration Client provides the ability for workstations from a variety of platforms to access and administer DB2 databases. The DB2 Administration Client has all the features of the DB2 Run-Time Client and also includes all the DB2 Administration tools, documentation, and support for Thin Clients.

The DB2 Administration Client also includes the client components for DB2 Query Patroller, a sophisticated query management and workload distribution tool. In order to use Query Patroller, you must have a Query Patroller server installed. For more information refer to the *DB2 Query Patroller Installation Guide*.

DB2 Administration Clients are available for the following platforms: AIX, HP-UX, Linux, OS/2, Solaris, Windows 32-bit operating systems.

DB2 Application Development Client

The DB2 Application Development Client was known as the DB2 Software Development Kit (DB2 SDK) in previous versions of DB2. DB2 Application Development Client provides the tools and environment you need to develop applications that access DB2 servers and application servers that implement Distributed Relational Database Architecture (DRDA). You can build and run DB2 applications with a DB2 Application Development Client installed. You can also run DB2 applications on a DB2 Administration Client and a DB2 Run-Time Client.

DB2 Application Development Clients are available for the following platforms: AIX, HP-UX, Linux, OS/2, NUMA-Q, Silicon Graphics IRIX, the Solaris Operating Environment, and Windows 32-bit operating systems.

The applicable DB2 Application Development Client can be found on the server product CD-ROM. The DB2 Application Development Client for all platforms can be found on the set of DB2 Application Development Client CD-ROMs.

Distributed Installation

If you are planning to install DB2 products across your network, you may consider using a distributed installation. With a network-based installation, you can roll out multiple identical copies of DB2 products. For more information refer to “Chapter 19. An Introduction to Distributed Installation” on page 375.

DB2 Thin Client

You can install a DB2 client for Windows 9x, Windows NT, or Windows 2000 on a code server, and have Thin Client workstations access the code across a LAN connection. Thin Client workstations function like any other DB2 client. In the setup, the main difference is that the DB2 client code is installed on a code server, and not individually on each workstation. Thin Client workstations need only a minimal configuration to set parameters and establish links to a code server. For more information, refer to “Chapter 23. Installing and Configuring Thin-Workstations” on page 409.

Chapter 3. Installing DB2 Clients on Windows 32-Bit Operating Systems

This section contains the information you need to install a DB2 client on Windows 32-bit operating systems.

Before You Begin Installing

1. Ensure that your system meets all of the memory, hardware, and software requirements to install your DB2 client. For more information, see “Chapter 1. Planning for Installation” on page 3.
2. You require a user account to perform the installation.

Windows 9x

Any valid Windows 9x user.

Windows NT or Windows 2000

A user account that belongs to a group with more authority than the Guests group. For example, the Users group or the Power Users group. For information about installing on Windows NT or Windows 2000 without administrator authority, see “Installing Without Administrator Authority”.

Installing Without Administrator Authority

When installing a DB2 client without administrator authority on Windows NT and Windows 2000, you will not be able to install these components:

- Control Center
- NetQuestion
- Integrated SNA Support

The following are some specific installation scenarios:

- *A user has installed a DB2 product without having administrator authority, and then an administrator installs a DB2 product on the same machine. In this scenario, the installation performed by the administrator will remove the previous installation performed by the user without administrator authority, resulting in a clean installation of the DB2 product. The installation performed by the administrator will overwrite all of the user’s services, shortcuts, and environment variables from the previous installation of DB2.*
- *A user without administrator authority has installed a DB2 product, and then a second user without administrator authority attempts to install a DB2 product on*

the same machine. In this scenario, the installation by the second user will fail, and return an error message that the user must be an administrator to install the product.

- *An administrator has installed a DB2 product, and then a user without administrator authority attempts to install a DB2 single-user product on the same machine.* In this scenario, the installation being attempted by the user without administrator authority will fail, and return an error message saying that the user must be an administrator to install the product.

Installation Steps

To install a DB2 client, perform the following steps:

- Step 1. Log on to the system with the user account that you want to use to perform the installation.
- Step 2. Shut down any other programs so that the setup program can update files as required.
- Step 3. Insert the appropriate CD-ROM into the drive. The auto-run feature automatically starts the setup program. The setup program will determine the system language, and launch the setup program for that language. If you want to run the setup program in a different language, or the setup program failed to start, see the following tip:



To manually invoke the setup program, perform the following steps:

- a. Click **Start** and select the **Run** option.
- b. In the **Open** field, enter the following command:

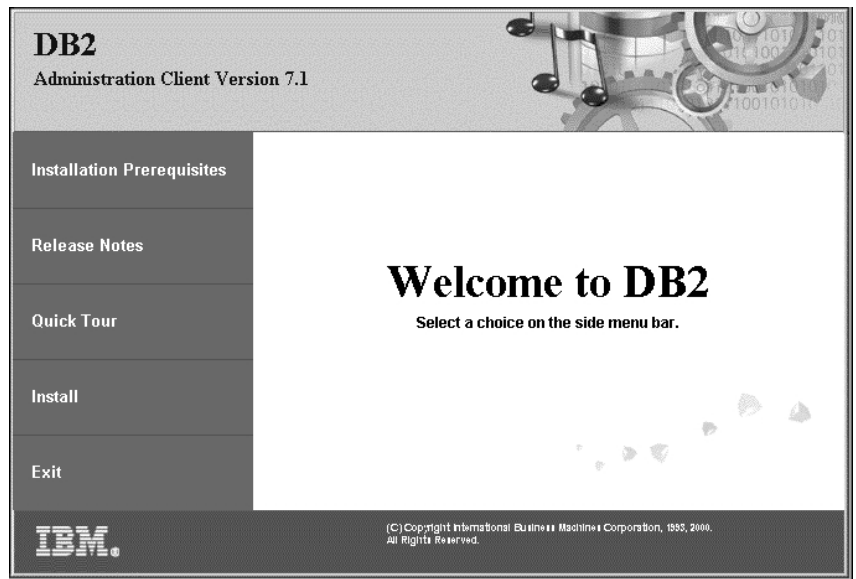
```
x:\setup /i language
```

where:

- *x*: represents your CD-ROM drive
- *language* represents the country code for your language (for example, EN for English). Table 39 on page 506 lists the code for each available language.

- c. Click **OK**.

Step 4. The DB2 Launchpad opens. It looks similar to the following:



Step 5. From this window, you can view the Installation Prerequisites and the Release Notes, you can take a Quick Tour to explore the features, capabilities, and benefits of DB2 Universal Database Version 7, or you can proceed directly to the installation.

Once you have initiated the installation, proceed by following the setup program's prompts. Online help is available to guide you through the remaining steps. Invoke the online help by clicking **Help**, or by pressing **F1** at any time. You can click **Cancel** at any time to end the installation.



For information on errors encountered during installation, see the db2.log file. The db2.log file stores general information and error messages resulting from the install and uninstall activities. By default, the db2.log file is located in the x:\db2log directory, where x: represents the drive on which your operating system is installed.

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

The setup program performs the following actions:

- Create DB2 program groups and items (or shortcuts).
- Update the Windows registry.
- Create a default client instance called DB2.



To configure your client to access remote servers, go to “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

Chapter 4. Installing DB2 Clients on OS/2 Operating Systems

This section contains the information that you need to install a DB2 client on OS/2 operating systems. If you have a pre-Version 7 DB2 client for OS/2, the WIN-OS/2 support installed will be kept at its current level.

If you want to run Windows 3.x applications on your OS/2 system, you must also install the DB2 Client Application Enabler for Windows 3.x on your system. For more information, connect to the IBM DB2 Client Application Enabler Web site at <http://www.ibm.com/software/data/db2/db2tech/clientpak.html>

Before You Begin Installing

Before you begin the installation, be sure that you have the following items and information:

1. Ensure that your system meets all of the memory, hardware, and software requirements to install your DB2 product. For more information, see “Chapter 1. Planning for Installation” on page 3.
2. A user ID to perform the installation.

If UPM is installed, the user ID you specify must have *Administrator* or *Local Administrator* authority. Create a user ID with these characteristics if necessary.

If UPM is not installed, DB2 will install it and set up the user ID USERID with password PASSWORD.

3. To verify that DB2 installed correctly, you will need to have a user account that belongs to the DB2 System Administrative (SYSADM) group, is 8 characters or less, and complies with all of DB2’s naming rules.

By default, any user that belongs to the *Local Administrators* group, on the local machine where the account is defined, has SYSADM authority on the instance. For more information, see “Working with the System Administrative Group” on page 451. For more information on valid DB2 usernames, see “Appendix F. Naming Rules” on page 509.

Installation Steps

To install a DB2 client for OS/2, perform the following steps:

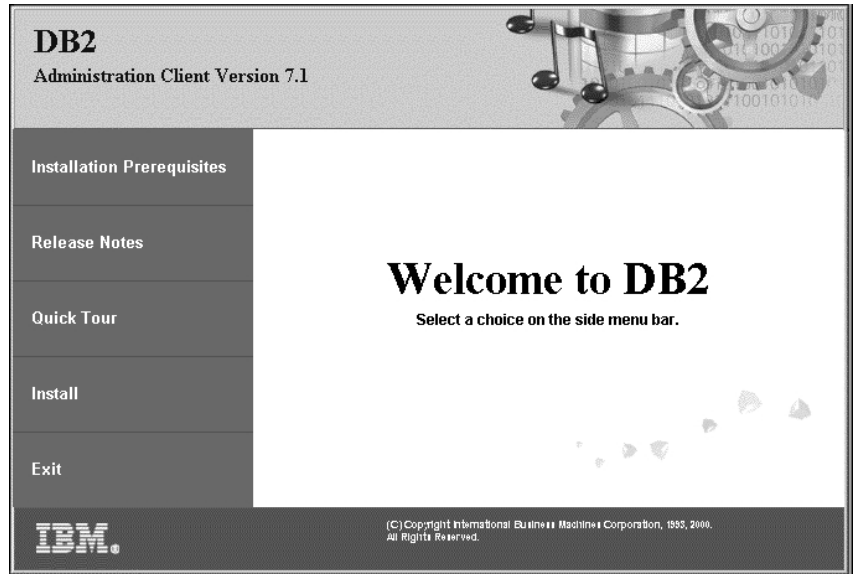
- Step 1. Insert the appropriate CD-ROM into the drive.

Step 2. Open an OS/2 command window, set the directory to your CD-ROM drive, and enter the following command:

```
x:\install
```

where *x* represents your CD-ROM drive.

Step 3. The DB2 Launchpad opens. It looks similar to the following:



Step 4. From this window, you can view the installation prerequisites and the release notes, you can take a Quick Tour to explore the features, capabilities, and benefits of DB2 Universal Database Version 7, or you can proceed directly to the installation.

Once you have initiated the installation, proceed by following the install program's prompts. Online help is available to guide you through the remaining steps. To invoke the online help, click **Help**, or press **F1**.



For information on errors encountered during installation, see the 11.log and 12.log files. These files store general information and error messages resulting from installation and uninstall activities. By default, these files are located in the `x:\db2log` directory; where *x*: represents the drive on which your operating system is installed.

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



To configure your client to access remote servers, go to "Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 29.

If you want to use ODBC applications with OS/2, you must ensure that the `\sql11ib\dll\odbc.dll` file appears as the first `odbc.dll` in the `LIBPATH` parameter of the `config.sys` file. The Install program does not automatically position the dynamic link library (dll) as of Version 7. If `odbc.dll` is not the first ODBC dll listed, you may have problems connecting to DB2 through ODBC applications.

Chapter 5. Installing DB2 Clients on UNIX Operating Systems

This section contains the information that you will need to install a DB2 client on a UNIX-based workstation.

Before You Begin

Before you begin installing a DB2 client using the `db2setup` utility, you need to ensure that your system meets all of the memory, hardware, and software requirements to install your DB2 product. For more information, see “Chapter 1. Planning for Installation” on page 3.

About the `db2setup` utility

We recommend that you use the `db2setup` utility to install DB2 products on UNIX-based systems. This utility leads you through the installation process using a simple interface that includes online help. Default values are provided for all required installation parameters, but you can also enter your own values.

The `db2setup` utility can:

- Create or assign groups and user IDs.
- Create a DB2 instance.
- Install product messages.

Note: Documentation is available in HTML format on the product CD-ROM.

If you choose to install DB2 using another method, such as using your operating system’s native administration tools, you will have to perform these tasks manually. For more information on installing DB2 manually, refer to the *DB2 for UNIX Quick Beginnings* manual.

The `db2setup` utility can generate a trace log to record errors during installation. To generate a trace log, run the `./db2setup -d` command. This generates a log in `/tmp/db2setup.trc`.

The `db2setup` utility works with Bash, Bourne, and Korn shells. Other shells are not supported.

Updating Kernel Configuration Parameters

This section applies only to DB2 clients running on HP-UX and NUMA-Q/PTX systems, and the Solaris** Operating Environment.

If you are installing a DB2 client on AIX, Linux, or SGI IRIX systems, go to “Installing Your DB2 Client” on page 27.

Before installing your DB2 client on a HP-UX, PTX or Solaris system, you may need to update your system’s kernel configuration parameters. We recommend that you set your system’s kernel configuration parameters to the values in the following sections:

- “HP-UX Kernel Configuration Parameters”
- “NUMA-Q/PTX Kernel Configuration Parameters” on page 25
- “Solaris Kernel Configuration Parameters” on page 26

HP-UX Kernel Configuration Parameters

Table 4 lists the recommended values for HP-UX kernel configuration parameters. These values are valid for DB2 clients running on HP-UX Version 11.

Note: You must reboot your machine after updating any kernel configuration parameters.

Table 4. HP-UX Kernel Configuration Parameters (Recommended Values)

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

Notes:

1. Parameters msgmnb and msgmax must be set to 65535 or higher.
2. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in Table 4.

To change a value, perform the following steps:

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

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

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

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

Step 5. Click on **OK**.

Step 6. Repeat these steps for all of the kernel configuration parameters that you want to change.

Step 7. When you are finished setting all of the kernel configuration parameters, select **Action** → **Process New Kernel** from the action menu bar.

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

After you have updated your kernel parameters, go to “Installing Your DB2 Client” on page 27 to continue with the installation.

NUMA-Q/PTX Kernel Configuration Parameters

Table 5 lists the recommended values for NUMA-Q/PTX kernel configuration parameters for DB2 clients.

Note: You must reboot your machine after updating any kernel configuration parameters.

Table 5. NUMA-Q/PTX Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Recommended Value
msgmax	65535
msgmnb	65535
msgseg	8192
msgssz	16

To modify your PTX kernel configuration parameters, perform the following steps:

Step 1. Log on as a user with root authority.

Step 2. Enter the menu command.

Step 3. Press A to select the **System Administration** option.

Step 4. Press C to select the **Kernel Configuration** option.

Step 5. Press Ctrl+F in the **Change Kernel Configuration Disk** form. If you want to build the new kernel in a disk other than the root disk, enter the disk and press Ctrl+F.

Step 6. In the **Compile, Configure, or Remove a Kernel** window, select the kernel configuration type with which your current kernel is built and press K.

Step 7. In the **Configure a kernel with site specific parameters** form, go down one page (press Ctrl+D), press A for **All** in the **Visibility level for parameter changes** and then press Ctrl+F.

Step 8. In the **Configure Files With Adjustable Parameters** window select **ALL** (press Ctrl+T) and press Ctrl+F.

- Step 9. In the Tunable Parameters window, use arrow keys to navigate. Press `Ctrl+T` to select the parameter you want to change and press `Ctrl+F`.
- Step 10. In the Detail of Parameter Expression(s) window, press `s` to set the new value.
- Step 11. In the **Add site specific 'set' parameter** form, enter the new value and press `Ctrl+F`.
- Step 12. Repeat steps 9 through 11 to change the values of all other parameters that you want to change.
- Step 13. Once you are done changing all the parameters, press `Ctrl+E` from the Tunable Parameters window.
- Step 14. Compile the kernel.
- Step 15. Press `Ctrl+X` to exit the menu.
- Step 16. Reboot the system so that the changes can take effect.

Notes:

1. The `msgmax` and `msgmnb` must be set to 65535 or larger.
2. The `msgsem` parameter must be set no higher than 32767.
3. The `shmmax` must be set to 2147483647 or larger.

After you have updated your kernel parameters, go to “Installing Your DB2 Client” on page 27 to continue with the installation.

Solaris Kernel Configuration Parameters

Table 6 lists the recommended values for Solaris kernel configuration parameters.

Note: You must reboot your machine after updating any kernel configuration parameters.

Table 6. Solaris Kernel Configuration Parameters (Recommended Values)

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

Notes:

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

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

```
set parameter_name = value
```


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

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

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

After you have updated your kernel parameters, go to “Installing Your DB2 Client” to continue with the installation.

Installing Your DB2 Client

After updating your kernel configuration parameters and rebooting your system (if necessary), you can install your DB2 client.

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

To install a DB2 client:

1. Log in as user with root authority.
2. Insert and mount the appropriate CD-ROM. For information on mounting CD-ROMs, see “Mounting CD-ROMs on UNIX Operating Systems” on page 453.
3. Change to the directory where the CD-ROM is mounted by entering the **cd /cdrom** command where **/cdrom** is the CD-ROM mount point.
4. Change to one of the following directories:

AIX	<code>/cdrom/db2/aix</code>
HP-UX Version 11	<code>/cdrom/db2/hpux11</code>
Linux	<code>/cdrom/db2/linux</code>
NUMA-Q/PTX	<code>/cdrom/db2/numaq</code>
SGI/IRIX	<code>/cdrom/db2/sgi</code>
Solaris	<code>/cdrom/unnamed_cdrom/db2/solaris</code>

5. Enter the **./db2setup** command. After a few moments the Install DB2 V7 window opens.
6. Select the product that you want to install and select **OK**.

Press **Tab** to move between available options and fields. Press **Enter** to select or deselect an option. Selected options are denoted by an asterisk.

When you select to install a DB2 product, you can choose the product’s **Customize** option to view and change the components that will be installed.

Select **OK** to continue the installation process or **Cancel** to go back to a previous window. Select **Help** for more information or assistance during the installation of any DB2 product.

When installation is complete DB2 software will be installed in the *DB2DIR* directory,

where <i>DB2DIR</i>	= /usr/lpp/db2_07_01	on AIX
	= /opt/IBMDB2/V7.1	on HP-UX, NUMA-Q/PTX, SGI IRIX, or Solaris
	= /usr/IBMDB2/V7.1	on Linux

You can use the **db2setup** program add additional products or components after your initial installation. To create or add additional DB2 products and components, enter the following command:

On AIX

```
/usr/lpp/db2_07_01/install/db2setup
```

On HP-UX, PTX, SGI IRIX, or Solaris

```
/opt/IBMDB2/V7.1/install/db2setup
```

On Linux

```
/usr/IBMDB2/V7.1/install/db2setup
```

Your next step

After installing your DB2 client, you should configure it to access a remote DB2 server. For more information see “Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor” on page 41.

Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant

This chapter describes how to configure client-to-server communications using the Client Configuration Assistant (CCA). In an LDAP-enabled environment, you may not need to perform the tasks described in this chapter.

Notes:

1. The CCA is available for DB2 clients running on OS/2 and Windows 32-bit systems.
2. LDAP support is available for Windows, AIX, and the Solaris operating environment.

LDAP Directory Support Considerations

In an LDAP-enabled environment, the directory information about DB2 servers and databases is kept in the LDAP directory. When a new database is created, the database is automatically registered in the LDAP directory. During a database connection, the DB2 client goes to the LDAP directory to retrieve the required database and protocol information and uses this information to connect to the database. There is no need to run the CCA to configure LDAP protocol information.

You may still want to use the CCA in the LDAP environment to:

- Manually catalog a database in the LDAP directory
- Register a database as an ODBC data source
- Configure CLI/ODBC information
- Remove a database cataloged in the LDAP directory

Before You Begin

When you add a database using this configuration method, the CCA will generate a default node name for the server where the database resides.

To complete the steps in this section, you should be familiar with how to start the CCA. For more information, see “Starting the Client Configuration Assistant” on page 447.

Note: To configure communications from a DB2 client to a server, the remote server must be configured to accept inbound client requests. By default, the server installation program automatically detects and configures

most protocols on the server for inbound client connections. It is recommended that you install and configure the communications protocols desired on the server before installing DB2.

If you have added a new protocol to your network, or want to modify any of the default settings, go to “Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor” on page 41.

Configuration Steps

To configure your workstation to access a database on a remote server, perform the following steps:

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

Step 2. Start the CCA. For more information, see “Starting the Client Configuration Assistant” on page 447.

The Welcome window opens each time you start the CCA, until you add at least one database to your client.

Step 3. Click on the **Add** push button to configure a connection.

You can use one of the following configuration methods:

- “Adding a Database Using a Profile”.
- “Adding a Database Using Discovery” on page 32.
- “Adding a Database Manually” on page 34.

Adding a Database Using a Profile

A server profile contains information about server instances on a system, and databases within each server instance. For information on profiles, see “Creating and Using Profiles” on page 36.

If your administrator provided you with a profile, perform the following steps:

Step 1. Select the **Use a profile** radio button and click the **Next** push button.

Step 2. Click the **...** push button and select a profile. Select a remote database from the object tree that is displayed from the profile, and if the

database selected is a gateway connection, select a connection route to the database. Click the **Next** push button.

- Step 3. Enter a local database alias name in the **Database alias** field and optionally enter a comment that describes this database in the **Comment** field. Click **Next**.
- Step 4. If you are planning to use ODBC, register this database as an ODBC data source.

Note: ODBC must be installed to perform this operation.

- a. Ensure that the **Register this database for ODBC** check box is selected.
 - b. Select the radio button that describes how you would like to register this database:
 - If you would like all users on your system to have access to this data source, select the **As a system data source** radio button.
 - If you would like only the current user to have access to this data source, select the **As a user data source** radio button.
 - If you would like to create an ODBC data source file to share database access, select the **As a file data source** radio button and enter the path and file name for this file in the **File data source name** field.
 - c. Click the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.
 - d. Click **Finish** to add the database that you selected. The Confirmation window opens.
- Step 5. Click the **Test Connection** push button to test the connection. The Connect to DB2 Database window opens.
- Step 6. In the Connect To DB2 Database window, enter a valid user ID and password for the remote database and click **OK**. If the connection is successful, a message confirming the connection appears.
- If the connection test failed, you will receive a help message. To change any settings you may have incorrectly specified, click the **Change** push button in the Confirmation window to return to the Add Database Wizard. If problems persist, refer to the *Troubleshooting Guide* for more information.
- Step 7. You are now able to use this database. Click **Add** to add more databases, or click **Close** to exit the Add Database Wizard. Click **Close** again to exit the CCA.

Adding a Database Using Discovery



This option cannot return information about DB2 systems earlier than Version 5 or any systems where an Administration Server is not running. For more information, refer to the *Administration Guide*.

You can use the Discovery feature to search a network for databases. To add a database to your system using Discovery, perform the following steps:

- Step 1. Select the **Search the network** radio button and click the **Next** push button.
- Step 2. Click the [+] sign beside the **Known Systems** icon to list all the systems known to your client.
- Step 3. Click the [+] sign beside a system to get a list of the instances and databases on it. Select the database that you want to add, click the **Next** push button, and proceed to Step 4.

If the system that contains the database that you want to add is not listed, perform the following steps:

- a. Click on the [+] sign beside the **Other Systems (Search the network)** icon to search the network for additional systems.
- b. Click on the [+] sign beside a system to get a list of the instances and databases on it.
- c. Select the database that you want to add, click **Next**, and proceed to Step 4.



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

- The Administration Server is not running on the remote system.
- The Discovery function times out. By default, the Discovery function will search the network for 40 seconds; this may not be long enough to detect the remote system. You can set the *DB2DISCOVERYTIME* registry variable to specify a longer period of time.
- The network that the Discovery request is running on is configured so that the Discovery request does not reach the remote system desired.
- You are using NetBIOS as the Discovery protocol. You may need to set the *DB2NBDISCOVERRCVBUFS* registry variable to a larger value to enable the client to receive more concurrent Discovery replies.

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

If the system that you want to add is still not listed, it can be added to the list of systems by performing the following steps:

- a. Click **Add System**. The Add System window opens.
 - b. Enter the required communication protocol parameters for the remote Administration Server and click **OK**. A new system is added. For more information, click **Help**.
 - c. Select the database that you want to add and click **Next**.
- Step 4. Enter a local database alias name in the **Database alias** field and optionally enter a comment that describes this database in the **Comment** field. Click **Next**.
- Step 5. If you are planning to use ODBC, register this database as an ODBC data source.

Note: ODBC must be installed to perform this operation.

- a. Ensure that the **Register this database for ODBC** check box is selected.
 - b. Select the radio button that describes how you would like to register this database:
 - If you would like all users on your system to have access to this data source, select the **As a system data source** radio button.
 - If you would like only the current user to have access to this data source, select the **As a user data source** radio button.
 - If you would like to create an ODBC data source file to share database access, select the **As a file data source** radio button and enter the path and file name for this file in the **File data source name** field.
 - c. Click the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.
 - d. Click **Finish** to add the database that you selected. The Confirmation window opens.
- Step 6. Click the **Test Connection** push button to test the connection. The Connect to DB2 Database window opens.
- Step 7. In the Connect To DB2 Database window, enter a valid user ID and password for the remote database and click **OK**. If the connection is successful, a message confirming the connection appears.
- If the connection test failed, you will receive a help message. To change any settings you may have incorrectly specified, click the **Change** push button in the Confirmation window to return to the Add Database Wizard. If problems persist, refer to the *Troubleshooting Guide* for more information.
- Step 8. You are now able to use this database. Click **Add** to add more databases, or click **Close** to exit the Add Database Wizard. Click **Close** again to exit the CCA.

Adding a Database Manually

If you have the information for the database you want to connect to and the server upon which it resides, you can manually enter all of the configuration information. This method is analogous to entering commands via the command line processor, however, the parameters are presented graphically for you.

To add a database to your system manually, perform the following steps:

Step 1. Select the **Manually configure a connection to a database** radio button and click **Next**.

Step 2. If you are using Lightweight Directory Access Protocol (LDAP), select the radio button that corresponds to the location where you would like your DB2 directories to be maintained:

- If you would like to maintain the DB2 directories locally, select the **Add database to your local machine** radio button and click **Next**.
- If you would like to maintain the DB2 directories globally at an LDAP server, select the **Add database using LDAP** radio button and click **Next**.

Step 3. Select the radio button that corresponds to the protocol that you want to use from the **Protocol** list.

If DB2 Connect (or the DB2 Connect Support Feature) is installed on your machine and you select TCP/IP or APPC, you can select **The database physically resides on a host or AS/400 system**. If you select this check box, you will have the option of selecting the type of connection that you want to make to the host or AS/400 database:

- To make a connection through a DB2 Connect gateway, select the **Connect to the server via the gateway** radio button.
- To make a direct connection, select the **Connect directly to the server** radio button.

Click **Next**.

Step 4. Enter the required communication protocol parameters and click **Next**. For more information, click **Help**.

Step 5. Enter the database alias name of the remote database that you want to add in the **Database name** field and a local database alias name in the **Database alias** field.

If this is a host or AS/400 database, type the Location name for an OS/390 database, the RDB name for an AS/400 database, or the DBNAME for a VSE or VM database in the **Database name** field, and optionally add a comment that describes this database in the **Comment** field.

Click **Next**.

Step 6. Register this database as an ODBC data source.

Note: ODBC must be installed to perform this operation.

- a. Ensure that the **Register this database for ODBC** check box is selected.
- b. Select the radio button that describes how you would like to register this database:
 - If you would like all users on your system to have access to this data source, select the **As a system data source** radio button.
 - If you would like only the current user to have access to this data source, select the **As a user data source** radio button.
 - If you would like to create an ODBC data source file to share database access, select the **As a file data source** radio button and enter the path and file name for this file in the **File data source name** field.
- c. Click the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.
- d. Click **Finish** to add the database that you selected. The Confirmation window opens.

Step 7. Click the **Test Connection** push button to test the connection. The Connect to DB2 Database window opens.

Step 8. In the Connect To DB2 Database window, enter a valid user ID and password for the remote database and click **OK**. If the connection is successful, a message confirming the connection appears.

If the connection test failed, you will receive a help message. To change any settings you may have incorrectly specified, click the **Change** push button in the Confirmation window to return to the Add Database Wizard. If problems persist, refer to the *Troubleshooting Guide* for more information.

Step 9. You are now able to use this database. Click **Add** to add more databases, or click **Close** to exit the Add Database Wizard. Click **Close** again to exit the CCA.

You can use the Export function of the CCA to create a client profile for an existing client configuration and use it to create identical target clients across your network. A client profile contains database connection, ODBC/CLI, and configuration information for an existing client. Use the CCA Import function to set up multiple clients across your network. Each target client will have the same configuration and settings as the existing client. For more information on creating and using client profiles, see “Creating and Using Profiles” on page 36.



You have now completed all the tasks that are involved in *Quick Beginnings* and are ready to start using DB2 Universal Database.

If you want to deploy this product using a distributed installation, go to “Chapter 19. An Introduction to Distributed Installation” on page 375.

Creating and Using Profiles

The information in this section describes how to create and use profiles to set up connections between DB2 clients and servers. To configure database connections on a DB2 client, you can use either a server profile or client profile.

Server Profiles

A server profile contains information about instances on a server system, and databases within each instance. The information for each instance includes the protocol information required to set up a client to connect to databases in that instance.



We recommend that you create a server profile only after you have created the DB2 databases that you want your remote clients to access.

To create a server profile, perform the following steps:

- Step 1. Start the Control Center. For more information, refer to “Starting the DB2 Control Center” on page 447.
- Step 2. Select the system that you want to create a profile for and right click. If the system that you want to create a profile for is not shown, select the **Systems** icon, click the right mouse button, and select the **Add** option. Click the **Help** push button and follow the online help.
- Step 3. Select the **Export Server Profile** option.
- Step 4. Enter a path and filename for this profile and select **OK**.



You are ready to use this profile on your system. For more information on how to add a database to your system using a server profile, go to “Configuration Steps” on page 30.

Client Profiles

Information in a client profile can be used to configure clients using the Import function in the Client Configuration Assistant (CCA). Clients can import all or a subset of the configuration information in a profile. The following scenario assumes that the database connections configured on one client will be exported and used to set up one or more clients.

Note: Configuration profiles can also be imported using the **db2cfimp** command. Refer to the *Command Reference* for more information.

A client profile is generated from a client using the Export function of the CCA. The information contained in a client profile is determined during the export process. Depending on the settings chosen, it can contain the existing client's:

- Database connection information (including CLI or ODBC settings).
- Client settings (including database manager configuration parameters and DB2 registry variables).
- CLI or ODBC common parameters.
- Configuration data for the local APPC or NetBIOS communications subsystem.

To create a client profile, perform the following steps:

Step 1. Start the CCA. For more information, see "Starting the Client Configuration Assistant" on page 447.

Step 2. Click **Export**. The Select Export Option window opens.

Step 3. Select one of the following export options:

- If you want to create a profile that contains all of the databases cataloged on your system, and all of the configuration information for this client, select the **All** radio button, click **OK**, and go to Step 8.
- If you want to create a profile that contains all of the databases cataloged on your system *without* any of the configuration information for this client, select the **Database connection information** radio button, click **OK**, and go to Step 8.
- If you want to select a subset of the databases that are cataloged on your system, or a subset of the configuration information for this client, select the **Customize** radio button, click **OK**, and go to the next step.

Step 4. Select the databases to be exported from the **Available databases** box and add them to the **Selected databases** box by clicking on the push button.



To add all of the available databases to the **Databases to be exported** box, click the >> button.

Step 5. Select the check boxes from the **Select custom export option** box that correspond to the options that you want to set up for the target client.

To customize settings, click on the appropriate **Customize** push button. The settings that you customize will only affect the profile to be exported, no changes will be made to your workstation. For more information, click **Help**.

Step 6. Click **OK**. The Export Client Profile window opens.

Step 7. Enter a path and file name for this client profile and click **OK**. The DB2 Message window appears.

Step 8. Click **OK**.

To import a client profile, perform the following steps:

Step 1. Start the CCA. For more information, see “Starting the Client Configuration Assistant” on page 447.

Step 2. Click **Import**. The Select Profile window opens.

Step 3. Select a client profile to import and click **OK**. The Import Profile window opens.

Step 4. You can select to import all or a subset of the information in a Client Profile. Select one of the following import options:

- To import everything in a client profile, select the **All** radio button.
- To import a specific database, or settings, that are defined in a Client Profile, select the **Customize** radio button. Select the check boxes that correspond to the options that you want to customize.

Step 5. Click **OK**.

Step 6. You are presented with a list of systems, instances, and databases. Select the database that you want to add and click **Next**.

Step 7. Enter a local database alias name in the **Database alias** field and optionally enter a comment that describes this database in the **Comment** field. Click **Next**.

Step 8. If you are planning to use ODBC, register this database as an ODBC data source.

Note: ODBC must be installed to perform this operation.

- a. Ensure that the **Register this database for ODBC** check box is selected.
- b. Select the radio button that describes how you would like to register this database:
 - If you would like all users on your system to have access to this data source, select the **As a system data source** radio button.
 - If you would like only the current user to have access to this data source, select the **As a user data source** radio button.

- If you would like to create an ODBC data source file to share database access, select the **As a file data source** radio button and enter the path and file name for this file in the **File data source name** field.
 - c. Click the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.
 - d. Click **Finish** to add the database that you selected. The Confirmation window opens.
- Step 9.** Click the **Test Connection** push button to test the connection. The Connect to DB2 Database window opens.
- Step 10.** In the Connect To DB2 Database window, enter a valid user ID and password for the remote database and click **OK**. If the connection is successful, a message confirming the connection appears.
- If the connection test failed, you will receive a help message. To change any settings you may have incorrectly specified, click the **Change** push button in the Confirmation window to return to the Add Database Wizard. If problems persist, refer to the *Troubleshooting Guide* for more information.
- Step 11.** You are now able to use this database. Click **Add** to add more databases, or click **Close** to exit the Add Database Wizard. Click **Close** again to exit the CCA.

Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor

This section describes how to configure a client to communicate with a server using the Command Line Processor (CLP).

If you plan to use an OS/2 or Windows 32-bit client to communicate with a server, the Client Configuration Assistant (CCA) makes it easy to automate configuration and administration tasks. If you have installed the CCA, it is recommended that you use this tool to configure your DB2 clients for communications. See “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29 for more information.

To configure a client to communicate with a server, the remote server must be configured to accept in-bound requests for the communications protocols that you want to use. By default, the installation program automatically detects and configures most protocols running on your server.

If you have added a new protocol to your network, or wish to change any of the default settings on the server, see the following instructions.

For instructions on entering DB2 commands, see “Entering Commands Using the Command Center” on page 448 or “Entering Commands Using the Command Line Processor” on page 449.



Go to the section that describes how to configure communication to access a remote server using the communication protocol of your choice.

- Named Pipes - see “Configuring Named Pipes on the Client” on page 42.
 - TCP/IP - see “Configuring TCP/IP on the Client” on page 47.
 - NetBIOS - see “Configuring NetBIOS on the Client” on page 56.
 - IPX/SPX - see “Configuring IPX/SPX on the Client” on page 63.
 - APPC - see “Configuring APPC on the Client” on page 71.
-

Configuring Named Pipes on the Client

This section assumes that Named Pipes is functional on the client and server workstations. See “Software Requirements” on page 4 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the supported communication protocols for your particular client and server.

Note: This protocol is only supported on Windows 32-bit operating systems.

To set up a client to use Named Pipes communications, perform the following steps:

Step 1. Identify and record parameter values.

Step 2. Configure the client:

- a. Catalog the Named Pipes node.
- b. Catalog the database.

Step 3. Test the connection between the client and server.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 7. Named Pipe Values Required at the Client

Parameter	Description	Sample Value	Your Value
Computer name (<i>computer_name</i>)	The computer name of the server machine. On the server machine, to locate the value for this parameter, click on Start and select Settings—>Control Panel . Double-click on the Network folder and select the Identification tab. Record the computer name.	server1	
Instance name (<i>instance_name</i>)	The name of the instance on the server to which you are connecting.	db2	
Node name (<i>node_name</i>)	A local alias, or nickname, that describes the node to which you are trying to connect. You can choose any name you want, however, all node name values within your local node directory must be unique.	db2node	

Step 2. Configure the Client

The following steps configure the client to communicate with the server using TCP/IP. Replace the sample values with your worksheet values.

A. Catalog the Named Pipes Node

You must add an entry to the client's node directory to describe the remote node. This entry specifies the chosen alias (*node_name*), the server's Computer name (*computer_name*), and the Instance name (*instance_name*) that the client will use to access the remote server.

To catalog the Named Pipes node, perform the following steps:

Step 1. Log on to the system with a valid DB2 user ID. For more information, see "Appendix F. Naming Rules" on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see "Working with the System Administrative Group" on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

Step 2. Catalog the node by entering the following commands:

```
db2 catalog npipe node node_name remote computer_name
      instance instance_name
terminate
```

For example, to catalog a remote node called *db2node*, which is located on the server called *server1*, in the *db2* instance, use:

```
db2 catalog npipe node db2node remote server1 instance db2
terminate
```



If you need to change values that were set with the **catalog node** command, perform the following steps:

Step 1. Run the **uncatalog node** command in the command line processor as follows:

```
db2 uncatalog node node_name
```

Step 2. Recatalog the node with the values that you want to use.

B. Catalog the Database

Before a client application can access a remote database, the database must be cataloged on the server node and on any client nodes that will connect to it. By default, when you create a database, it is automatically cataloged on the server with the database alias (*database_alias*) the same as the database name

(*database_name*). The information in the database directory, along with the information in the node directory, is used on the client to establish a connection to the remote database.

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

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

Table 8. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (<i>database_name</i>)	The database alias (<i>database_alias</i>) of the <i>remote</i> database. When you create a database, it is automatically cataloged on the server with the database alias (<i>database_alias</i>) the same as the database name (<i>database_name</i>), unless specified otherwise.	sample	
Database alias (<i>database_alias</i>)	An arbitrary local nickname for the remote database, on the client. If you do not provide one, the default is the same as the database name (<i>database_name</i>). The database alias the name that you use when connecting to a database from a client.	tor1	
Authentication (<i>auth_value</i>)	The value of the authentication required by your enterprise. Please refer to the <i>DB2 Connect User’s Guide</i> for more information on this parameter.	DCS This means that the userid and password supplied are validated at the host or AS/400 only.	

Table 8. Worksheet: Parameter Values for Cataloging Databases (continued)

Parameter	Description	Sample Value	Your Value
Node name (<i>node_name</i>)	The name of the node directory entry that describes where the database resides. Use the same value for node name (<i>node_name</i>) that you used to catalog the node in the previous step.	db2node	

Step 3. Catalog the database by entering the following commands:

```
db2 catalog database database_name as database_alias
at node node_name
db2 terminate
```

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

```
db2 catalog database sample as tor1 at node db2node
db2 terminate
```



If you need to change values that were set with the **catalog database** command, perform the following steps:

Step a. Run the **uncatalog database** command as follows:

```
db2 uncatalog database database_alias
```

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

Step 3. Test the Client-to-Server Connection

After configuring the client for communications, you will need to connect to a remote database to test the connection.

Step 1. Start the database manager by entering the **db2start** command on the server (if it was not automatically started at boot time).

Step 2. If you are using a UNIX client, run the start-up script as follows:

```
. INSTHOME/sql1lib/db2profile (for Bash, Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 3. Enter the following command on the client to connect the client to the remote database:

```
db2 connect to database_alias user userid using password
```

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

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

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

When you are finished using the database connection, enter the **command reset** command to end the database connection. The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

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

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

When you are finished using the database connection, enter the **db2 connect reset** command to end the database connection.



You are now ready to start using DB2. For more advanced topics, refer to the *Administration Guide*.

Troubleshooting the Client-to-Server Connection

If the connection fails, check the following items:

At the *server*:

- ___ 1. The *db2comm* registry value includes the value *npipe*. Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, refer to the *Administration Guide*.
- ___ 2. The security service was started. Enter the **net start db2ntsecserver** command (for Windows NT and Windows 2000 servers only).
- ___ 3. The database was created and cataloged properly.
- ___ 4. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).



If there are problems starting a protocol's connection managers, a warning message appears and the error messages are logged in the *db2diag.log* file located in the *INSTHOME/sql11ib/db2dump* directory for UNIX platforms or in the *x:\sql11ib\db2dump* directory for non-UNIX platforms.

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

At the *client*:

- ___ 1. The node was cataloged with the correct computer name (*computer_name*) and instance name (*instance_name*) of the server.
- ___ 2. The node name (*node_name*) that was specified in the database directory points to the correct entry in the node directory.
- ___ 3. The database was cataloged properly, using the *server's* database alias (the *database_alias* that was cataloged when the database was created on the server), as the database name (*database_name*) on the client.

If the connection still fails after you verify these items, refer to the *Troubleshooting Guide*.

Configuring TCP/IP on the Client

This section assumes that TCP/IP is functional on the client and server workstations. See "Software Requirements" on page 4 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 9 for the supported communication protocols for your particular client and server.

To set up TCP/IP communications on a DB2 client, perform the following steps:

Step 1. Identify and record parameter values.

Step 2. Configure the client:

- a. Resolve the server's host address.
- b. Update the services file.
- c. Catalog a TCP/IP node.
- d. Catalog the database.

Step 3. Test the connection between the client and server.



Due to the characteristics of the TCP/IP protocol, TCP/IP may not be immediately notified of the failure of a partner on another host. As a result, a client application accessing a remote DB2 server using TCP/IP, or the corresponding agent at the server, may sometimes appear to be hung. DB2 uses the TCP/IP SO_KEEPALIVE socket option to detect when there has been a failure and the TCP/IP connection has been broken.

If you are experiencing problems with your TCP/IP connection, refer to the *Troubleshooting Guide* for information on how to adjust this parameter and other common TCP/IP problems.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 9. TCP/IP Values Required at the Client

Parameter	Description	Sample Value	Your Value
Host Name	Use the <i>hostname</i> or <i>ip_address</i> of the remote server workstation.	serverhost	
• Hostname (<i>hostname</i>) or • IP address (<i>ip_address</i>)	To resolve this parameter: <ul style="list-style-type: none">• Enter the hostname command at the server to obtain the <i>hostname</i>.• Contact your network administrator to obtain the <i>ip_address</i> or enter the ping hostname command.• On UNIX systems, you can also use the DB2/bin/hostlookup hostname command, where <i>DB2</i> is the directory where DB2 is installed.	or 9.21.15.235	

of the server in the local hosts file. See Table 10 for the location of the hosts file for your particular platform.

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

Table 10. Location of the Local Hosts and Services Files

Platform	Location
OS/2	Specified by the <i>etc</i> environment variable. Enter the set etc command to determine the location of your local hosts or services files.
Windows NT or Windows 2000	Located in the <code>winnt\system32\drivers\etc</code> directory.
Windows 9x	Located in the <code>windows</code> directory.
UNIX	Located in the <code>/etc</code> directory.

Edit the client's hosts file and add an entry for the server's hostname. For example:

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

where:

9.21.15.235 represents the *ip_address*

serverhost represents the *hostname*

represents a comment describing the entry

If the server is not in the same domain as the client, you must provide a fully qualified domain name such as `serverhost.vnet.ibm.com`, where `vnet.ibm.com` is the domain name.

Step B. Update the Services File



If you are planning to catalog a TCP/IP node using a port number (*port_number*), skip this step and go to "Step C. Catalog a TCP/IP Node" on page 51.

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

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

where:

server1 represents the Connection Service name

3700 represents the connection port number. The port number specified on the client must match the port number used on the server.

tcp represents the communication protocol that you are using

represents a comment describing the entry

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

The services file is located in the same directory as the local hosts file that you may have edited in “A. Resolve the Server’s Host Address” on page 49.

See Table 10 on page 50 for the location of the services file for your particular platform.

Step C. Catalog a TCP/IP Node

You must add an entry to the client’s node directory to describe the remote node. This entry specifies the chosen alias (*node_name*), the *hostname* (or *ip_address*), and the *svcname* (or *port_number*) that the client will use to access the remote server.

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCtrl) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

Step 2. If you are using a UNIX client, run the start-up script as follows:

```
. INSTHOME/sql1lib/db2profile    (for bash, Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 3. Catalog the node by entering the following commands:

```
db2 "catalog tcpip node node_name remote [hostname|ip_address]
server [svcname|port_number]"
db2 terminate
```

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

```
db2 catalog tcpip node db2node remote serverhost server server1
db2 terminate
```

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

```
db2 catalog tcpip node db2node remote 9.21.15.235 server 3700
db2 terminate
```



If you need to change values that were set with the **catalog node** command, perform the following steps:

Step 1. Run the **uncatalog node** command in the command line processor as follows:

```
db2 uncatalog node node_name
```

Step 2. Recatalog the node with the values that you want to use.

Step D. Catalog the Database

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

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

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

Table 11. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (<i>database_name</i>)	The database alias (<i>database_alias</i>) of the <i>remote</i> database. When you create a database, it is automatically cataloged on the server with the database alias (<i>database_alias</i>) the same as the database name (<i>database_name</i>), unless specified otherwise.	sample	
Database alias (<i>database_alias</i>)	An arbitrary local nickname for the remote database, on the client. If you do not provide one, the default is the same as the database name (<i>database_name</i>). The database alias the name that you use when connecting to a database from a client.	tor1	
Authentication (<i>auth_value</i>)	The value of the authentication required by your enterprise. Please refer to the <i>DB2 Connect User's Guide</i> for more information on this parameter.	DCS This means that the userid and password supplied are validated at the host or AS/400 only.	
Node name (<i>node_name</i>)	The name of the node directory entry that describes where the database resides. Use the same value for node name (<i>node_name</i>) that you used to catalog the node in the previous step.	db2node	

Step 3. If you are using a UNIX client, run the start-up script as follows:

```
. INSTHOME/sql11b/db2profile    (for bash, Bourne or Korn shell)
source INSTHOME/sql11b/db2cshrc (for C shell)
```

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

Step 4. Catalog the database by entering the following commands:

```
db2 catalog database database_name as database_alias
at node node_name
db2 terminate
```

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

```
db2 catalog database sample as tor1 at node db2node
db2 terminate
```



If you need to change values that were set with the **catalog database** command, perform the following steps:

Step a. Run the **uncatalog database** command as follows:

```
db2 uncatalog database database_alias
```

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

Step 3. Test the Client-to-Server Connection

After configuring the client for communications, you will need to connect to a remote database to test the connection.

Step 1. Start the database manager by entering the **db2start** command on the server (if it was not automatically started at boot time).

Step 2. If you are using a UNIX client, run the start-up script as follows:

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

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

Step 3. Enter the following command on the client to connect the client to the remote database:

```
db2 connect to database_alias user userid using password
```

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

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

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

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

Troubleshooting the Client-to-Server Connection

If the connection fails, check the following items:

At the *server*:

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



Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, refer to the *Administration Guide*.

2. The services file was updated correctly.
3. The service name (*svccname*) parameter was updated correctly in the database manager configuration file.
4. The security service was started. Enter the **net start db2ntsecserver** command (for Windows NT and Windows 2000 servers only).
5. The database was created and cataloged properly.
6. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).



If there are problems starting a protocol's connection managers, a warning message appears and the error messages are logged in the *db2diag.log* file located in the *INSTHOME/sql1lib/db2dump* directory for UNIX platforms or in the *x:\sql1lib\db2dump* directory for non-UNIX platforms.

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

At the *client*:

1. If used, the services and hosts files were updated correctly.
2. The node was cataloged with the correct hostname (*hostname*) or IP address (*ip_address*).
3. The port number matches, or the services name maps to, the port number used on the server.
4. The node name (*node_name*) that was specified in the database directory points to the correct entry in the node directory.
5. The database was cataloged properly, using the *server's* database alias (the *database_alias* that was cataloged when the database was created on the server), as the database name (*database_name*) on the client.

If the connection still fails after you verify these items, refer to the *Troubleshooting Guide*.

Configuring NetBIOS on the Client

This section assumes that NetBIOS is functional on the client and server workstations. See “Software Requirements” on page 4 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the supported communication protocols for your particular client and server.

To set up a client to use NetBIOS communications, perform the following steps:

Step 1. Identify and record parameter values.

Step 2. Configure the client:

- a. Record the logical adapter number used for the NetBIOS connection.
- b. Update the database manager configuration file.
- c. Catalog the NetBIOS node.
- d. Catalog the database.

Step 3. Test the connection between the client and server.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 12. NetBIOS Values Required at the Client

Parameter	Description	Sample Value	Your Value
Logical adapter number (<i>adapter_number</i>)	The local logical adapter that will be used for the NetBIOS connection.	0	
Workstation name (<i>nname</i>) - on the client	The NetBIOS name of the <i>client</i> workstation. <i>nname</i> is chosen by the user and must be unique among all NetBIOS nodes in the network.	client1	
Workstation name (<i>nname</i>) - on the server	The NetBIOS name of the <i>server</i> workstation. Locate this parameter in the database manager configuration file on the server.	server1	

Table 12. NetBIOS Values Required at the Client (continued)

Parameter	Description	Sample Value	Your Value
Node name (<i>node_name</i>)	A local alias, or nickname, that describes the node to which you are trying to connect. You can choose any name you want, however, all node name values within your local node directory must be unique.	db2node	

Step 2. Configure the Client

The following steps configure the client to communicate with the server using TCP/IP. Replace the sample values with your worksheet values.

A. Record the Logical Adapter Number of the Local Logical Adapter Used for the NetBIOS Connection

To view and record the logical adapter number (*adapter_number*) used for the NetBIOS connection, perform the following steps:

- For OS/2:
 1. Double-click on the **OS/2 System** icon.
 2. Double-click on the **System Setup** icon.
 3. Double-click on the **MPTS Network Adapters and Protocol Services** icon.
 4. Click on the **Configure** push button.
 5. Select the **LAN adapters and protocols** radio button and click on the **Configure** push button.
 6. Record the Logical adapter number associated with the **IBM OS/2 NETBIOS** entry in the Current Configuration window.
 7. Click **Cancel**.
 8. Click **Close**.
 9. Click **Exit**.
- For Windows 32-bit operating systems:
 1. From a command prompt, enter the **regedit** command to start the Registry Editor
 2. Locate the NetBIOS adapter assignments by expanding the **HKEY_LOCAL_MACHINE** folder and locating the **Software/Microsoft/Rpc/NetBIOS** folder.
 3. Double-click on the entry that begins with **ncacn_nb_nx**, where *x* can be 0, 1, 2.. (normally you want to select the **nb0** adapter), to see that

adapter number that is associated with the NetBIOS connection. Record this setting from the **Value** field in the pop-up window.

B. Update the Database Manager Configuration File

You must update the database manager configuration file with the *client's* workstation name (*nname*) parameter.

To update the database manager configuration file, perform the following steps:

- Step 1. Log on to the system as a user with System Administrative (SYSADM) authority. For more information, see “Working with the System Administrative Group” on page 451.
- Step 2. Update the database manager configuration file with the client's Workstation name (*nname*) parameter using the following commands in the command line processor:

```
update database manager configuration using nname nname
terminate
```

For example, if the client's workstation name (*nname*) is *client1*, use:

```
update database manager configuration using nname client1
terminate
```

C. Catalog the NetBIOS Node

You must add an entry to the client's node directory to describe the remote node. This entry specifies the chosen alias (*node_name*), the remote *server's* workstation name (*nname*), and the Logical adapter number (*adapter_number*) that the client will use to access the remote DB2 server.

To catalog the NetBIOS node, perform the following steps:

- Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

- Step 2. Catalog the node by entering the following commands in the command line processor:

```
catalog netbios node node_name remote nname adapter adapter_number
terminate
```


For example, to catalog a remote database server *server1* on the node called *db2node*, using the logical adapter number *0*, use:

```
catalog netbios node db2node remote server1 adapter 0
terminate
```



If you need to change values that were set with the **catalog node** command, perform the following steps:

Step 1. Run the **uncatalog node** command in the command line processor as follows:

```
db2 uncatalog node node_name
```

Step 2. Recatalog the node with the values that you want to use.

D. Catalog the Database

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

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCtrl) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

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

Table 13. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (<i>database_name</i>)	The database alias (<i>database_alias</i>) of the <i>remote</i> database. When you create a database, it is automatically cataloged on the server with the database alias (<i>database_alias</i>) the same as the database name (<i>database_name</i>), unless specified otherwise.	sample	
Database alias (<i>database_alias</i>)	An arbitrary local nickname for the remote database, on the client. If you do not provide one, the default is the same as the database name (<i>database_name</i>). The database alias the name that you use when connecting to a database from a client.	tor1	
Authentication (<i>auth_value</i>)	The value of the authentication required by your enterprise. Please refer to the <i>DB2 Connect User's Guide</i> for more information on this parameter.	DCS This means that the userid and password supplied are validated at the host or AS/400 only.	
Node name (<i>node_name</i>)	The name of the node directory entry that describes where the database resides. Use the same value for node name (<i>node_name</i>) that you used to catalog the node in the previous step.	db2node	

Step 3. Catalog the database by entering the following commands:

```
db2 catalog database database_name as database_alias
at node node_name
db2 terminate
```

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

```
db2 catalog database sample as tor1 at node db2node
db2 terminate
```



If you need to change values that were set with the **catalog database** command, perform the following steps:

Step a. Run the **uncatalog database** command as follows:

```
db2 uncatalog database database_alias
```

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

Step 3. Test the Client-to-Server Connection

After configuring the client for communications, you will need to connect to a remote database to test the connection.

Step 1. Start the database manager by entering the **db2start** command on the server (if it was not automatically started at boot time).

Step 2. If you are using a UNIX client, run the start-up script as follows:

```
. INSTHOME/sql1lib/db2profile    (for Bash, Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 3. Enter the following command on the client to connect the client to the remote database:

```
db2 connect to database_alias user userid using password
```

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

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

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

When you are finished using the database connection, enter the **command reset** command to end the database connection. The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

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

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

When you are finished using the database connection, enter the **db2 connect reset** command to end the database connection.

Troubleshooting the Client-to-Server Connection

If the connection fails, check the following items:

At the *server*:

- __ 1. The *db2comm* registry value includes the value *netbios*.



Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, refer to the *Administration Guide*.

- __ 2. The logical adapter number is equal to 0 (or the *DB2NBADAPTERS* registry value was updated to override the default value).
- __ 3. The server's workstation name (*nname*) parameter was updated correctly in the database manager configuration file (or the admin configuration file, if you are setting up the Administration Server).
- __ 4. The network route associated with the logical adapter number is **Nbf** (for Windows NT and Windows 2000 servers only).
- __ 5. The security service was started. Enter the **net start db2ntsecserver** command (for Windows NT and Windows 2000 servers only).
- __ 6. The database was created and cataloged properly.
- __ 7. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).



If there are problems starting a protocol's connection managers, a warning message appears and the error messages are logged in the *db2diag.log* file located in the *INSTHOME/sql1lib/db2dump* directory for UNIX platforms or in the *x:\sql1lib\db2dump* directory for non-UNIX platforms.

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

At the *client*:

- __ 1. The client's workstation name (*nname*) parameter was updated correctly in the database manager configuration file.
- __ 2. The node was cataloged with the correct *server's* workstation name (*nname*) and local logical adapter number (*adapter_number*).
- __ 3. The node name (*node_name*) that was specified in the database directory points to the correct entry in the node directory.

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

If the connection still fails after you verify these items, refer to the *Troubleshooting Guide*.

Configuring IPX/SPX on the Client

This section assumes that IPX/SPX is functional on the client and server workstations. See “Software Requirements” on page 4 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the supported communication protocols for your particular client and server.

A client can access a server via Direct Addressing or File Server Addressing. See Table 14 for a list of the available IPX/SPX clients and their supported addressing methods. For a description of Direct Addressing or File Server Addressing, see “Configuring IPX/SPX on the Server” on page 147.

You need to know the IPX/SPX addressing method that was used to configure the server before you configure your DB2 client. If the server was configured to use Direct Addressing, you must configure your client to use Direct Addressing to communicate with the server. If the server was configured for File Server Addressing, you can choose to configure your client to use either Direct Addressing or File Server Addressing, provided that your client supports the method that you want to use. See Table 14 for a list of client specific IPX/SPX supported addressing methods.

Table 14. IPX/SPX Supported Communication Methods for a DB2 Client

Client Platform	Direct Addressing	File Server Addressing
OS/2	*	*
UNIX	no support	
Windows NT	*	
Windows 9x	*	
Windows 2000	*	

To set up a DB2 client to use IPX/SPX communications, perform the following steps:

Step 1. Identify and record parameter values.

Step 2. Configure the client:

- a. Catalog the IPX/SPX node.
- b. Catalog the database.

Step 3. Test the connection between the client and server.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 15. IPX/SPX Values Required at the Client

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

Table 15. IPX/SPX Values Required at the Client (continued)

Parameter	Description	Sample Value	Your Value
DB2 server object name (<i>OBJECTNAME</i>)	<p>Direct Addressing: The server's IPX/SPX internetwork address of the form: netid.nodeid.socket#</p> <p>where netid is 8 bytes, nodeid is 12 bytes, and socket# is 4 bytes.</p> <p>To resolve this parameter, enter the db2ipxad command at the server. See "A. Catalog the IPX/SPX Node" on page 66 for more information.</p> <p>File Server Addressing: The database manager server instance, represented as the object <i>OBJECTNAME</i> on the NetWare file server. The server's IPX/SPX internetwork address is stored and retrieved from this object.</p> <p>This parameter must be entered in uppercase and be unique on the NetWare file server system.</p> <p>Locate this parameter in the database manager configuration file on the server.</p>	<p>Direct Addressing</p> <p>09212700.400011527745.879E</p> <p>File Server Addressing</p> <p>DB2INST1</p>	
Node name (<i>node_name</i>)	A local alias, or nickname, that describes the node to which you are trying to connect. You can choose any name you want, however, all node name values within your local node directory must be unique.	db2node	

Step 2. Configure the Client

The following steps configure the client to communicate with the server using TCP/IP. Replace the sample values with your worksheet values.

A. Catalog the IPX/SPX Node

You must add an entry to the client's node directory to describe the remote node. This entry specifies the chosen alias (*node_name*), the file server name (*FILESERVER*), and the DB2 server object name (*OBJECTNAME*) that the client will use to access the remote DB2 server.

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see "Appendix F. Naming Rules" on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCtrl) authority on the instance. For more information, see "Working with the System Administrative Group" on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

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

```
. INSTHOME/sql1lib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

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

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

Direct Addressing Example

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

To determine the value for the *OBJECTNAME* parameter, enter the **db2ipxad** command on the server. (This command is located in the *x:\sql1lib\misc* directory, where *x*: represents the drive where your DB2 product is installed.

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

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

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

File Server Addressing Example

To catalog a remote node called *db2node* that uses the file server *NETWSRV* and the server instance objectname *DB2INST1*, enter the following commands:

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



If you need to change values that were set with the **catalog node** command, perform the following steps:

Step 1. Run the **uncatalog node** command in the command line processor as follows:

```
db2 uncatalog node node_name
```

Step 2. Recatalog the node with the values that you want to use.

B. Catalog the Database

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

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

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

Table 16. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (<i>database_name</i>)	The database alias (<i>database_alias</i>) of the <i>remote</i> database. When you create a database, it is automatically cataloged on the server with the database alias (<i>database_alias</i>) the same as the database name (<i>database_name</i>), unless specified otherwise.	sample	
Database alias (<i>database_alias</i>)	An arbitrary local nickname for the remote database, on the client. If you do not provide one, the default is the same as the database name (<i>database_name</i>). The database alias the name that you use when connecting to a database from a client.	tor1	
Authentication (<i>auth_value</i>)	The value of the authentication required by your enterprise. Please refer to the <i>DB2 Connect User's Guide</i> for more information on this parameter.	DCS This means that the userid and password supplied are validated at the host or AS/400 only.	
Node name (<i>node_name</i>)	The name of the node directory entry that describes where the database resides. Use the same value for node name (<i>node_name</i>) that you used to catalog the node in the previous step.	db2node	

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

```
. INSTHOME/sql1lib/db2profile    (for bash, Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 4. Catalog the database by entering the following commands:

```
db2 catalog database database_name as database_alias
at node node_name
db2 terminate
```

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

```
db2 catalog database sample as tor1 at node db2node
db2 terminate
```



If you need to change values that were set with the **catalog database** command, perform the following steps:

Step a. Run the **uncatalog database** command as follows:

```
db2 uncatalog database database_alias
```

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

Step 3. Test the Client-to-Server Connection

After configuring the client for communications, you will need to connect to a remote database to test the connection.

Step 1. Start the database manager by entering the **db2start** command on the server (if it was not automatically started at boot time).

Step 2. If you are using a UNIX client, run the start-up script as follows:

```
. INSTHOME/sql11ib/db2profile    (for Bash, Bourne or Korn shell)
source INSTHOME/sql11ib/db2cshrc (for C shell)
```

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

Step 3. Enter the following command on the client to connect the client to the remote database:

```
db2 connect to database_alias user userid using password
```

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

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

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

When you are finished using the database connection, enter the **command reset** command to end the database connection. The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

If the connection is successful, you will get a message showing the name of the database to which you have connected. You are now able to retrieve data

from that database. For example, to retrieve a list of all the table names listed in the system catalog table, enter the following SQL command in the Command Center or command line processor:

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

When you are finished using the database connection, enter the **db2 connect reset** command to end the database connection.

Troubleshooting the Client-to-Server Connection

If the connection fails, check the following items:

At the *server*:

- __ 1. The *db2comm* registry value includes the value *ipxspx*.



Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, refer to the *Administration Guide*.

- __ 2. The *FILESERVER*, *OBJECTNAME*, and *IPX_SOCKET* parameters were updated correctly in the database manager configuration file.
- __ 3. The database was created and cataloged properly.
- __ 4. The security service was started. Enter the **net start db2ntsecserver** command (for Windows NT and Windows 2000 servers only).
- __ 5. If you are using File Server Addressing, ensure that the DB2 server was registered at the NetWare file server *after* the database manager configuration file was updated with the required IPX/SPX parameters.
- __ 6. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).



If there are problems starting a protocol's connection managers, a warning message appears and the error messages are logged in the *db2diag.log* file located in the *INSTHOME/sql11ib/db2dump* directory for UNIX platforms or in the *x:\sql11ib\db2dump* directory for non-UNIX platforms.

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

At the *client*:

- __ 1. If you are using Direct Addressing, check that the node was cataloged with a value of * for *FILESERVER*, and the correct IPX/SPX internetwork address value for the *OBJECTNAME* parameter.
- __ 2. If you are using File Serving Addressing, check that the *FILESERVER* and *OBJECTNAME* parameters, used to catalog the node, match those configured on the server.

- 3. The node name (*node_name*) that was specified in the database directory points to the correct entry in the node directory.
- 4. The database was cataloged properly, using the *server's* database alias (the *database_alias* that was cataloged when the database was created on the server), as the database name (*database_name*) on the client.

If the connection still fails after you verify these items, refer to the *Troubleshooting Guide*.

Configuring APPC on the Client

This section describes how to configure a client workstation to communicate with a server using the APPC communication protocol, and assumes that APPC is functional on the client and server workstations. See “Software Requirements” on page 4 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the supported communication protocols for your particular client and server.



You need to ensure that the server that you are attempting to establish communications with supports APPC clients. APPC client communications are supported by AIX, OS/2, Solaris, Windows NT, and Windows 2000 servers.

The following steps are required to set up a client to use APPC communications:

- Step 1. Identify and record parameter values.
- Step 2. Configure the client:
 - a. Update the APPC profiles.
 - b. Catalog the APPC or APPN Node.
 - c. Catalog the database.
- Step 3. Test the connection between the client and server.

Step 1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Before you configure the client workstation, have your DB2 administrator and LAN administrator fill in copies of the worksheet in Table 17 on page 72 for *each* DB2 instance to which you want to connect.

After you fill in the *Your Value* entries, you can use the worksheet to configure APPC communications on the client. During the configuration process, replace

the sample values that appear in the configuration instructions with your values from the worksheet, using the boxed numbers (for example, **1**) to relate the configuration instructions to the worksheet values.

Note: The worksheet and configuration instructions supply suggested or sample values for required configuration parameters. For other parameters, use the communications program's default values. If your network configuration is different from that used in the instructions, consult your Network Administrator for values that are appropriate to your network.

Table 17. Worksheet for Planning Client to Server APPC Connections

Ref.	Name at the Client Workstation	Network or Server Name	Sample Value	Your Value
Network Elements at the Server				
1	Server name	Local Network Name	SPIFNET	
2	Partner LU Name	Local LU Name	NYX1GW0A	
3	PLU Alias		NYX1GW0A	
4	Partner Node name	Local Control Point Name	NYX1GW	
5	Database Alias		sample	
6	Mode Name		IBMRDB	
7	Connection name (Link name)		LINKSERV	
8	Remote Network or LAN address	Local Adapter or Destination Address	400009451901	
Network Elements at the Client Workstation				
9	Network ID		SPIFNET	
10	Local Control Point Name		CLI1GW	
11	(Local) LU name		CLI1GW0A	
12	(Local) LU alias		CLI1GW0A	
13	Local Node or Node ID	ID BLK	071	
14		ID NUM	27509	
15	Mode name		IBMRDB	
16	Symbolic Destination name		DB2CPIC	
17	(Remote) Transaction program (TP) name		DB2DRDA (Application TP) or 'X'X'07'6DB' (Service TP)	

Table 17. Worksheet for Planning Client to Server APPC Connections (continued)

Ref.	Name at the Client Workstation	Network or Server Name	Sample Value	Your Value
DB2 Directory Entries (at the client workstation)				
18	Node name		db2node	
19	Security		None	
20	Database name		sample	
21	Database alias		TOR1	

For each server that you are connecting to, fill in a copy of the worksheet as follows:

1. For *network ID*, determine the network name of both the server and client workstations (**1** , **3** , and **9**). Usually these values will be the same. (For example, SPIFNET.)
2. For the *partner LU name* (**2**), determine the Local LU name defined on the server for inbound connections.
3. For *partner node name* (**4**), determine the local control point name defined on the server.
4. For *database alias* (**5**), determine the name of the target database
5. For *mode name* (**6** and **15**), usually the default IBMRDB is sufficient.
6. For *remote network address* (**8**), determine the controller address or local adapter address of the target server system.
7. Determine the *local control point* name (**10**) of the client workstation. This is usually the same as the PU name for the system.
8. Determine the *local LU name* (**11**) to be used by the client workstation.
9. For *local LU alias* (**12**), you usually use the same value as for the local LU name (**11**).
10. For *symbolic destination name* (**16**), choose a suitable value.
11. For (remote) *transaction program (TP) name* (**17**), determine the transaction program name defined on the server for your APPC connections.
12. Leave the other items blank for now (**18** to **21**).

Step 2. Update the APPC Profiles

Use the completed worksheet in Table 17 on page 72 to configure DB2 client APPC communications for access to a remote DB2 Connect or DB2 Universal Database server.



Go to the sections that describe how to configure APPC communications on the platforms that are present in your network:

- “Configuring IBM eNetwork Communications Server for OS/2”
 - “Configuring IBM eNetwork Personal Communications for Windows 32-bit Operating Systems” on page 79
 - “Configuring IBM eNetwork Communications Server for Windows NT and Windows 2000” on page 83
 - “Configuring IBM eNetwork Communications Server for Windows NT SNA API Client” on page 88
 - “Configuring Microsoft SNA Server for Windows NT and Windows 2000” on page 90
 - “Configuring Microsoft SNA Client” on page 93
 - “Configuring IBM eNetwork Communications Server for AIX” on page 95
 - “Configuring Bull SNA for AIX” on page 99
 - “Configuring SNAPLus2 for HP-UX” on page 102
 - “Configuring SunLink SNA for Solaris” on page 106
-

Configuring IBM eNetwork Communications Server for OS/2

This section describes how to manually configure IBM eNetwork Communications Server V5 for OS/2 (CS/2 V5) on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC.

Before you begin, ensure that your workstation has CS/2 V5 or later installed.



The steps in this section describe how to use IBM eNetwork Communications Server for OS/2 Version 5. If you have Communications Manager for OS/2 V.1.x, the steps you perform are similar, but the interface and menu names are different.

For more information on setting up your environment, refer to the online help supplied with CS/2, or to the following publications:

- *Connectivity Supplement*
- *DRDA Connectivity Guide*

The following assumptions have been made:

- The basic installation of the IBM eNetwork Communication Server V5 for OS/2 package has already been completed.
- The DB2 client for OS/2 has been installed.

These instructions describe how to create new profiles within a new configuration. If you are modifying an existing configuration, you may need to delete some profiles before you can verify the configuration.

To configure your system, perform the following steps:

Step 1. Start a new configuration

- a. Double-click on the **IBM eNetwork Communications Server** icon.
- b. Double-click on the **Communications Manager Setup** icon.
- c. In the Communications Manager Setup window, click on the **Setup** push button.
- d. In the Open Configuration window, provide a name for a new configuration file and click on **OK**. The Communications Manager Configuration Definition window opens.

Step 2. Configure the protocol

- a. Select the **Commonly used definitions** radio button.
- b. In the Communications Definitions box, select the protocol that you want to use. These instructions assume that you are using APPC APIs over a Token-Ring connection.
- c. Click on the **Configure** push button. The APPC APIs over Token-Ring window opens.
- d. Enter your Network ID (**9**) in the **Network ID** field.
- e. Enter your local Control Point name (**10**) in the **Control point name** field.
- f. Click on the **End node** push button that your network administrator advises you to use. You can select either the **End node - to a network node server** radio button or the **End node - no network node server** radio button. A network node server is used when many users are routed through the same connection. This example assumes no network node server is used.
- g. Click on the **Advanced** push button. The Communications Manager Profile List window opens.



Subsequent steps begin from this window. You will return to this window when each of the following steps are completed.

Step 3. Prepare a LAN DLC profile

- a. On the Profile List window, select the **DLC - Token ring or other LAN Types Adapter Parameters** option and click on the **Configure** push button. The Token Ring or Other LAN Types Adapter Parameters window opens.
- b. Enter your Network ID (**9**) in the **Network ID** field.

- c. Click on **OK**.
- Step 4. Update the SNA local node characteristics
- a. On the Profile List window, select the **SNA local node characteristics** option and click on the **Configure** push button. The Local Node Characteristic window opens.
 - b. Enter your Network ID (**9**) in the **Network ID** field.
 - c. The Local node name was probably set when CS/2 was installed. If you are not sure, consult your local network administrator.
 - d. Enter your Node ID (**13** , **14**) in the **Local node ID (hex)** field. The first part should be filled in for you already when you display the profile. You only need to complete the second part.
 - e. Click on **OK**.
- Step 5. Prepare the SNA Connection Profiles
- a. On the Profile List window, select the **SNA Connections** option and click on the **Configure** push button. The Connection List window opens.
 - b. From the **Partner Type** window, select either the **To peer node** radio button (normally used for OS/400 connections), or the **To host** radio button (normally used for OS/390, MVS, VSE, and VM connections), and click on the **Create** push button. The Adapter List window opens.
 - c. Select the **Token-ring, or other LAN types** adapter type, and specify the same adapter number that you specified previously in the DLC profile.
 - d. Click on the **Continue** push button. The Connection to a Peer Node window or the Connection to a Host window opens.
- Step 6. Configure the connection in the Connection to a Peer Node or Connection to Host window
- a. Enter the Link name (**7**) in the **Link name** field.
 - b. In the Connection window, click on the **Additional Parameters** push button. The **Additional Connection Parameters** window opens.
 - c. Enter your local Control Point name (**10**) in the **Local PU name** field.
 - d. Clear the **Backup Link** check box.
 - e. Enter your Node ID (**13** and **14**) in the **Node ID** fields.
 - f. Click on **OK**.
 - g. Enter the remote LAN address (**8**) in the **LAN destination address** field.
 - h. Enter the Network ID (**1**) of the remote system in the **Partner network ID** field.

- i. Enter the Partner Node name (**4**) in the **Partner node name** field.
- j. Click on the **Define Partner LUs** push button. The Partner LU window opens.

Step 7. Create a partner LUs profile

- a. Enter the Network ID (**3**) of the remote system in the **Network ID** field.
- b. Enter the Partner LU name (**2**) in the **LU name** and **Alias** fields.
- c. Click on the **Add** push button to add the partner LU profile to the connection profile.
- d. Click on **OK**.
- e. Click on the **Additional Parameters** push button. The Additional Connection Parameters window opens.
- f. Verify that the **Multiple PU Parameters** fields are filled in. This value is the Local Node ID in hex (**13** and **14**).
- g. Click on **OK** to return to the Connection window.
- h. Click on **OK** to return to the Connections List window.
- i. Click on the **Close** push button to return to the Profile List window.

Step 8. Set SNA features

- a. On the Profile List window, select the **SNA features** option and click on the **Configure** push button. The SNA Features List window opens. Subsequent steps begin from this window.

Step 9. Prepare a local LU profile

If the DB2 client workstation is defined as an independent LU, prepare a Local LU Profile by performing the following steps:

- a. On the SNA Features List window, select **Local LUs** —> **Create** from the action menu bar.
- b. Enter your Local LU name (**11**) in the **LU name** field.
- c. Enter your Local LU alias (**13**) in the **alias** field.
- d. Select the **Independent LU** radio button in the **NAU address** box.
- e. Click on **OK**.
- f. To use this local LU when the client workstation starts the APPC connection, select the **Use this local LU as your default local LU alias** check box. By default, all APPC connections that are started from this client workstation will use this local LU.

Step 10. Prepare a mode definition

- a. From the SNA Features List box, select the **Modes** option and click on the **Create** push button. The Mode Definition window opens.

- b. Enter your Mode name (**6** , **15**) in the **mode name** field.
- c. For the other fields, you can either specify values that match the mode profile defined on your server system, or tune the parameters.
- d. Click on **OK** to finish the creation of the mode and to return to the SNA Features List panel.

Step 11. Create the CPIC side information

- a. From the SNA Features List box, select the **CPI Communications Side Information** option and click the **Create** push button. The CPI Communications Side Information window opens.
- b. Enter the Symbolic destination name (**16**) in the **Symbolic destination name** field.
- c. Select the **Alias** radio button.
- d. Click on the **Alias** drop down box and select the Partner LU alias (**12**) you defined.
- e. Enter the remote Transaction Program (TP) name (**17**) in the **Partner TP** field.
- f. Select the **None** radio button in the **Security type** group. This does not mean that you will have no security. You will specify the security type later when you update the DB2 directories.
- g. Enter the mode name (**6**) in the **Mode name** field.
- h. Click on **OK** to save the CPI side information profile and return to the SNA Features List panel.
- i. Click on **Close** to return to the Communications Server Profile List panel.

Step 12. Save the configuration

- a. Click on the **Close** button to return to the Communication Server Configuration Definition window.
- b. Click on the **Close** button to automatically verify and save the new configuration file, and exit the configuration windows.
- c. Stop and start the Communications Server by clicking on the **Stop Communications Normally** button, then on the **Start Communications** button.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring IBM eNetwork Personal Communications for Windows 32-bit Operating Systems

This section describes how to configure IBM Personal Communications for the Windows 32-bit operating systems on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC.

Before you begin, ensure that the IBM Personal Communications for Windows 32-bit operating systems you installed meets the following requirements:

- ___ 1. It is Version 4.2 or higher
- ___ 2. The IBM Personal Communications IEEE 802.2 LAN interface or LLC2 driver is installed. The LLC2 driver is installed from the IBM Communications Server installation directory. To verify this, perform the following steps:
 - a. Click **Start** and select **Settings** —> **Control Panel**.
 - b. Double-click on the **Network** icon.
 - c. Select the **Protocols** tab. The **IBM LLC2 Protocol** must be one of the protocols listed. If it is not, you need to install this protocol from your IBM Personal Communications for Windows 32-bit operating systems software media. Refer to the documentation supplied with IBM Personal Communications for instructions.

The following assumptions are made:

- The basic installation of the IBM Personal Communication package has already been completed, with the requirements listed above met.
- The DB2 client has been installed.

To start IBM Personal Communications, perform the following steps:

Step 1. Click **Start** and select **Programs** —> **IBM Communications Server** —> **SNA Node Configuration**. The IBM Personal Communications SNA Node Configuration window opens.

Step 2. Select **File** —> **New** from the menu bar. The Define the Node window opens. Subsequent steps will begin from this window.

To configure APPC communications, perform the following steps:

Step 1. Configure the Node

- a. In the **Configuration options** box, select the **Configure Node** option and click **New**. The Define the Node window opens.
- b. In the **Fully qualified CP name** fields, type in the network name (**9**) and the local control point name (**10**)
- c. Optionally, in the **CP alias** field, type in a CP alias. If you leave this blank the local control point name will be used.
- d. In the **Local Node ID** fields, type in the block ID (**13**) and the physical unit ID (**14**).
- e. Click **OK**.

Step 2. Configure the device

- a. In the **Configuration options** box, select the **Configure devices** option and click **New**.
- b. Select the appropriate DLC from **DLCs** field. These instructions in this section use the **LAN** DLC.
- c. Click **New**. The appropriate window opens with default values displayed. In our example, the Define a LAN device window opens.<
- d. Click **OK** to accept the default values.

Step 3. Configure Connections

- a. In the **Configuration options** box, the select **Configure connections** option.
- b. Ensure that the **LAN** option is selected in the **DLCs** field.
- c. Click **New**. The Define a LAN connection window opens.
- d. On the Basic tab:
 - 1) In the **Link station name** field, type in the link name (**7**).
 - 2) In the **Destination address** field, type in the remote LAN address (**8**).
- e. On the Adjacent Node tab:
 - 1) In the **Adjacent CP name** fields, type in the network ID (**3**) and the Partner node name (**4**).
 - 2) In the **Adjacent CP type** field, select **Back-level LEN**.
 - 3) Ensure that **TG number** is set to 0 (the default).
 - 4) Click **OK**.

Step 4. Configure Partner LU 6.2

- a. In the **Configuration options** box, select the **Configure partner LU** option and click **New**. The Define a partner LU 6.2 window opens.

- b. In the **Partner LU name** fields, type in the network ID (**3**) and the partner LU name (**2**).
- c. In the **Partner LU alias** field, type in the partner LU name (**2**).
- d. In the **Fully-qualified CP name** fields, type in the network ID (**3**) and the control point name (**4**).
Accept the defaults in the Advanced tab.
- e. Click **OK**.

Step 5. Configure Modes

- a. In the **Configuration options** box, select the **Configure modes** option and click **New**. The Define a mode window opens.
- b. Enter your Mode name (**15**) in the **Mode name** field of the Basic tab.
- c. Select the **Advanced** tab.
- d. Select the **#CONNECT** option from the **Class of Service Name** field.
- e. Click **OK**.

Step 6. Configure Local LU 6.2

- a. In the **Configuration options** box, select the **Configure local LU 6.2** option and click on the **New** push button. The Define a local LU 6.2 window opens.
- b. Enter your Local LU name (**11**) in the **Local LU name** field.
- c. Type in a value for the **LU session limit** field. The default, 0, specifies the maximum allowed value. Accept the defaults for the other fields.
- d. Click **OK**.

Step 7. Configure CPI-C Side Information

- a. In the **Configuration options** box, select the **Configure CPI-C side information** option and click **New**. The Define CPI-C side information window opens.
- b. In the **Symbolic destination name** field, type in the symbolic destination name (**16**).
- c. In the **Mode name** field, type in the mode name (**15**).
- d. In the **Partner LU Name** fields, type the Network ID (**3**) in the first field, and the Partner LU Name (**2**) in the second field.
- e. Specify the TP name. In the **TP name** field:
 - To specify a application TP, in the **TP name** field, type in the name of the application TP (**17**), and ensure that the **Service TP** check box *is not* selected.

- To specify a service TP, in the **TP name** field, type in the name of the service TP (**17**), and ensure that the **Service TP** check box *is* selected.

Accept the defaults for the other fields.

- f. Click **OK**.

Step 8. Save the Configuration

- a. Select **File** → **Save As** from the menu bar. The Save As window opens.
- b. Type in a file name, for example ny3.acg.
- c. Click **OK**.
- d. In the dialog box that opens, you are asked if you want this configuration to be the default. Click **Yes**.

Step 9. Update the environment

IBM Personal Communications uses an environment variable called **appc1lu** to set the default Local LU used for APPC communications. You may set this variable on a per-session basis by opening a command window and enter the **set appc1lu=local_lu_name** command, where *local_lu_name* is the name of the local LU you want to use. However, you will probably find it more convenient to permanently set the variable. To permanently set the variable in Windows NT or Windows 2000, perform the following steps:

- a. Click **Start** and select **Settings** → **Control Panel**.
- b. Double-click on the **System** icon. The System Properties window opens.
- c. Select the **Environment** tab.
- d. Type appc1lu in the **Variable** field.
- e. Type your local LU name (**11**) in the **Value** field.
- f. Click **Set** to accept the changes.
- g. Click **OK** to exit the System Properties window.

The environment variable will now remain set for future sessions.

Step 10. Start SNA Node Operations

- a. Click **Start** and select **Programs** → **IBM Personal Communications** → **Administrative and PD Aids** → **SNA Node Operations**. The Personal Communications SNA Node Operations window opens.
- b. From the menu bar, select **Operations** → **Start Node**.
- c. In the window that opens, select the configuration file you saved in the previous step (for example, ny3.acg) and click **OK**.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring IBM eNetwork Communications Server for Windows NT and Windows 2000

This section describes how to configure IBM eNetwork Communications Server for Windows NT (CS/NT) on a DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server.

Note: References to Windows NT and CS/NT in this section also apply to Windows 2000.

Before you begin, ensure that the IBM eNetwork Communications Server for Windows NT (CS/NT) you installed meets the following requirements:

- 1. It is Version 5.0 or higher if you are planning to update multiple databases within the same transaction. If you are planning to use 2-phase commit, then Version 5.01 of CS/NT is required
- 2. The IBM Communications Server IEEE 802.2 LAN interface or LLC2 driver was installed. The LLC2 driver was installed from the CS/NT installation directory. During installation CS/NT asks if you want to install LLC2. If you are not sure whether LLC2 was installed with your copy of CS/NT, you can find out as follows:
 - a. Click on the **Start** push button and select **Settings** —> **Control Panel**.
 - b. Double-click on the **Network** icon.
 - c. On the Network window, select the **Protocols** tab. The **IBM LLC2 Protocol** must be one of the protocols listed. If it is not, you need to install this protocol from your IBM Communications Server for Windows NT software media. Refer to documentation supplied with CS/NT for instructions.
- 3. The APAR fixes JR11529 and JR11170 were applied. These fixes are required to enable cancelling of queries in progress by using Ctrl-BREAK or issuing the SQLCancel ODBC/CLI call.

To start IBM eNetwork Communications Server, perform the following steps:

- Step 1. Click **Start** and select **Programs** → **IBM Communications Server** → **SNA Node Configuration**. The IBM Communications Server SNA Node Configuration window opens.
- Step 2. Select **File** → **New** → **Advanced** from the menu bar. The Configuration options window opens. Subsequent steps will begin from this window.

To configure IBM eNetwork Personal Server for APPC communications, perform the following steps:

- Step 1. Configure Node
 - a. In the **Configuration options** field, select the **Configure Node** option and click on the **New** push button. The Define the Node window opens.
 - b. In the **Fully qualified CP name** fields, type in the network name (**9**) and the local control point name (**10**).
 - c. Optionally, in the **CP alias** field, type in a CP alias. If you leave this blank the local control point name will be used.
 - d. In the **Local Node ID** fields, type in the block ID (**13**) and the physical unit ID (**14**).
 - e. Select the appropriate node type. The default is to select the **End Node** radio button.
 - f. Click on **OK**.
- Step 2. Configure the device
 - a. In the **Configuration options** box, select the **Configure devices** option and click on the **New** push button. The appropriate window opens with default values displayed.
 - b. Select the appropriate DLC from **DLCs** field. The instructions in this section assume that you are using the **LAN** DLC.
 - c. Click on **OK** to accept the default values.
- Step 3. Configure Connections
 - a. In the **Configuration options** box, select the **Configure connections** option and click on the **New** push button.
 - b. Ensure that in the **DLCs** field, the **LAN** option is selected.
 - c. Click on the **New** button. The Define a LAN connection window opens.
 - d. On the Basic tab:
 - 1) In the **Link station name** field, type in the link name (**7**).
 - 2) In the **Destination address** field, type in the remote LAN address (**8**).
 - e. On the Security tab:

- 1) In the **Adjacent CP name** fields, type in the network ID (**3**) and the Control Point name (**4**).
- 2) In the **Adjacent CP type** field, select the appropriate CP type (eg. **Back-level LEN**).
- 3) Ensure that **TG number** is set to 0 (the default).
- 4) Click on **OK**.

Step 4. Configure Partner LU 6.2

- a. In the **Configuration options** box, select the **Configure partner LU** option and click on the **New** push button. The Define a partner LU 6.2 window opens.
- b. In the **Partner LU name** fields, type in the network ID (**3**) and the partner LU name (**2**).
- c. In the **Partner LU alias** field, type in the partner LU name (**2**).
- d. If you are configuring Communications Server for SNA Clients, in the **Fully-qualified CP name** fields, type in the network ID (**3**) and the adjacent control point name (**4**).
Leave the other fields blank.
- e. Click on **OK**.

Step 5. Configure Modes

- a. In the **Configuration options** box, select the **Configure modes** option and click on the **New** push button. The Define a mode window opens.
- b. In the **Mode name** field, type in the mode name (**6**).
- c. Select the **Advanced** tab and ensure that the **Class of Service Name** is set to **#CONNECT**.
Accept the defaults for the other fields.
- d. Click on **OK**.

Step 6. Configure Local LU 6.2

- a. In the **Configuration options** box, select the **Configure local LU 6.2** option and click on the **New** push button. The Define a local LU 6.2 window opens.
- b. In the **Local LU name** field, type in the local LU name (**11**).
- c. Type in a value for the **LU session limit** field. The default, 0, specifies the maximum allowed value.
Accept the defaults for the other fields.
- d. Click on **OK**.

Step 7. Configure CPI-C Side Information

- a. In the **Configuration options** box, select the **Configure CPI-C side information** option and click on the **New** push button. The Define CPI-C side information window opens.

- b. In the **Symbolic destination name** field, type in the symbolic destination name (**16**).
- c. In the **Mode name** field, type in the mode name (**15**).
- d. Select the **Use Partner LU alias** radio button and select a Partner LU alias.
- e. Specify the TP name. In the **TP name** field:
 - To specify a application TP, in the **TP name** field, type in the name of the application TP (**17**), and ensure that the **Service TP** check box *is not* selected.
 - To specify a service TP, in the **TP name** field, type in the name of the service TP (**17**), and ensure that the **Service TP** check box *is* selected

Accept the defaults for the other fields.

- f. Click on **OK**.

Step 8. Save the Configuration

- a. Select **File** → **Save as** from the menu bar. The Save As window opens.
- b. Type in a file name, for example ny3.acg
- c. Click on **OK**.
- d. In the window that opens you are asked if you want this configuration to be the default. Click on the **Yes** push button.

Step 9. Update the Environment

CS/NT uses an environment variable called *appclu* to set the default APPC Local LU. You may set this variable on a per-session basis by opening a command window and typing **set appclu=local_lu_name** command, where *local_lu_name* is the Local LU name; however, you will probably find it more convenient to permanently set the variable. In order to permanently set the variable in Windows NT, perform the following steps:

Step a. Click the **Start** push button and select **Settings** → **Control Panel**.

Step b. Double-click on the **System** icon. The System Properties window opens.

Step c. Select the **Environment** tab.

Step d. Type appclu in the **Variable** field, and type your local LU name (**11**) in the **Value** field.

Step e. Click on **Set** push button to accept the changes

Step f. Click on **OK**.

The environment variable will now remain set for future sessions.

Step 10. Start SNA Node Operations

To start SNA node operations on your machine, perform the following steps:

Step a. Click on the **Start** push button and select **Programs** → **IBM Communications Server** → **SNA Node Operations**. The **SNA Node Operations** window opens.

Step b. Select **Operations** → **Start Node** from the menu bar. In the dialog box that opens, select the configuration file you saved at the end of Step 2 (in our example, *ny3.acg*).

Step c. Click **OK**.

SNA node operations will now begin running.

Step 11. Registering Communications Server as a Windows NT Service

To automatically start Communications Server when the machine is started, you can register it as a Windows NT Service.

To register Communications Server as an NT service execute one of the following commands:

```
csstart -a  
(to register Communications Server with default configuration)
```

or:

```
csstart -a c:\ibmcs\private\your.acg
```

where *c:\ibmcs\private\your.acg* represents the fully qualified name of the non-default Communications Server configuration file you wish to use.

Whenever your machine is booted in the future, Communications Server will be started automatically with the required configuration file.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring IBM eNetwork Communications Server for Windows NT SNA API Client

Read this section if you have a Windows NT workstation that has IBM eNetwork Communications Server for Windows NT SNA API Client Version 5.0 or higher installed and you want to connect to an IBM eNetwork Communications Server for Windows NT server.

Note: References to Windows NT in this section also apply to Windows 2000.

The Communications Server for Windows NT server and its SNA API client act as a split client. This configuration requires that you have an APPC-enabled application (such as DB2 client) running on the SNA API client workstation.



The instructions in this section use a Windows NT client. The instructions for other supported operating systems are similar; refer to your Communications Server for Windows NT documentation for more information.

To configure the Windows NT SNA API client for APPC communications, perform the following steps:

- Step 1. Create a user account for the SNA API client on the Communications Server for Windows NT server
 - a. Click on the **Start** button and select **Programs** —> **Administrative Tools (Common)** —> **User Manager**. The User Manager window opens.
 - b. Select **Users** —> **New User** from the menu bar. The New User window opens.
 - c. Fill in the fields for the new SNA client user account. For more information, refer to the Windows NT online help.
 - d. Ensure that this user account is a member of the *Administrators*, *IBMCSADMIN*, and *IBMCSAPI* groups. To add this user account to these groups, perform the following steps:
 - 1) Click on the **Groups** push button
 - 2) Select a group from the **Not member of** box and click on the <- **Add** push button. Repeat this step for each group that your user account must belong to.
 - e. Click on **OK**.
 - f. Click on the **Add** push button.
- Step 2. Start the configuration GUI for the IBM eNetwork CS/NT SNA API Client. Click on the **Start** button and select **Programs** —> **IBM Communications Server SNA Client** —> **Configuration**. The CS/NT SNA Client Configuration window opens.

Step 3. Configure Global Data

- a. In the **Configuration options** box, select the **Configure Global Data** option and click on the **New** push button. The Define Global Data window opens.
- b. Enter the user name for the SNA API client in the **User name** field.
- c. Enter the password for the user account in the **Password** and **Confirm Password** fields.
- d. Click on **OK**.

Step 4. Configure APPC Server List

- a. In the **Configuration options** box, select the **Configure APPC Server List** option and click on the **New** push button. The Define APPC Server list window opens.
- b. Type in the IP address of the server (for example, 123.123.123.123).
- c. Click on **OK**.

Step 5. Configure CPI-C Side Information

- a. In the **Configuration options** box, select the **Configure CPI-C side information** option and click on the **New** push button. The Define CPI-C side information window opens.
- b. Enter the symbolic destination name (**16**) in the **Symbolic destination name** field.
- c. Enter your Local LU alias (**12**) in the **Local LU alias** field.
- d. Enter the mode name (**15**) in the **Mode name** field.
- e. Enter the transaction program name (**17**) in the **TP name** field.
- f. Select the **For SNA API Client use** check box for this transaction program.
- g. Enter the network ID (**3**) and partner LU name (**2**) in the **Partner LU name** field.
- h. Click on **OK**.

Step 6. Save the Configuration

- a. Select **File** → **Save As** from the menu bar. The Save As window opens.
- b. Type in a file name and click on **Save** push button.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring Microsoft SNA Server for Windows NT and Windows 2000

This section describes how to configure Microsoft SNA Server Version 4.0 for Windows NT on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC. Although Microsoft SNA Server will run on Windows NT 4.0 Workstation, Windows NT 4.0 Server is recommended.

Note: References to Windows NT in this section also apply to Windows 2000.



For instructions on how to configure Microsoft SNA Client for Windows NT, go to “Configuring Microsoft SNA Client” on page 93.

You can define the properties of your SNA connections in the Microsoft SNA Server Manager (Server Manager). The Server Manager uses an interface similar to that of the Windows NT Explorer. There are two panes in the main window of the Server Manager. All the configuration options we will be using can be accessed by right-clicking on objects in the left-hand pane of the window. Every object has a *context menu* that you can access by right-clicking on the object.

To configure APPC communications for use by the DB2 client using Microsoft SNA Server Manager, perform the following steps:

- Step 1. Start the Server Manager by clicking on the **Start** button and selecting **Programs** → **Microsoft SNA Server** → **Manager**. The Microsoft SNA Server Manager window opens.
- Step 2. Define the control point name
 - a. Open the Servers folder in the left pane of the Server Manager by clicking on the [+] sign beside the **Servers** folder.
 - b. Right-click on the **SNA Service** folder and select the **Properties** option. The Properties window opens.
 - c. Enter the correct **NETID** (**9**) and **Control Point Name** (**10**) in the corresponding fields.

d. Click on **OK**.

Step 3. Define the link service (802.2)

- a. Right-click on the **SNA Server** icon and Select **Insert** → **Link Service** from the menu bar. The Insert Link Service window opens.
- b. Select **DLC 802.2 Link Service**.
- c. Click on the **Add** push button.
- d. Click on the **Finish** push button.

Step 4. Define the connection properties

- a. Right-click on the **SNA Service** icon and select **Insert** → **Connection** → **802.2** from the menu bar. The Connection Properties window opens.
- b. Enter a connection name (**7**) in the **Name** field under the **General** tab.
- c. Click on the **Link Service** drop-down box and select the **SnaDlc1** option.
- d. Select the **Remote End** radio button from the **Host System** group.
- e. Select the **Both Directions** radio button from the **Allowed Directions** group.
- f. Select the **On Server Startup** radio button from the **Activation** group.
- g. Select the **Address** tab.
- h. Fill in the **Remote Network Address** field (**8**). Accept the default numbers in the other fields.
- i. Select the **System Identification** tab.
- j. Enter the following information:
 - 1) For the **Local Node Name**, add the **Network ID** (**9**), the **Local PU Name** (**10**), and the **Local Node ID** (**13** plus **14**). Accept the **XID Type** default.
 - 2) For the **Remote Node Name**, add the **NETID** (**1**) and the **Control Point Name** (**4**). Accept the other defaults
- k. Click on **OK**.

Step 5. Define a local LU

- a. Right-click on the **SNA Service** icon and select the **Insert** → **APPC** → **Local LU** option. The Local APPC LU Properties window opens.
- b. Enter the following information:
 - The **LU Alias** (**12**).
 - The **NETID** (**9**).
 - The **LU Name** (**11**).

- c. Select the **Advanced** tab.
- d. Select the **Member of Default Outgoing Local APPC LU Pool** option. Accept the other defaults.
- e. Click on **OK**.

Step 6. Define a remote LU

- a. Right-click on the **SNA Services** icon and select the **Insert** → **APPC** → **Remote LU** option. The Remote APPC LU Properties window opens.
- b. Click on the **Connection** drop down box and select the appropriate connection name (**7**).
- c. Enter the partner LU name (**2**) in the **LU Alias** field.
- d. Enter the Network ID (**1**) in the **Network Name** field.

The other fields will be filled in by the program. If your LU alias is not the same as your LU Name, make sure you specify the LU Name in the appropriate field. The program will fill it in automatically, but it will be incorrect if the alias and the name are not the same.

Step 7. Click on **OK**.

Define a mode

- 1. Right-click on the **APPC Modes** folder and select the **Insert** → **APPC** → **Mode Definition** option. The APPC Mode Properties window opens.
- 2. Enter the Mode name in the **Mode Name** field (**6**).
- 3. Select the **Limits** tab.
- 4. Enter appropriate numbers for the **Parallel Session Limit** and **Minimum Contention Winner Limit** fields. Your Server-Side or LAN administrator should be able to supply you with the numbers if you do not know which limits to use.
- 5. Accept the other defaults, and click on **OK**.

Define the CPIC name properties

- 1. Right-click on **CPIC Symbolic Name** folder icon and select the **Insert** → **APPC** → **CPIC Symbolic Name** option. The CPIC Name Properties window opens.
- 2. Enter the Symbolic Destination Name (**16**) in the **Name** field.
- 3. Click on the **Mode Name** drop down box and select a mode name, for example, *IBMRDB* (**15**).
- 4. Select the **Partner Information** tab.
- 5. Select the **SNA Service TP (in hex)** radio button, in the **Partner TP Name** box, and enter the Service TP name (**17**), or select the **Application TP** radio button and enter the Application TP name (**17**).
- 6. In the **Partner Lu Name** box, select the **Fully Qualified** radio button.

7. Enter the fully-qualified Partner LU Name (**1** and **2**).
8. Click on **OK**.

Save the configuration.

1. Select **File** → **Save** from the menu bar of the Server Manager window. The Save File window opens.
2. Type a unique name for your configuration into the **File Name** field.
3. Click on the **Save** push button.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring Microsoft SNA Client

This section provides step-by-step instructions for setting up communications between your Microsoft SNA Client workstation and a Windows workstation that has Microsoft SNA Server V4.0 (or later) installed.

Note: References to Windows in this section apply to Windows NT and Windows 2000.



For instructions on how to configure Microsoft SNA Server Version 4.0 for Windows on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC, go to “Configuring Microsoft SNA Server for Windows NT and Windows 2000” on page 90.

The rest of this section assumes:

1. The Microsoft SNA Server has already been configured for APPC communications to connect to a DB2 Connect or DB2 Universal Database server using APPC. Refer to the Microsoft SNA Server documentation for more information.
2. Microsoft SNA Client Version 2.11 is not already installed on your client workstation.

To configure the Microsoft SNA client, perform the following steps:

Step 1. Obtain Required Information: For your Microsoft SNA client software to function properly you must have access to a properly-configured Microsoft SNA Server. Request that your SNA Server administrator:

- Step 1. Obtain the proper license for you to use Microsoft SNA Client on your workstation.
- Step 2. Define a user ID and password for you on the SNA Server domain.
- Step 3. Define connections to server databases that you need to access, as described in "Configuring Microsoft SNA Server for Windows NT and Windows 2000" on page 90.
- Step 4. Provide you with the symbolic destination name (**16**), database name (**5**), and user account to use for each database connection defined in the previous step.

If you plan to change server passwords, the SNA administrator will also need to provide you with symbolic destination names for password management tasks on each server.

- Step 5. Provide you with the Microsoft SNA Server domain name and the protocol used for communicating with the SNA server (TCP/IP, NetBEUI, IPX/SPX).

Step 2. Install the Microsoft SNA Client on the DB2 Client Workstation:

Obtain the Microsoft SNA Client software, and follow the instructions provided with it to start the installation program. When you reach the Optional Components window, *deselect* the Install ODBC/DRDA driver check box.

Step 3. Install the DB2 client:

- Step 1. Click on the **Start** button and select **Programs—> DB2 for Windows —> Client Configuration Assistant**.
- Step 2. You need to provide the following information:
 - ___ a. The Symbolic destination name (**16**) defined at the Microsoft SNA Server for the Partner LU (**2**) of the target database server.
 - ___ b. The real database name (**5**).



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring IBM eNetwork Communications Server for AIX

This section describes how to configure IBM eNetwork Communication Server V5.0.2.5 for AIX on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC. IBM eNetwork Communication Server for AIX is the only SNA product supported for DB2 Connect running on RS/6000 machines.

Before you begin, ensure that your workstation has IBM eNetwork Communication Server V5.0.2.5 for AIX (CS/AIX) installed. If you need further information in order to configure your SNA environment, refer to the online help provided with CS/AIX.

The following assumptions are made:

- The basic installation of the IBM eNetwork Communication Server V5 for AIX package has already been completed.
- The DB2 client has been installed.
- The user is logged on to the system as a user with root authority.

To configure CS/AIX for use by a DB2 client, log on to the system as a user with root authority and use either the `/usr/bin/snaadmin` tool or the `/usr/bin/X11/xsnaadmin` tool. Information about these programs can be found in the system documentation. To use the `xsnaadmin` program to configure CS/AIX, perform the following steps:

- Step 1. Enter the `xsnaadmin` command. The Node window for the server opens.
- Step 2. Define a Node
 - a. Select **Services** → **Configure Node Parameters** from the menu bar. The Node Parameters window opens.
 - b. Click on the **APPN support** drop down box and select the **End node** option.
 - c. Enter your Network ID and the Local PU Name (**9** and **10**) in the **Control point name** fields.

- d. Enter Local PU Name (**10**) in the **Control point alias** field.
- e. Enter your Node ID (**13** and **14**) in the **Node ID** fields.
- f. Click on **OK**.

Step 3. Define a port

- a. Select the **Connectivity and Dependent LUs** window.
- b. Click on the **Add** push button. The Add to Node window opens.
- c. Select the **Port using** radio button.
- d. Click on the **Port using** drop down box and select the appropriate port type. For our example, we will select the **Token ring card** option.
- e. Click on **OK**. The Port window for the chosen port type opens.
- f. Enter a name for the port in the **SNA port name** field.
- g. Select the **Initially active** check box.
- h. Select the **Define on connection network** check box.
- i. Enter your SNA Network Name (**9**) in the first part of the **CN name** field.
- j. Enter the Control Point Name (**10**) associated with your AIX computer in the second part of the **CN name** field.
- k. Click on **OK**. The **Port** window closes and a new port appears in the **Connectivity and Dependent LUs** window.

Step 4. Define a link station

- a. In the **Connectivity and Dependent LUs** window, select the port you defined in the previous step.
- b. Click on the **Add** push button. The Add to Node window opens.
- c. Select the **Add a link station to port** radio push button.
- d. Click on **OK**. The Token ring link station window opens.
- e. Enter a name for the link in the **Name** field.
- f. Click on the **Activation** drop down box and select the **On demand** option
- g. Select the **Independent only** radio button in the **LU traffic** box.
- h. In the **Independent LU traffic** group box:
 - 1) Enter the Network ID (**3**) and the Partner LU Name (**2**) in the **Remote Node** fields.
 - 2) Click on the **Remote node type** drop down box and select the type of node that applies to your network.
- i. In the **Contact information** box, enter the SNA Destination Address (**8**) assigned to the DB2 server in the **Mac address** field.

- j. Click on **OK**. The Link Station window is closed and a new link station appears as a child of the port in the **Connectivity and Dependent LUs** window.

Step 5. Define a local LU

- a. Select the **Independent local LUs** window.
- b. Click on the **Add** push button. The Local LU window opens.
- c. Enter your independent local LU Name (**11**) in the **LU name** field.
- d. Enter the local LU alias in the **LU alias** field (**12**).
- e. Click on **OK**. The new LU appears in the **Independent local LUs** window.

Step 6. Define a partner LU over the link station

- a. Select **Services** → **APPC** → **New Partner LUs** → **Partner LU on link station** from the menu bar. The Partner LU on link station window opens.
- b. Enter the name for the Local LU (**11**) you defined previously in the **LU name** field.
- c. Enter the name for the Link station you defined previously in the **LS name** field.
- d. Enter the name of the Partner LU (**1** + **2**) you want to connect to in the **Partner LU name** fields.
- e. Click on **OK**. The Partner LU opens in the **Independent Local LUs** window of the Local LU that was created in the previous step.

Step 7. Define an alias for the partner LU

- a. Select the **Remote Systems** window.
- b. Click on the **Add** push button. The Add to node window opens.
- c. Select the **Define partner LU alias** radio push button.
- d. Click on **OK**. The Partner LU window opens.
- e. Enter an alias for the partner LU in the **Alias** field.
- f. Enter the same value in the **Uninterpreted name** field.
- g. Click on **OK**.

Step 8. Define a mode

- a. Select **Services** → **APPC** → **Modes** from the menu bar. The Modes window opens.
- b. Click on the **New** push button. The Mode window opens.
- c. Enter a mode name (**15**) in the **Name** field.
- d. The configuration values below are suggested for the following fields:

- **Initial Session limits:** 20
- **Maximum Session limits:** 32767
- **Min con. winner sessions:** 10
- **Min con. loser sessions:** 10
- **Auto-activated session:** 4
- **Initial Receive pacing window:** 8



These values are suggested because they are known to work. You may need to tailor these values so that they are optimized for your particular environment.

- e. Click on **OK**. The new mode appears in the Modes window.
- f. Click on **Done**.

Step 9. Define the CPI-C destination name

- a. Select **Services** → **APPC** → **CPI-C** from the menu bar. The CPI-C destination names window opens.
- b. Click on the **New** push button. The CPI-C destination window opens.
- c. Enter the Symbolic Destination Name (**16**) you want to associate with the server's database in the **Name** field.
- d. In the **Partner LU and mode** box:
 - 1) Select the **Use PLU Alias** field and enter the Partner LU Alias (**2**) you created in a previous step.
 - 2) Enter the Mode name (**15**) for the mode you created in a previous step in the **Mode** field.
- e. Select the type of conversation level security you want to use in the **Security** group box. This is usually None.
- f. Click on **OK**. The new destination name appears in the Destination names window.
- g. Click on the **Done** push button.

Step 10. Test the APPC connection

- a. Start the SNA subsystem by entering the **/usr/bin/sna start** command. You can enter the **/usr/bin/sna stop** command to stop the SNA subsystem first, if required.
- b. Start the SNA administration program. You can enter either the **/usr/bin/snaadmin** command or the **/usr/bin/X11/xsnaadmin** command.
- c. Start the subsystem node. Select the appropriate node icon in the push button bar, and click on the **Start** push button.

- d. Start the link station. Select the link station you defined previously in the **Connectivity and Dependent LUs** window, and click on the **Start** push button.
- e. Start the session. Select the LU you defined previously in the **Independent Local LUs** window, and click on the **Start** push button. A session activation window opens.
- f. Select or enter the Partner LU and Mode desired
- g. Click on **OK**.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring Bull SNA for AIX

This section describes how to configure Bull DPX/20 SNA/20 Server on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC. If Bull DPX/20 SNA/20 Server is installed prior to installing the DB2 client, the client uses Bull SNA. Otherwise, you need to configure DB2 Connect to work with IBM eNetwork Communications Server V5.0.2.5 for AIX. See “Configuring IBM eNetwork Communications Server for AIX” on page 95 for more information.

To determine if Bull SNA is installed on your AIX 4.2 (or later) system, enter the following command:

```
ls1pp -l express.exsrv+dsk
```

If Bull SNA is installed, you will see output similar to the following:

Fileset	Level	State	Description
Path: /usr/lib/objrepos express.exsrv+dsk	2.1.3.0	COMMITTED	EXPRESS SNA Server and Integrated Desktop

If you install Bull SNA after installing the DB2 client and you want the client to use Bull SNA instead of IBM eNetwork Communications Server for AIX, log on to the system as a user with root authority and enter the following command:

```
/usr/lpp/db2_06_01/cfg/db2cfgos
```

If you want to install the Bull DPX/20 SNA/20 Server, then you must have the following software:

- __ 1. AIX V4.2
- __ 2. Express SNA Server V2.1.3

For more information on setting up your SNA environment, refer to the Bull DPX/20 SNA/20 Server Configuration Guide.



DB2 Connect, when used with the Bull SNA server, cannot have inbound APPC connections from remote clients. The only APPC connections it can have are outbound APPC connections to the host.

To configure Bull SNA for use by DB2 Connect, enter the **express** command to configure the following SNA parameters:

Config	Express	Default configuration for EXPRESS
Node	CLI1	SPIFNET.CLI1 (HOSTNAME=CLI1)
Indep. LUs	6.2 LUs Using All	Stations
LU	CLI1GW	Control Point LU
Link	tok0.00001	Link (tok0)
Station	SERV	To SERV from CLI1
LU	CLI1GW0A	To SERV from CLI1
LU Pair	NYX1GW0A	To SERV from CLI1
Mode	IBMRDB	IBMRDB

Use default values for fields not listed.

The following example illustrates the sample configuration:

Defining hardware:

```
System (hostname) = CLI1
Adapter and Port  = CLI1.tok0
MAC Address      = 400011529778
```

Defining SNA node:

```
Name           = CLI1
Description    = SPIFNET.CLI1 (HOSTNAME=CLI1)
Network ID     = SPIFNET
Control Point  = CLI1GW
XID Block     = 071
XID ID        = 27509
```

Defining token ring link:

```
Name           = tok0.00001
Description    = Link (tok0)
Connection Network name
Network ID     = SPIFNET
Control Point  = NYX1GW
```

Defining token ring station:

```
Name           = SERV
Description    = To SERV from CLI1
Remote MAC address = 400009451901
Remote Node name
  Network ID   = SPIFNET
  Control Point = NYX1GW
```

Defining Local LU 6.2:

```
Name           = CLI1GW0A
Description    = To SERV from CLI1
Network ID    = SPIFNET
LU name       = CLI1GW0A
```

Defining Remote LU 6.2:

```
Name           = NYX1GW0A
Description    = To SERV from NYX1
Network ID    = SPIFNET
LU name       = NYX1GW0A
Remote Network ID = SPIFNET
Remote Control Point = NYX1GW
Uninterpreted Name = NYX1GW
```

Defining Mode:

```
Name           = IBMRDB
Description    = IBMRDB
Class of service = #CONNECT
```

Defining Symbolic Destination Info:

```
Name           = DB2CPIC
Description    = To SERV from NYX1
Partner LU    = SPIFNET.NYX1GW0A
Mode          = IBMRDB
Local LU      = CLI1GW0A
Partner TP    = DB2DRDA
```

After you have configured these SNA parameters, you must stop and start the SNA server. To do this, perform the following steps:

- Step 1. Log on to the system as a user with root authority.
- Step 2. Make sure your PATH contains the \$express/bin (/usr/lpp/express/bin) entry.
- Step 3. Check for active users before stopping by entering the following command:

```
express_admin shutdown
```
- Step 4. Stop all EXPRESS activity by entering the following command:

```
express_admin stop
```

Step 5. Start EXPRESS by entering the following command:

```
express_admin start
```



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring SNAPplus2 for HP-UX

This section describes how to configure SNAPplus2 for HP-UX on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC.

Before you begin, ensure that your workstation has HP-UX SNAPplus2 installed. If you need more information in order to configure your SNA environment, refer to the online help provided with SNAPplus2.

The following assumptions are made:

- The basic installation of the SNAPplus2 for HP-UX package has already been completed.
- The DB2 client has been installed.
- The user is logged on as root.

To configure SNAPplus2 for DB2 Connect, log on to the system as a user with root authority and use either the `/opt/sna/bin/snapadmin` program or the `/opt/sna/bin/X11/xsnapadmin` program. Information about these programs can be found in the system documentation. The following steps describe how to use the `xsnapadmin` program to configure SNAPplus2.

Step 1. Enter the command `xsnapadmin`. The Servers window opens. Double-click on your node.

Step 2. Define a Node

- a. Select **Services** —> **Configure Node Parameters** from the menu bar. The Node Parameters window opens.
- b. Click on the **APPN support** drop down box and select the **End node** option.
- c. Enter your Network ID and the Local PU Name (**9** and **10**) in the **Control point name** fields.

- d. Enter Local PU Name (**10**) in the **Control point alias** field.
- e. Enter your Node ID (**13** and **14**) in the **Node ID** fields.
- f. Click **OK**.

Step 3. Define a port

- a. Select the **Connectivity and Dependent LUs** window.
- b. Click **Add**. The Add to Node window opens.
- c. Select the **Port using** radio button.
- d. Click on the **Port using** drop down box and select the appropriate port type. For our example, we will select the **Token ring card** option.
- e. Click **OK**. The Port window for the chosen port type opens.
- f. Enter a name for the port in the **SNA port name** field.
- g. Select the **Initially active** checkbox.
- h. From the **Connection network** box, select the **Define on a connection network** checkbox.
- i. Enter your Network ID (**9**) in the first part of the **CN name** field.
- j. Enter your local Control Point name (**10**) in the second part of the **CN name** field.
- k. Click **OK**. The **Port** window closes and a new port appears in the **Connectivity and Dependent LUs** window.

Step 4. Define a link station

- a. In the **Connectivity and Dependent LUs** window, select the port that you defined in the previous step.
- b. Click **Add**. The Add to Node window opens.
- c. Select the **Add a link station to port** radio button.
- d. Click **OK**. The Token ring link station window opens.
- e. Enter a name for the link in the **Name** field.
- f. Click on the **Activation** drop down box and select the **On demand** option.
- g. Select the **Independent only** option in the **LU traffic** box.
- h. In the **Independent LU traffic** box:
 - 1) Enter the Network ID (**3**) and the Partner LU Name (**2**) in the **Remote Node** fields.
 - 2) Click on the **Remote node type** drop down box and select the type of node that applies to your network.
- i. In the **Contact information** box, enter the SNA Destination Address (**8**) assigned to the DB2 server in the **Mac address** field.

- j. Click **OK**. The Link Station window is closed and a new link station appears as a child of the port in the **Connectivity and Dependent LUs** window.

Step 5. Define a local LU

- a. Select the **Independent local LUs** window.
- b. Click **Add**. The Local LU window opens.
- c. Enter your independent local LU Name (**11**) in the **LU name** field.
- d. Enter the same name in the **LU alias** field (**12**).
- e. Click **OK**. The new LU appears in the **Independent local LUs** window.

Step 6. Define a remote node

- a. Select the **Remote Systems** window.
- b. Click **Add**. The Add to Node window opens.
- c. Select **Define remote node**.
- d. Click **OK**. The Remote Node configuration window appears.
- e. Enter the Network ID (**3**) and the Partner LU Name (**2**) in the **Node's SNA network name** field.
- f. Click **OK**. The remote node appears in the **Remote Systems** window, and a default partner LU is defined for the node and also appears as a child of the remote node.

Step 7. Define a partner LU

- a. In the **Remote Systems** window, double-click the default partner LU that was created when you defined a remote node in the previous step. The Partner LU window opens.
- b. Enter the same Partner LU name (**2**) in the **Alias** and **Uninterpreted name** fields.
- c. Select **Supports parallel sessions**.
- d. Click **OK**.

Step 8. Define a mode

- a. Select **Services** —> **APPC** —> **Modes** from the menu bar. The Modes window opens.
- b. Click **New**. The Mode window opens.
- c. Enter a mode name (**15**) in the **Name** field.
- d. The configuration values below are suggested for the following fields:
 - 1) **Initial Session limits**: 20
 - 2) **Maximum Session limits**: 32767
 - 3) **Min con. winner sessions**: 10

- 4) **Min con. loser sessions:** 10
- 5) **Auto-activated session:** 4
- 6) **Receive pacing window:** 8

These values are suggested because they are known to work. You will need to tailor these values so that they are optimized for your particular application environment.

- e. Click **OK**. The new mode appears in the Modes window.
- f. Click **Done**.

Step 9. Define the CPI-C destination name

- a. Select **Services** → **APPC** → **CPI-C** from the menu bar. The CPI-C destination names window opens.
- b. Click **New**. The CPI-C destination window opens.
- c. Enter the Symbolic Destination Name (**16**) you want to associate with the DB2 server database in the **Name** field.
- d. In the **Partner TP** box:
 - 1) Select **Service TP (hex)**, and enter the hexadecimal TP number (**17**), or
 - 2) Select **Application TP**, and enter the application TP name. (**17**).
- e. In the **Partner LU and mode** box:
 - 1) Select the **Use PLU Alias** radio button, and enter the Partner LU Alias (**2**) that you created in a previous step.
 - 2) Enter the Mode name (**15**) for the mode that you created in a previous step in the **Mode** field.
- f. In the **Security** box, select the radio button that corresponds to the type of security level that you want to run on your network.
- g. Click **OK**. The new destination name appears in the Destination names window.
- h. Click **Done**.

Step 10. Test the APPC connection

- a. Start the SNA subsystem by entering the **/opt/sna/bin/sna start** command. You can enter the **/opt/sna/bin/sna stop** command to stop the SNA subsystem first, if required.
- b. Start the SNA administration program. You can enter either the **/opt/sna/bin/snaadmin** command, the **/opt/sna/bin/X11/xsnaadmin** command.
- c. Start the subsystem node. Select the appropriate node icon in the button bar, and click the **Start** button.

- d. Start the link station. Select the link station you defined previously in the **Connectivity and Dependant LUs** window, and click **Start**.
- e. Start the session. Select the LU you defined previously in the **Independant Local LUs** window, and click **Start**. A session activation window opens. Select or enter the Partner LU and Mode desired.
- f. Click **OK**.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, using the Client Configuration Assistant (CCA) is recommended. For more information, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 109 and the sections that follow.

Configuring SunLink SNA for Solaris

This section describes how to configure SunLink SNA PU 2.1 (SunLink SNA) for Solaris on your DB2 client workstation to connect to a DB2 Connect or DB2 Universal Database server using APPC.

Before you begin, ensure that your workstation has SunLink SNA installed. If you need further information in order to configure your SNA environment, refer to the *SunLink PU 2.1 Server Configuration and Administrator's Manual*.

The following assumptions are made:

- The basic installation of the SunLink SNA PU 2.1 for Solaris package has already been completed.
- The DB2 client has been installed.
- The user is logged on as root.

To configure the SunLink SNA Server for use by a DB2 Client, log on as root and perform the following steps:

Step 1. “Create the CPIC Side File” on page 107

Step 2. “Create the SNA Server Configuration File” on page 107

Step 3. “Define the Environment Variables Required by SunLink SNA” on page 108

Step 4. “Start the SunLink SNA Subsystem” on page 108

Create the CPIC Side File: You can use any plain text editor to create the CPIC side file. The file must be placed in the application's path for a standalone DB2 Connect for Solaris system.

Note: The name of the CPIC side file must be the same as the symbolic destination name specified in the DB2 node directory on the DB2 client.

The following example shows the CPIC side file sections required for configuring the SunLink SNA to connect to a DB2 server:

```
# CPIC Side File information
#
PTNR_LU_NAME=NYX1GW0A
MODE_NAME=IBMRDB
TP_NAME=DB2DRDA
SECURITY=NONE
```

Create the SNA Server Configuration File: You can use any plain text editor to create the SNA server configuration file. This file is called sunpu2.config, and it must be placed in /opt/SUNWpu21, or the directory where SunLink SNA PU 2.1 Server is installed.

The following example shows the configuration file sections required for configuring the SunLink SNA to connect to a DB2 server:

```
// SunLink SunLU6.2/SunPU2.1 SNA Server Sample Configuration
// Token Ring Peer-to-Peer System A @(#)sunlu62.a.tr
//
// The physical connection is a Token Ring interface adapter.

CP      NAME=CLI1GW           // Local name (8 char max)
        NQ_CP_NAME=SPIFNET.CLI1GW // Network Qualified Name
        ;

TRLINE  NAME=MAC1           // SunLink specific name
        SOURCE_ADDRESS=x'400011527509' // sysA_mac_addr for Sun machine
        ;

DLC     NAME=SERVLINK       // User defined name (8 char max)
        LINK_NAME=MAC1     // Line name this station is on
        LCLLSAP=x'04'      // Local Link Service Access Point
        RMTLSAP=x'04'      // Remove Link Service Access Point
        RMTMACADDR=x'400009451901 // sysB_mac_addr
        TERMID=x'07127509'  // XID negotiation
        ;

LU      NAME=CLI1GW0A       // Local name (8 char max)
        NQ_LU_NAME=SPIFNET.CLI1GW0A // Network Qualified Name
        SESS_LMT=50        // Max LU sessions
        LUTYPE=6.2
        ;

PTNR_LU NAME=NYX1GW0A      // Partner LU name(8 char max)
```

```

        LOC_LU_NAME=CLI1GWOA           // Associated Local LU
        NQ_LU_NAME=SPIFNET.NYX1GW0A   // Network Qualified Name
        ;

MODE   NAME=IBMRDB                     // Mode Name (8 char max)
      DLC_NAME=SERVLINK                 // Associated DLC
      PTNR_LU_NAME=NYX1GWOA           // Associated Partner LU
      LCL_MAX_SESS_LMT=30              // Max Session Limit
      MIN_CW_SESS=15                   // Min Conwinners
      MIN_CL_SESS=15                   // Min Conlosers
      ;

```

Define the Environment Variables Required by SunLink SNA: In order to run any application, you must set the following environment variables:

APPC_GATEWAY

Name of the DB2 for Solaris server (usually the TCP/IP hostname).

APPC_LOCAL_LU

Name of the local LU name provided in the SNA configuration file.

Export these on the DB2 client machine before proceeding with the next step.

Start the SunLink SNA Subsystem: To start the SunLink SNA subsystem, perform the following steps:

Step 1. Change to the SunLink installation directory, usually:

```
cd /opt/SUNWpu21
```

Step 2. Set up environment variables for *FlexLM* licensing. For example:

```
export LD_LIBRARY_PATH=/usr/openwin/lib:/usr/lib
export LM_LICENSE_FILE=/etc/opt/licenses/licenses_combined
```

Refer to the SunLink documentation for more information.

Step 3. Ensure you have created the CPIC side file as described in “Create the CPIC Side File” on page 107

Step 4. Ensure you have created the SNA server configuration file as described in “Create the SNA Server Configuration File” on page 107.

Step 5. Use the sunop utility to check the status of SunLink SNA if it is already started.

Check to see if the PU and/or DLC status is *connected*. You can also use sunop to check the status of links. Refer to the SunLink documentation for details of the sunop utility.

Step 6. Stop SunLink if it is active. For example, enter:

```
kill -9 sunpu2.pid
kill -9 sunlu2.pid
```

Step 7. Start SunLink using the following command:

```
sunpu2.1
```



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Step 3. Catalog the APPC or APPN Node

You must add an entry to the DB2 client workstations’s node directory to describe the remote node. In most cases, you will add an APPC node entry to the node directory. For OS/2 and Windows 32-bit operating systems, you can alternatively add an APPN node entry if your local SNA node has been set up as an APPN node.

To catalog the node, perform the following steps:

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

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

```
. INSTHOME/sql1lib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 3. To catalog an APPC node, specify the chosen alias (*node_name*), Symbolic destination name (*sym_dest_name*), and the APPC security type (*security_type*) that the client will use for the APPC connection. Enter the following commands in the command line processor:

```
catalog "appc node node_name remote sym_dest_name \
security security_type";
terminate
```



The *sym_dest_name* parameter is case-sensitive and must match *exactly* the case of the Symbolic destination name you defined previously.

For example, to catalog a remote database server with the Symbolic destination name *DB2CPIC* on the node called *db2node*, using APPC Security type *NONE*, enter the following commands:

```
catalog appc node db2node remote DB2CPIC security NONE
terminate
```

Step 4. To catalog an APPN node, specify the chosen alias (*node_name*), the network ID (**1**), the remote partner LU (**4**), the transaction program name (**17**), the mode (**15**), and the security type. Enter the following commands, substituting your values from the worksheet in Table 30 on page 268:

```
catalog "appn node db2node network SPIFNET remote NYX1GW0A
tpname DB2DRDA mode IBMRDB security NONE"
terminate
```



If you need to change values that were set with the **catalog node** command, perform the following steps:

Step 1. Run the **uncatalog node** command in the command line processor as follows:

```
db2 uncatalog node node_name
```

Step 2. Recatalog the node with the values that you want to use.

Step 4. Catalog the Database

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

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see “Appendix F. Naming Rules” on page 509.



If you are adding a database to a system that has a DB2 server or DB2 Connect server product installed, log on to this system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority on the instance. For more information, see “Working with the System Administrative Group” on page 451.

This restriction is controlled by the *catalog_noauth* database manager configuration parameter. For more information, refer to the *Administration Guide*.

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

Table 18. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (<i>database_name</i>)	The database alias (<i>database_alias</i>) of the <i>remote</i> database. When you create a database, it is automatically cataloged on the server with the database alias (<i>database_alias</i>) the same as the database name (<i>database_name</i>), unless specified otherwise.	sample	
Database alias (<i>database_alias</i>)	An arbitrary local nickname for the remote database, on the client. If you do not provide one, the default is the same as the database name (<i>database_name</i>). The database alias the name that you use when connecting to a database from a client.	tor1	
Authentication (<i>auth_value</i>)	The value of the authentication required by your enterprise. Please refer to the <i>DB2 Connect User's Guide</i> for more information on this parameter.	DCS This means that the userid and password supplied are validated at the host or AS/400 only.	
Node name (<i>node_name</i>)	The name of the node directory entry that describes where the database resides. Use the same value for node name (<i>node_name</i>) that you used to catalog the node in the previous step.	db2node	

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

```
. INSTHOME/sql1lib/db2profile    (for bash, Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 4. Catalog the database by entering the following commands:

```
db2 catalog database database_name as database_alias
at node node_name
db2 terminate
```

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

```
db2 catalog database sample as tor1 at node db2node
db2 terminate
```



If you need to change values that were set with the **catalog database** command, perform the following steps:

Step a. Run the **uncatalog database** command as follows:

```
db2 uncatalog database database_alias
```

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

Step 5. Test the Client-to-Server Connection

After configuring the client for communications, you will need to connect to a remote database to test the connection.

Step 1. Start the database manager by entering the **db2start** command on the server (if it was not automatically started at boot time).

Step 2. If you are using a UNIX client, run the start-up script as follows:

```
. INSTHOME/sql1lib/db2profile    (for Bash, Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 3. Enter the following command on the client to connect the client to the remote database:

```
db2 connect to database_alias user userid using password
```

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

If the connection is successful, you will get a message showing the name of the database to which you have connected. You are now able to retrieve data

from that database. For example, to retrieve a list of all the table names listed in the system catalog table, enter the following SQL command in the Command Center or CLP:

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

When you are finished using the database connection, enter the **command reset** command to end the database connection. The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server for a DB2 server and on the host or AS/400 machine for a DB2 Connect server.

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

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

When you are finished using the database connection, enter the **db2 connect reset** command to end the database connection.



You are now ready to start using DB2. For more advanced topics, refer to the *Administration Guide*.

Troubleshooting the Client-to-Server Connection

If the connection fails, check the following items:

At the *server*:

1. The *db2comm* registry value includes the value *appc*.



Check the settings for the *db2comm* registry value by entering the **db2set DB2COMM** command. For more information, refer to the *Administration Guide*.

2. The transaction program name (*tpname*) parameter was updated correctly in the database manager configuration file (or the admin server configuration file, if you are setting up the Administration Server).
3. The security service was started. Enter the **net start db2ntsecserver** command (for Windows NT and Windows 2000 servers only).
4. The database was created and cataloged properly.
5. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).



If there are problems starting a protocol's connection managers, a warning message appears and the error messages are logged in the `db2diag.log` file located in the `INSTHOME/sql1lib/db2dump` directory for UNIX platforms or in the `x:\sql1lib\db2dump` directory for non-UNIX platforms.

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

At the *client*:

1. The node was cataloged with the correct Symbolic Destination Name (*sym_dest_name*).
2. The node name (*node_name*) that was specified in the database directory points to the correct entry in the node directory.
3. The database was cataloged properly, using the *server's* database alias (the *database_alias* that was cataloged when the database was created on the server), as the database name (*database_name*) on the client.

If the connection still fails after you verify these items, refer to the *Troubleshooting Guide*.

Chapter 8. Control Center Installation and Configuration

This chapter describes how to install and configure the DB2 Control Center.

The Control Center is the main DB2 graphical tool for administering your database. It is available on Windows 32-bit, OS/2, and UNIX operating systems.

The Control Center provides a clear overview of all the systems and database objects being managed. You can also access other administration tools from the Control Center by selecting icons on the Control Center toolbar or from the Tools pop-up menu.

Application versus Applet

You can run the Control Center either as a Java application or as a Java applet through a web server. In both cases you need a supported Java Virtual Machine (JVM) installed on your machine to run the Control Center. A JVM can be a Java Runtime Environment (JRE) for running applications, or a Java-enabled browser for running applets.

- Java *applications* run just like other applications on your machine, provided you have the correct JRE installed.

On Windows 32-bit operating systems, the correct JRE level was installed or upgraded for you during DB2 installation.

On AIX systems, the correct JRE was installed for you during DB2 installation only if another JRE was not detected on your system. If another JRE was detected on your AIX system during DB2 installation, the JRE that comes with DB2 was not installed. In this case, you must install the correct JRE level before running the Control Center.

On all other operating systems you must install the correct JRE level before running the Control Center. See Table 20 on page 117 for a list of correct JRE levels.

Note: Some operating systems, including OS/2 Warp Server for e-business and AIX 4.3, have built-in Java support. For more information, check with your administrator.

- Java *applets* are programs that run within Java-enabled browsers. The Control Center applet code can reside on a remote machine and is served to the client's browser through a web server. This type of client is often called a *thin client* because a minimal amount of resources (a Java-enabled browser) is required to run the Java applet.

You must use a supported Java-enabled browser to run the Control Center as a Java applet. See Table 20 on page 117 for a list of supported browsers.

Machine Configurations

You can set up your Control Center in a number of different ways. The following table identifies four scenarios, each showing a different way of installing the required components. These scenarios are referenced throughout the Control Center Services Setup (Applet Mode only) section that follows the table.

Table 19. Control Center Machine Configuration Scenarios

Scenario	Machine A	Machine B	Machine C
1 - Stand-alone, Application	JRE Control Center application DB2 server		
2 - Two Tier, Application	JRE Control Center application DB2 client		DB2 server
3 - Two Tier, Browser	Supported Browser (Windows and OS/2 only) Control Center applet	Web server JDBC Applet Server DB2 server	
4 - Three Tier, Browser	Supported Browser (Windows and OS/2 only) Control Center applet	JDBC Applet Server DB2 client	DB2 server

Figure 1 on page 117 summarizes the four basic Control Center machine configurations:

Table 20. Supported Java Virtual Machines (JVMs) for the Control Center (continued)

Operating System	Correct Java Runtime Environments	Supported Browsers
Solaris	JRE 1.1.8	None
HP-UX 11	JRE 1.1.8	None
IRIX	JRE 1.1.8 (3.1.1 SGI) + Cosmo code 2.3.1	None
PTX	JRE 1.1.8	None

For the latest information on supported JREs and browsers, go to <http://www.ibm.com/software/data/db2/udb/db2cc>

Setting Up and Working with the Control Center

This section describes how to set up and customize the Control Center for your environment.

Control Center Services Setup (Applet Mode only)

If you will run the Control Center as an application, skip this section and go to “Running the Control Center as a Java Application” on page 120.

To set up the Control Center to run as an applet:

1. Start the Control Center JDBC Applet Server.
2. On Windows NT or Windows 2000, start the security server.

1. Start the Control Center JDBC Applet Server

To start the Control Center JDBC Applet Server, enter the `db2jstrt 6790` command, where 6790 represents any 4-digit port number that is not already in use.

We recommend that you start the Control Center JDBC Applet Server with a user account that has SYSADM authority.

The first time you start the Control Center JDBC Applet Server, it will create several node directory entries, together with various files for administration purposes. In Scenarios 1 and 3 in “Machine Configurations” on page 116, all of these administration files and directory entries will be created in the current DB2 instance.

Most DB2 resources are accessed over **database connect** or **instance attach**. In both cases the user must supply valid user ID and password combinations to gain access. However, some resources are accessed directly by the Control Center JDBC Applet Server, including the database and node directories (catalogs), and the Command Line Processor. Access to these resources is

performed by the Control Center JDBC Applet Server on behalf of the logged in Control Center user. Both the user and the server are required to have the proper authorization before access will be granted. For example, to update the database directory, a minimum of SYSCTRL authority is required.

It is possible to run an instance of the Control Center JDBC Applet Server with any level of security, but you will be unable to update certain resources such as database and node directories. In particular, you may encounter a **SQL1092N** message informing you of the lack of authorization for a request. The user specified in the message can be either the user signed into Control Center, or the user account running the Control Center JDBC Applet Server.

On Windows NT, you can start the Control Center JDBC Applet Server by clicking **Start** and selecting **Settings** → **Control Panel** → **Services**. Select the **DB2 JDBC Applet Server - Control Center** service and click **Start**.

On Windows 2000, you can start the Control Center JDBC Applet Server by clicking **Start** and selecting **Settings** → **Control Panel** → **Administrative Tools** → **Services**. Select the **DB2 JDBC Applet Server - Control Center** service, click the **Action** menu and then select **Start**.

On any system, you can start the Control Center JDBC Applet Server by entering:

```
net start DB2ControlCenterServer
```

This step is not required if your Control Center JDBC Applet Server is autostarted.

If you start the Control Center JDBC Applet Server as a Windows NT or Windows 2000 service, you must configure the startup in the services dialog to change the account information.

2. Start the Windows NT or Windows 2000 Security Server

To work with the Control Center on Windows NT or Windows 2000, the security server must be running. During DB2 installs, the Security Server is usually set up to autostart.

You can check if the Security Server is running on Windows NT by clicking **Start** and selecting **Settings** → **Control Panel** → **Services**.

On Windows 2000, click **Start** and select **Settings** → **Control Panel** → **Administrative Tools** → **Services**.

If the **DB2 Security Server** is not started on Windows NT, select it and click **Start**. On Windows 2000, select the **Action** menu and click **Start**.

After you have started the Control Center JDBC Applet Server and started the Windows NT or Windows 2000 security server (if necessary), go to “Running the Control Center as a Java Applet”.

Working with the Control Center

You can run the Control Center as a Java application or as a Java applet. If your environment is configured similar to Scenarios 1 or 2 in Table 19 on page 116, you must run the Control Center as an application. If your environment is configured like Scenarios 3 or 4, you must run it as an applet.

Running the Control Center as a Java Application

To run the Control Center as a Java application, you must have the correct Java Runtime Environment (JRE) installed. See Table 20 on page 117 for the correct JRE level for your operating system.

1. To start the Control Center as an application:

On Windows 32-bit operating systems:

Click **Start** and select **Programs** → **IBM DB2** → **Control Center**.

On OS/2:

Open the **IBM DB2** folder and double-click on the **Control Center** icon.

On all supported platforms:

Start the Control Center from a command prompt by entering the **db2cc** command.

2. The DB2 Control Center window opens.
3. You can start working with the Control Center without an existing database by creating a sample database. Enter the **db2sampl** command on the DB2 Universal Database server. On UNIX operating systems, ensure that you are logged in to the DB2 instance before you enter the **db2sampl** command.

Running the Control Center as a Java Applet

To run the Control Center as a Java applet, you must have a Web server set up on the machine that contains the Control Center applet code and the Control Center JDBC Applet Server. The Web server must allow access to the `sqllib` directory.

If you choose to use a virtual directory, substitute this directory for the home directory. For example, if you map `sqllib` to a virtual directory called `temp` on a server named `yourserver`, a client would use the URL:

`http://yourserver/temp`

If you do not have the DB2 documentation installed and you would like to configure your web server to work with DB2's online documentation, see “Appendix C. Setting up DB2 Documentation on a Web Server” on page 479.

To run Control Center as an applet on Windows 32-bit or OS/2 operating systems, you must run **db2classes.exe** on the machine where the DB2 JDBC Applet Server resides to unzip the required Java class files. On UNIX-based systems, you must uncompress and untar **db2classes.tar.Z** to make the required Java class files available.

To load the Control Center HTML page, perform the following steps:

1. Start the **Control Center Launch** page through your web server. In your browser, select **File -> Open Page**. The **Open Page** dialog box appears. Enter the URL of your Web server and the main Control Center page and click on the **Open** push button. For example, if your server is named **yourserver**, you would open `http://yourserver/cc/prime/db2cc.htm`
2. In the **Server port** field, enter a value for the Control Center JDBC Applet Server port. The default server port value is 6790.
3. Click on the **Start Control Center** push button.
4. The Control Center Sign On window opens. Enter your user ID and password. This user ID must have an account on the machine that is running the Control Center JDBC Applet Server. Your initial logon will be used for all database connections. It can be changed from the Control Center pull-down menu. A unique user profile will be assigned to each user ID. Click **OK**.
5. The DB2 Control Center window opens.
6. You can start working with the Control Center without an existing database by creating a sample database. Enter the **db2sampl** command on the DB2 Universal Database server. On UNIX operating systems, ensure that you are logged in to the DB2 instance before you enter the **db2sampl** command.

Customizing Your Control Center HTML File

To automatically start the Control Center the next time you open `db2cc.htm`, perform the following steps:

- For Scenarios 1 or 2, modify the `autoStartCC` parameter tag in `db2cc.htm` from

```
param name="autoStartCC" value="false"
```

to

```
param name="autoStartCC" value="true"
```

- For Scenarios 3 or 4, modify the `autoStartCC`, `hostNameText`, and `portNumberText` parameter tags in `db2cc.htm` to

```
param name="autoStartCC" value="true"  
param name="hostNameText" value="yourserver"  
param name="portNumberText" value="6790"
```

where `yourserver` represents the Server name or IP address and `6790` represents the server port value of the machine to which you want to connect.

Configuring Your Web Server to Work with the Control Center

For general web server configuration information, consult the setup documentation that came with your web server.

For more information on serving DB2 online documentation through a web server, see "Appendix C. Setting up DB2 Documentation on a Web Server" on page 479.

Functional Considerations

If you are using the Control Center over the Internet, be aware that there is no encryption of the data flow between the Control Center JDBC Applet Server and the browser.

To use the color options of Visual Explain on Netscape, you must set your operating system to support more than 256 colors.

On OS/2 systems, you must install the Control Center on an HPFS-formatted drive. DB2 does not support the installation of the Control Center on an OS/2 FAT drive, because an OS/2 FAT drive does not support long filenames required by Java.

Every activity will be associated with an explicit DB2 connection or attachment. For security purposes, every DB2 activity will be validated.

When you are using the Control Center under Scenarios 3 or 4, the local system is Machine B. The local system is the system name as it appears in the DB2 Control Center window.

Installation Tips for Control Center Help on UNIX Operating Systems

When installing the Control Center online help on UNIX operating systems you should keep the following in mind:

- You should install the Control Center help and the product documentation at the same time. If you install the Control Center help and the DB2 online product documentation separately, you can expect the second installation to take some time. This is true regardless of which package is installed first.
- You must select the Control Center help for any non-English language explicitly. Installing the product messages for a particular language does not mean that the Control Center help for that language is automatically

installed. However, if you install the Control Center help for a particular language, the product messages for that language are installed automatically.

- If you manually install the Control Center on UNIX-based workstations rather than using the `db2setup` utility, you must run the `db2insthtml` command to install the online documentation. For more information, refer to the *DB2 for UNIX Quick Beginnings*.

Configuring TCP/IP on OS/2

To run the Control Center on OS/2 Warp 4 while disconnected from a LAN, you will need to configure TCP/IP to enable local loopback and localhost. If you are running OS/2 Warp Server for e-business, local loopback is enabled by default.

Enabling Local Loopback

To enable local loopback on your system:

1. Open the **System Setup** folder.
2. Open the **TCP/IP Configuration** notebook.
3. View the **Network** page.
4. In the **Interface to Configure** list box, highlight **loopback interface**.
5. If the **Enable interface** check box is not selected, select it now.
6. Verify that the **IP address** is `127.0.0.1` and **Subnet Mask** is empty.

Enabling Localhost

To enable localhost on your system:

1. To check if localhost is enabled, enter the **ping localhost** command.
 - If data is returned and localhost is enabled, you can skip steps 2 and 3 below and go directly to step 4.
 - If `localhost unknown` is returned, or if the command hangs, localhost is not enabled. Go to step 2.
2. If you are on a network, make sure that loopback is enabled. To enable local loopback see “Enabling Local Loopback”.
3. If you are *not* on a network, enable localhost by performing these steps:
 - a. Add the following line after other `ifconfig` lines in the `MPTN\BIN\SETUP.CMD` command file:

```
ifconfig lo 127.0.0.1
```
 - b. In the TCP/IP configuration folder, perform the following steps:
 - 1) Go to the **Configure Name Resolution Services** page.
 - 2) In the **Hostname configuration without a Nameserver** table, add an entry with *IP Address* set to `127.0.0.1` and *Hostname* set to `localhost`.

Note: If you have a hostname for your machine on the **Configure LAN Name Resolution Services** page, you must add this name as an alias when you set the *IP Address 127.0.0.1* to localhost.

- c. Select the **Look through HOSTS list before going to the nameserver** list box. This step tells your OS/2 system that when it is looking for a host, such as localhost, it should use the host address found on your machine rather than checking the nameserver. If the host is not defined on your machine, OS/2 continues looking for the host by using the nameserver you configured.
 - d. Close **TCP/IP Configuration** and reboot the system.
 - e. You should be able to ping localhost without being connected to any network.
4. Verify that your hostname is correct. On an OS/2 command line, enter the **hostname** command. The hostname returned should match the one listed in the **TCP/IP Configuration** notebook on the **Hostnames** page and it must be less than 32 characters. If the hostname deviates from these conditions, correct it on the **Hostnames** page.
 5. Verify that your hostname is set properly in CONFIG.SYS. You should see a line similar to the following:

```
SET HOSTNAME=<correct_name>
```

where *<correct_name>* represents the value returned by the **hostname** command. If this is not the case, make the necessary changes and reboot your system when you are finished.

Verifying TCP/IP Configuration on OS/2

If you are having trouble running the Control Center on OS/2 while disconnected from a LAN, try running the **sniffle /P** command to diagnose the problem.

Troubleshooting Information

For the latest service information on the Control Center, point your browser to <http://www.ibm.com/software/data/db2/udb/db2cc>

If you are having problems running the Control Center, check the following:

- Ensure the Control Center JDBC Applet Server (db2jd) is running.
- Verify that the server port number is correct.
- Check that the Control Center JDBC Applet Server is running under a user account that has SYSADM authority.
- Ensure that the Database Administration Server (DAS) is running on any DB2 Universal Database systems you are trying to administer by entering the **db2admin start** command. On UNIX-based systems, ensure that you are logged in as the DAS instance owner when you issue this command.

If you are having problems running the Control Center as an *application*, also check the following:

- Verify that the correct JRE is installed. See Table 20 on page 117 for more information.

If you are having problems running the Control Center as an *applet*, also check the following:

- Verify that you are running a supported browser. See Table 20 on page 117 for more information.
- Check your browser's Java console window for diagnostic and trace information for the Control Center.
- Make sure that the client browser does not have CLASSPATH set. To make sure CLASSPATH is not set, open a command window and enter **SET CLASSPATH=** then start your client browser from this command window. Also, note that if CLASSPATH is not set in a Windows NT or Windows 2000 environment, it may still get picked up from autoexec.bat from a Windows 9x installation on the same machine.
- Ensure that you are using the db2cc.htm file from the machine running the Control Center JDBC Applet Server.
- Remember that the Control Center works within the DB2 client's locale, and that the DB2 client is located at the Control Center JDBC Applet Server's location.

Administering DB2 for OS/390 and DB2 Connect Enterprise Edition servers with the Control Center

The Control Center has been greatly enhanced to deliver new management functions to database administrators who need to manage DB2 for OS/390 V5.1 and later database servers.

The Control Center has also been enhanced to manage operational and performance characteristics of DB2 Connect Enterprise Edition connectivity servers. The combination of DB2 for OS/390 server management and new DB2 Connect monitoring support provide complete end-to-end administration and monitoring for desktop and web applications that work with DB2 for OS/390 servers.

The DB2 Control Center uses the familiar "explorer" interface to allow database administrators to easily navigate between different database servers and the database objects they manage. Context sensitive right-mouse activated menus provide administrators with the ability to change attributes of database objects and to launch commands and utilities.

Database objects are presented in a consistent fashion for all DB2 family servers. This greatly reduces the amount of learning that is required for administrators who need to manage both DB2 for OS/390 and DB2 Universal Databases on Windows NT, Windows 2000, UNIX and OS/2 servers. While the Control Center preserves consistency across servers it does not hide capabilities that are unique to each DB2 server. This gives database administrators the power to perform all aspects of their tasks.

The ability to manage DB2 Connect connectivity servers is delivered through management of user connections and by keeping vital statistics on various performance aspects of the connectivity server. For example, database administrators can easily view all of the users connected through a particular DB2 Connect server, and their connection characteristics.

Administrators can also collect load and performance information such as the number of SQL statements and transactions executed, number of bytes sent and received, statement and transaction execution times and much more. Collected data can be displayed using easy to understand live graphs.

Preparing DB2 for OS/390 Servers for the Control Center

The DB2 Control Center uses stored procedures to deliver many of its management functions. Therefore, for the Control Center to function properly each DB2 for OS/390 server that will be managed from the Control Center needs to have stored procedures enabled and the proper stored procedures installed on that server.

For more information on applying service and required Function Modification Identifiers, refer to the *DB2 for OS/390 Program Directory*.

Working with the Control Center

Before you can work with a server and its databases you will need to catalog information about the server on the Control Center workstation. The DB2 Control Center only works with servers and databases that are cataloged on the workstation where the Control Center is running. The easiest way to accomplish this on Windows and OS/2 workstations is to use the DB2 Client Configuration Assistant (CCA).

Once the Control Center is running, start by clicking on the plus sign next to the server you want to administer. Select the database or connectivity server objects that you want to administer and right click on the object to work with object properties or to execute actions on the object. You can invoke the online help by clicking **Help**, or by pressing **F1** at any time.

Other Sources of Information

For more information about using the Control Center to administer DB2 for OS/390, refer to the following online resource:

<http://www.ibm.com/software/data/db2/os390/v6facts/db2cc.html>

For complete information about DB2 for OS/390 Version 6, refer to the online library: <http://www.ibm.com/software/data/db2/os390/v6books.html>

For more information about stored procedures and the Control Center for OS/390, go to: <http://www.ibm.com/software/data/db2/os390/cc390/>

Chapter 9. Configuring Stored Procedure Builder

This section describes how to configure IBM DB2 Stored Procedure Builder as an add-in with other Windows 32-bit development environments. It also describes specific JDK configuration instructions for running Stored Procedure Builder on Solaris systems.

Stored Procedure Builder is part of the DB2 Application Development Client.

Configuring Stored Procedure Builder to run as an add-in with Microsoft Visual Basic

These instructions apply to Stored Procedure Builder running on Windows 32-bit operating systems.

If Microsoft Visual Basic was not installed when you installed DB2, you must perform the following steps to register the add-in with Visual Basic:

1. Switch to the `x:\sql11ib\bin\` directory, where `x`: represents the drive on which you installed DB2, and run the **`db2spbvb -addtoini`** command.
2. Start Visual Basic.
3. Select **Add-Ins** → **Add-In Manager**. The Add-In Manager window opens.
4. Select **IBM DB2 Stored Procedure Builder**.
5. Click **OK**.

Stored Procedure Builder is added to the Add-Ins menu.

Configuring Stored Procedure Builder to run as an add-in with Microsoft Visual C++

These instructions apply to Stored Procedure Builder running on Windows 32-bit operating systems.

1. If Microsoft Visual Studio was not installed when you installed DB2, you must perform one of following steps to register the add-in with Visual Studio:
 - For Visual Studio 5, copy `DB2SSPB.DLL` from `x:\sql11ib\bin` to `y:\Program Files\DevStudio\SharedIDE\AddIn`, where `x`: represents the drive on which you installed DB2 and `y`: represents the drive on which you installed Visual Studio 5.

- For Visual Studio 6, copy DB2SPBVS.DLL from x:\sql11b\bin to y:\Program Files\Microsoft Visual Studio\Common\MSDev98\AddIns, where x: represents the drive on which you installed DB2 and y: represents the drive on which you installed Visual Studio 6.
2. Start Microsoft Visual C++.
 3. Select **Tools** → **Customize**. The Customize window opens.
 4. In the Customize window, make sure that **IBM DB2 Stored Procedure Builder** is selected.
 5. Click **Close**.

The Stored Procedure Builder icon is added to the icon toolbar.

Configuring Stored Procedure Builder on AIX and Solaris

In order to run Stored Procedure Builder on AIX or Solaris, you need to ensure that you have a Java Development Kit (JDK) installed on your system. DB2 installs a Java Run-Time Environment (JRE) on your system which can be used for the Control Center, however, it will not allow you to use Stored Procedure Builder.

Once you have installed your JDK, you need to ensure that DB2 is aware of location of the JDK by setting its path in the instance's database manager configuration file. To set the JDK's path, perform the following steps:

1. Log on to the system as a user with SYSADM authority. For more information, see "Working with the System Administrative Group" on page 451.
2. Update the database manager configuration file by entering the following command:

```
db2 update dbm cfg using jdk11_path /usr/java
```

where */usr/java* represents the JDK installation path.

3. Once you update an instance's database manager configuration file, you must stop and restart the instance for the changes to take affect. To stop and start the database manager for the current instance, enter the following commands:

```
db2stop  
db2start
```

Part 2. Setting Up Server Communications

Chapter 10. Using the Control Center to Configure Server Communications

When DB2 is installed, most communications protocols on your system are automatically detected and configured. This chapter describes how to use the Control Center to update your DB2 server's communication configuration settings after installation. It also describes how to add support for a new communications protocol. Note that when you add a new protocol to your network, you must configure the appropriate support on the DB2 server.

Note: You cannot use the Control Center to set up communications on a DB2 Universal Database Enterprise - Extended Edition server.

Before You Begin

To complete the steps in this section, you should know how to start the Control Center. For more information, see "Appendix A. Basic Task Knowledge" on page 447.

Before you can use the Control Center, ensure that you have a running Administration Server. By default, the installation program created and configured an Administration Server for you.

About the Control Center and Communications Protocols

The Control Center is a graphical tool used to administer DB2 databases. The Control Center's setup communications function allows you to display the protocols and configuration parameters that a server instance is configured to use. It also allows you to modify the parameter values of a configured protocol, as well as add or delete protocols.

When you add support for a new protocol to the server system, the setup communications function detects and generates server instance parameter values for the new protocol. You can accept or modify these values before use. When you remove support for an existing protocol from the server system, the setup communications function detects the protocol that has been removed and disables its use by the server instance.

You can add a protocol that has not been detected, however, you must supply all of the parameter values required before you proceed.

The setup communications function can be used to maintain communications for both local and remote server instances, provided that an Administration Server is running on the server system.



Modifying an instance's communications settings may require you to update the database connection catalogs on the client.

You can do this by:

- Using the Client Configuration Assistant on the client, select the database connection you want to change and then click on the **Properties** push button. This will launch a SmartGuide that will help you with the changes. For more information on cataloging using the Client Configuration Assistant, see "Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 29.
- Using the command line processor on the client to uncatalog and recatalog the node, depending on the values changed on the server. For more information, see "Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor" on page 41

Configuring DB2 Communications for Local Instances

To configure communications for local instances, perform the following steps:

- Step 1. Start the Control Center. For more information, see "Starting the DB2 Control Center" on page 447.
- Step 2. Click on the [+] beside a system's name to get the instances folder.
- Step 3. Select either the Databases or Gateway Connections folder and click on the [+] beside the **Instances** folder to get a list of instances on a particular system.
- Step 4. Select the instance that you want to configure and click on the right mouse button.
- Step 5. Select the **Setup communications** option from the pop-up menu. The Setup communications window opens.
- Step 6. Use the Setup communications window to configure communication protocols for the instance that you selected. Invoke the online help by clicking on the **Help** push button or by pressing the **F1** key.
- Step 7. You must stop and start the instance for these changes to take effect.
 - a. To stop the database manager instance, select the instance, click with the right mouse button and select the **Stop** option from the pop-up menu.
 - b. To start the database manager instance, select the instance, click with the right mouse button and select the **Start** option from the pop-up menu.

Configuring DB2 Communications for Remote Instances

To configure DB2 communications for remote instances, perform the following steps:

- Step 1. Start the Control Center. For more information, see “Starting the DB2 Control Center” on page 447.
- Step 2. If the system containing the remote instance you want is listed, click on the [+] sign beside the system name to get the Instances folder. Click on the [+] beside the Instances folder to get a list of the system’s instances, then go to step 13. If the system containing the remote instance you want is listed, but the instance you want does not appear under that system, go to step 8.
- Step 3. If the system containing the remote instance that you want to configure is not listed, select the **Systems** folder, click on the right mouse button and select the **Add** option. The Add System window opens.
- Step 4. To add a system to the Control Center, you can do one of the following:
 - Search the network for known systems.
 - a. Click on the **Refresh** push button.
 - b. Select the system you want to add from the **System name** drop-down list.
 - a. Fill in the appropriate protocol specific information
 - b. Click on the **Retrieve** push button.
- Step 5. Click on the **Apply** push button to add the system to the Control Center window.
- Step 6. Click on the **Close** push button.
- Step 7. Click on the [+] sign beside the system name you just added to get the Instances folder.
- Step 8. Select the **Instances** folder for the new system and click on the right mouse button.
- Step 9. Select the **Add** option. The Add Instance window opens.
- Step 10. Click on the **Refresh** push button to obtain a list of available instances.
- Step 11. Select the instance that you want to add from the **Remote instance** drop-down list and click on the **Apply** push button.
- Step 12. Click on the **Close** push button.
- Step 13. Select the instance you want to configure and click on the right mouse button.
- Step 14. Select the **Setup communications** option from the pop-up menu. The Setup Communications window opens.

- Step 15. Use the Setup Communications window to configure communication protocols for the instance. Click on the **Help** push button for more information.
- Step 16. You must stop and start the instance for these changes to take effect:
- a. To stop an instance, select the instance, click on the right mouse button, and select the **Stop** option.
 - b. To start an instance, select the instance, click on the right mouse button, and select the **Start** option.

Chapter 11. Using the Command Line Processor to Configure Server Communications

This section describes how to configure your server to accept inbound requests from remote client workstations. Also described in this section are considerations when configuring the DB2 Administration Server for communications. The Control Center and the Discovery function of the Client Configuration Assistant (CCA) are dependent on the Administration Server's protocol configuration.

Follow the instructions in this section if:

- You deselected a detected communication protocol when you installed DB2.
- You have added a communication protocol to your network since running the DB2 setup program.
- You are using a communications protocol that could not be detected by the DB2 setup program.
- You installed a UNIX-based DB2 product using your operating system's native installation tools.

You can use the Control Center's Setup Communications function to configure an instance for communications; however, it cannot be used to set up communications for an Administration Server or a DB2 Enterprise - Extended Edition server. For information on how to set up communications using the Control Center, see "Chapter 10. Using the Control Center to Configure Server Communications" on page 133.

For instructions on entering DB2 commands, see "Entering Commands Using the Command Center" on page 448 or "Entering Commands Using the Command Line Processor" on page 449.

Setting the DB2COMM Registry Parameter

Your server can support multiple communication protocols concurrently; however, you only need to enable the protocols that you want to use.

You must update the *db2comm* registry variable with the protocol that you now want to support.

The *db2comm* registry variable determines which protocol's connection managers will be enabled when the database manager is started. You can set this variable for multiple communication protocols by separating the keywords with commas.

For your DB2 server, *DB2COMM* can be any combination of the following keywords, separated by commas:

appc	starts APPC support
ipxspx	starts IPX/SPX support
netbios	starts NetBIOS support
npipe	starts NAMED PIPE support (for Windows 32-bit operating systems only)
tcpip	starts TCP/IP support

To avoid problems with the Control Center and the Client Configuration Assistant, ensure that the *DB2COMM* parameter is set in the DB2 registry using the *db2set* command using the *-i* option. It is not recommended that you use any other mechanism to set the *DB2COMM* value. For more information on the DB2 registry, refer to the *Administration Guide*.

To set the *db2comm* registry variable for the current instance, enter the **db2set DB2COMM=*protocol_names*** command. Change the *protocol_names* parameter to reflect those protocols that you want to start when the database manager is started.

For example, to set the database manager to start connection managers for the APPC and TCP/IP communication protocols, enter the following command:

```
db2set DB2COMM=appc,tcpip
db2stop
db2start
```

When the **db2start** command is reissued, connection managers for the protocols specified by the *db2comm* registry parameter are started.

If you are setting up communications for the Administration Server, enter the **db2set** command:

```
db2set DB2COMM=appc,tcpip -i DB2DAS00
db2admin stop
db2admin start
```

where **DB2DAS00** represents the name of the Administration Server. If you are not sure of the Administration Server name, enter the following command:

```
db2set DB2ADMINSERVER
```


If the *db2comm* registry variable is undefined or set to null, no protocol connection managers are started when the database manager is started.



If there are problems starting a protocol's connection managers, a warning message appears and the error messages are logged in the *db2diag.log* file located in the *INSTHOME/sql11ib/db2dump* directory for UNIX platforms or in the *x:\sql11ib\db2dump* directory for non-UNIX platforms.

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



You are now ready to configure the server workstation to use any of the following communication protocols:

- Named Pipes - see "Configuring Named Pipes on the Server".
- TCP/IP - see "Configuring TCP/IP on the Server".
- NetBIOS - see "Configuring NetBIOS on the Server" on page 143.
- IPX/SPX - see "Configuring IPX/SPX on the Server" on page 147.
- APPC - see "Configuring APPC on the Server" on page 152.

Configuring Named Pipes on the Server

To access a remote server through Named Pipes, you must first have installed and configured communication software for both the client and server workstations. See "Software Requirements" on page 4 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 9 for the supported communication protocols for your particular client and server.

If you have read and completed the instructions in "Setting the DB2COMM Registry Parameter" on page 137, there are no further steps required to set up Named Pipe communications on the server or to support the Administration Server.

Configuring TCP/IP on the Server

To access a remote server through TCP/IP, you must first have installed and configured communication software for both the client and server workstations. See "Software Requirements" on page 4 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 9 for the supported communication protocols for your particular client and server.

Before you complete the steps in this section, ensure that you have completed the instructions in "Setting the DB2COMM Registry Parameter" on page 137.



If you are setting up communications for the Administration Server, it was configured for TCP/IP (using the DB2 registered Port number 523) when it was created. There are no further steps required to enable the Administration Server to use TCP/IP.

The following steps are required to set up TCP/IP communications:

Step 1. Identify and record parameter values.

Step 2. At the server:

- a. Update the services file.
- b. Update the database manager configuration file.



Due to the characteristics of the TCP/IP protocol, TCP/IP may not be immediately notified of the failure of a partner on another host. As a result, a client application accessing a remote DB2 server using TCP/IP, or the corresponding agent at the server, may sometimes appear to be hung. DB2 uses the TCP/IP SO_KEEPALIVE socket option to detect when there has been a failure and the TCP/IP connection has been broken.

If you are experiencing problems with your TCP/IP connection, refer to the *Troubleshooting Guide* for information on how to adjust this parameter and other common TCP/IP problems.

1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 21. TCP/IP Values Required at the Server

Parameter	Explanation	Sample Value	Your Value
Connection Port • Connection Service name (<i>svcsname</i>) • Port number/Protocol (<i>port_number/tcp</i>)	<p>Values required in the services file.</p> <p>The Connection Service name is arbitrary, but must be unique within the services file. If you are using DB2 Enterprise - Extended Edition, ensure that these numbers do not conflict with the port numbers used by the Fast Communication Manager (FCM).</p> <p>The Connection Port must be unique within the services file.</p>	<p>server1</p> <p>3700/tcp</p>	

Table 21. TCP/IP Values Required at the Server (continued)

Parameter	Explanation	Sample Value	Your Value
Service name (<i>svcname</i>)	The name used to update the Service name (<i>svcname</i>) parameter in the database manager configuration file at the server. This value must be the same as the Connection Service name specified in the services file, if a service name is specified. You don't have to specify a service name in the <i>svcname</i> field, if you have directly specified a port number.	server1	

2. Configure the Server

The following steps describe how to set up your workstation to accept inbound client requests using this protocol. Replace the sample values with your values and record them on your worksheet.

A. Update the Services File

The TCP/IP services file specifies the ports that server applications can listen on for client requests. To update the services file, you can either:

- Update the services file and specify the ports that you want the server to listen on for incoming client requests, or
- Update the services file if you directly specified a port number in the *svcname* field.

The location of the services file depends on the operating system:

OS/2 path is specified by the ETC environment variable. To resolve this path, enter the **set etc** command.

UNIX /etc

Windows NT and Windows 2000 \winnt\system32\drivers\etc

Windows 9x \windows

Using a text editor, add the Connection entry to the services file for TCP/IP support. For example:

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

where:

server1 represents the connection service name

3700 represents the connection port number

tcp represents the communication protocol that you are using

B. Update the Database Manager Configuration File

You must update the database manager configuration file with the Service name (*svcename*) parameter.

To update the database manager configuration file, perform the following steps:

- Step 1. Log on to the system as a user with System Administrative (SYSADM) authority. For more information, see “Working with the System Administrative Group” on page 451.
- Step 2. If you are using a UNIX server, set up the instance environment and invoke the DB2 command line processor as follows:
 - a. Run the start up script as follows:

```
. INSTHOME/sql1lib/db2profile    (for Bash, Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

where *INSTHOME* is the home directory of the instance.
 - b. Start the DB2 command line processor by entering the **db2** command.
- Step 3. Update the database manager configuration file with the Service name (*svcename*) parameter by entering the following commands:

```
update database manager configuration using svcename svcename
db2stop
db2start
```

For example, if the Connection Service name in the services file was entered as *server1*, enter the following commands:

```
update database manager configuration using svcename server1
db2stop
db2start
```



The *svcename* used must match the Connection Service name specified in the services file.

After the database manager is stopped and started again, view the database manager configuration file to ensure that these changes have taken effect. View the database manager configuration file by entering the following command:

```
get database manager configuration
```

Configuring NetBIOS on the Server

To access a remote server through NetBIOS, you must first have installed and configured communication software for both the client and server workstations. See “Software Requirements” on page 4 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the supported communication protocols for your particular client and server.

Before you complete the steps in this section, ensure that you have completed the instructions in “Setting the DB2COMM Registry Parameter” on page 137.

The following steps are required to set up NetBIOS communications:

Step 1. Identify and record parameter values.

Step 2. Configure the server:

a. Configure the NetBIOS Interface.

b. Update the database manager configuration file.

1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 22. NetBIOS Values Required at the Server

Parameter	Description	Sample Value	Your Value
Adapter number (<i>adapter_number</i>)	The local logical adapter that will be used for the NetBIOS connection. The server uses adapter 0 if this parameter is not configured.	0	

Table 22. NetBIOS Values Required at the Server (continued)

Parameter	Description	Sample Value	Your Value
Workstation name (<i>nname</i>)	<p>The NetBIOS name of the server workstation.</p> <p><i>nname</i> is a name, chosen by the user, which must be unique among all NetBIOS nodes in the network.</p> <p>If you are using DB2 Enterprise - Extended Edition, ensure that the last 4 characters are unique among all NetBIOS nodes in the network.</p> <p>For more information on workstation names, see "Workstation Name (<i>nname</i>) Rules" on page 512</p>	server1	

2. Configure the Server

The following steps describe how to set up your workstation to accept inbound client requests using this protocol. Replace the sample values with your values and record them on your worksheet.

A. Configure the NetBIOS Interface

DB2 uses registry parameters to control its use of the NetBIOS resources on the server. Use the *db2nbadapters* registry parameter when you want to specify a value other than the default Logical adapter number 0.



For the DB2 server, set the *db2nbadapters* parameter by entering the **db2set db2nbadapters=adapter_number** command. The *adapter_number* can be a list of adapter numbers separated by commas.

For the Administration Server, set the *db2nbadapters* parameter by entering the **db2set db2nbadapters=adapter_number -i DB2DAS00** command. The *adapter_number* can be a list of adapter numbers separated by commas.

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

To view or modify the NetBIOS interface configuration, which maps network routes to adapter numbers, perform the following steps:

- For OS/2:
 - Step 1. Double-click on the **MPTS** icon.
 - Step 2. Click on the **Configure** push button.

- Step 3. Select the **LAN adapters and protocols** radio button and click on the **Configure** push button.
 - Step 4. Record the Logical adapter number associated with the **IBM OS/2 NETBIOS** entry in the Current Configuration window.
 - Step 5. If you want to change the logical adapter number associated with NetBIOS, select the **IBM OS/2 NetBIOS** entry from the Current Configuration window and click on the **Change number** push button.
 - Step 6. Select a logical adapter number and click on the **Change** push button.
 - Step 7. Record the new logical adapter number associated with the **IBM OS/2 NETBIOS** entry in the Current Configuration window on your worksheet.
 - Step 8. Click on **OK**.
 - Step 9. Click on the **Close** push button.
 - Step 10. Click on the **Exit** push button.
 - Step 11. Ensure that the **Update CONFIG.SYS** check box is selected and click on the **Exit** push button.
 - Step 12. Click on the **Exit** push button.
 - Step 13. The Network Settings Change pop-up box opens. You must shutdown and reboot your system for these changes to take effect. Select the **No** push button to shutdown and reboot your system at a later time.
- For Windows NT and Windows 2000:
 - Step 1. Click on **Start** and select **Settings** → **Control Panel**.
 - Step 2. Double-click on the **Network** icon and select the **Services** tab.
 - Step 3. Select the **NetBIOS Interface** icon from the Network Services window and click on the **Properties** push button.
 - Step 4. Scroll through the network routes until you find the Logical adapter number associated with **Nbf** and record it on your worksheet. If this adapter number is associated with **Nbf**, and you do not want to change it, go to step 7.
 - Step 5. To change the logical adapter number associated with **Nbf**, select the associated **LAN Number**, and click on the **Edit** push button. Enter the new adapter number 0 or the value that you set for the *db2nbadapters*) registry variable.
 - Step 6. Record the new adapter number associated with **Nbf** on your worksheet.
 - Step 7. Click **OK**.
 - Step 8. Click on the **Close** push button. The Network Settings Change window opens.

- Step 9. You must shutdown and reboot your system for these changes to take effect. Select the **Yes** push button to shutdown and reboot your system or select the **No** push button to shutdown and reboot your system at a later time.



Each adapter number must be uniquely associated with a network route. Windows NT and Windows 2000 have a built in checking facility that will not allow you to specify the same adapter number for different network routes. If a Network route already exists that is using the adapter number *0*, assign a different number to that route. (The valid range for adapter numbers is *0* to *255*.) This will allow you to select *0* as the adapter number that corresponds to **Nbf**. Approve the changes by clicking on **OK**.

B. Update the Database Manager Configuration File

You must update the database manager configuration file with the server's Workstation name (*nname*) parameter.

To update the database manager configuration file, perform the following steps:

- Step 1. Log on to the system as a user with System Administrative (SYSADM) authority. For more information, see "Working with the System Administrative Group" on page 451.
- Step 2. Update the database manager configuration file with the server's Workstation name (*nname*) parameter by entering the following commands:

```
update database manager configuration using nname nname
db2stop
db2start
```

For example, if the server's workstation name (*nname*) is *server1*, use:

```
update database manager configuration using nname server1
db2stop
db2start
```

After the database manager is stopped and started again, view the database manager configuration file to ensure that these changes have taken effect. View the database manager configuration file by entering the following command:

```
get database manager configuration
```

If you are configuring the Administration Server for NetBIOS, you must update the admin server configuration file. Use the following command:


```
update admin configuration using nname nname
db2admin stop
db2admin start
```

Autostarting DB2 with NetBIOS (for Windows NT only)

If your NetBIOS protocol was configured when you installed the server (or the instance-owning machine on DB2 Enterprise - Extended Edition), the setup program automatically created a NetBIOS dependency for the server and Administration Server. You will need to manually create a dependency on NetBIOS for any new instances.

To create this dependency, perform the following steps:

Step 1. Go to the `x:\sql11ib\bin` directory, where `x`: represents the drive on which the server was installed

Step 2. Enter the **db2depnb** command as follows:

```
db2depnb instance_name
```

where *instance_name* is the name of the instance that you want to create a dependency for.

This records a dependency on the startup order which causes NetBIOS to start before a DB2 instance starts.



If you remove the NetBIOS protocol from your network, you **must** remove the dependencies that were created during installation, and any dependencies that you created for additional instances. Failure to remove these dependencies may cause problems when running DB2 after the NetBIOS protocol has been removed from the network.

To remove a dependency, enter the **db2depnb** command as follows:

```
db2depnb instance_name /r
```

where *instance_name* is the name of the instance for which you are removing a dependency.

Configuring IPX/SPX on the Server

To access a remote database server through the IPX/SPX communications software, you must first have installed and configured communication software for both the client and server workstations. See “Software Requirements” on page 4 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the supported communication protocols for your particular client and server.

Before you complete the steps in this section, ensure that you have completed the instructions in “Setting the DB2COMM Registry Parameter” on page 137.

Depending on the platform of your server, you can support IPX/SPX client communications via Direct Addressing or File Server Addressing.

Direct Addressing:

The client connects to the server instance by directly specifying the IPX/SPX internetwork address of the server instance (bypassing the NetWare file server). Using this method, a NetWare file server is not required on the network. Configure a server using this addressing method if it will be accessed solely by clients using Direct Addressing.

File Server Addressing:

The server instance registers its address at the NetWare file server. The client connects to the server instance using the address stored at a NetWare file server. Configure a server using this addressing method if it will be accessed by clients using File Server and/or Direct Addressing.

For a list of the supported IPX/SPX addressing methods for your server, see Table 23. For a list of the supported IPX/SPX addressing methods for DB2 clients, see Table 14 on page 63.

Table 23. IPX/SPX Supported Communication Methods for a Server

Server	Direct Addressing	File Server Addressing
AIX	*	*
HP-UX	Not Supported	
Linux	Not supported	
OS/2	*	*
Solaris	*	
Windows NT	*	
Windows 2000	*	

The following steps are required to set up IPX/SPX communications:

Step 1. Identify and record parameter values.

Step 2. Configure the server:

- a. Update the database manager configuration file.
- b. Register the server on the NetWare file server (for File Server Addressing only).

1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 24. IPX/SPX Values Required at the Server

Parameter	Explanation	Our Example	Your Value
File server name (FILESERVER)	<p>Direct Addressing: A * value indicates that you are using Direct Addressing.</p> <p>File Server Addressing: The name of the NetWare file server where the database server instance is registered. This parameter must be entered in uppercase.</p>	<p>Direct Addressing</p> <p>*</p> <p>File Server Addressing</p> <p>DB2INST1</p>	
DB2 server object name (OBJECTNAME)	<p>Direct Addressing: A * value indicates that you are using Direct Addressing.</p> <p>File Server Addressing: The database manager server instance represented as the object <i>OBJECTNAME</i> on the NetWare file server. The server's IPX/SPX internetwork address is stored and retrieved from this object.</p> <p>This parameter must be entered in uppercase and be unique on the NetWare file server system.</p>	<p>Direct Addressing</p> <p>*</p> <p>File Server Addressing</p> <p>NETWSRV</p>	
Socket number (IPX_SOCKET)	<p>Represents the connection end point in a server's internetwork address. It must be unique for all server instances, and all IPX/SPX applications running on the workstation. It is entered in hexadecimal format.</p> <p>DB2 has registered well known sockets with Novell in the range 0x879E to 0x87A1. If you run more than 4 instances on the server machine, you must prevent socket collisions for instances 5 and up by choosing a socket number that is not 0x0000, in the dynamic socket range 0x4000 to 0x7FFF, or in the range 0x8000 to 0x9100 (these are well known sockets that are registered to various applications). The maximum value for this parameter is 0xFFFF.</p>	879E (default)	

Table 24. IPX/SPX Values Required at the Server (continued)

Parameter	Explanation	Our Example	Your Value
IPX/SPX Internetwork Address	The internetwork address required when configuring a client to communicate with a server using Direct Addressing.	09212700.400011527745.879E	



The following characters are not valid in the File server (*FILESERVER*) or the DB2 server object name (*OBJECTNAME*) parameters: / \ ; , * ?

2. Configure the Server

The following steps describe how to set up your workstation to accept inbound client requests using this protocol. Replace the sample values with your values and record them on your worksheet.

A. Update the Database Manager Configuration File

You must update the database manager configuration file with the File server (*FILESERVER*), DB2 server object name (*OBJECTNAME*), and Socket number (*IPX_SOCKET*) parameters.

To update the database manager configuration file, perform the following steps:

- Step 1. Log on to the system as a user with System Administrative (SYSADM) authority. For more information, see “Working with the System Administrative Group” on page 451.
- Step 2. If you are using a UNIX server, set up the instance environment and invoke the DB2 command line processor as follows:
 - a. Run the start up script as follows:


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

where *INSTHOME* is the home directory of the instance.
 - b. Start the DB2 command line processor by entering the **db2** command.
- Step 3. In the database manager configuration file, update the File server (*FILESERVER*), DB2 server object name (*OBJECTNAME*), and Socket number (*IPX_SOCKET*) parameters by entering the following commands:

```
update dbm cfg using fileserver FILESERVER objectname OBJECTNAME \
ipx_socket IPX_SOCKET
db2stop
db2start
```

Note: If you are configuring the Administration Server instance (DAS) for communications, you need to update the Administration Server's configuration file by entering the **db2 update adm cfg** command. The Administration Server has a reserved IPX socket, 879A, that should be used when you update this configuration file.

Direct Addressing

For example, if you have chosen a value of 87A2 for the Socket number (*IPX_SOCKET*) parameter, use:

```
update dbm cfg using fileserver * objectname * ipx_socket 879F
db2stop
db2start
```

File Server Addressing

For example, if the name of the File server (*FILESERVER*) is NETWSRV, the DB2 server's object name (*OBJECTNAME*) is DB2INST1, and you have chosen a value of 879F for the Socket number (*IPX_SOCKET*) parameter, use:

```
update dbm cfg using fileserver NETWSRV objectname DB2INST1 \
ipx_socket 879F
db2stop
db2start
```

After the database manager is stopped and started again, view the database manager configuration file to ensure that these changes have taken effect. View the database manager configuration file by entering the following command:

```
get database manager configuration
```



If you are planning to only support clients that use Direct Addressing, you will be required to enter the server's IPX/SPX internetwork address as the DB2 server object name when cataloging the node on the client.

Determine the value for the *OBJECTNAME* parameter by entering the **db2ipxad** command on the server. This command is located in the `sqllib/bin/` directory for UNIX platforms, or the `sqllib\bin\` directory for non-UNIX platforms.

Make note of this output on your worksheet for use when you configure an IPX/SPX client.

If you are planning to support clients using only Direct Addressing, you are now finished configuring your server for inbound IPX/SPX communications.

B. Register the Server Instance on the NetWare File Server (for File Server Addressing Only)

The server must be registered *after* the database manager configuration file has been updated with the IPX/SPX parameters. To register the server instance at the NetWare File server, enter the following command in the command line processor:

```
register db2 server in nwbindery user USERNAME password PASSWORD
```



If you want to modify the IPX/SPX configuration parameters or change the DB2 IPX/SPX internetwork address, you need to deregister your server instance, before making the changes, and reregister the server instance after the changes are made.

Notes:

1. *USERNAME* and *PASSWORD* must be specified in UPPERCASE.
2. The *USERNAME* and *PASSWORD* are used to log on to the NetWare file server and must have Supervisor/Administrator or Workgroup Manager security equivalence.
3. If you want to register at a NetWare 4.x file server (which uses directory services and provides bindery emulation capability), the *USERNAME* used must be created within the same context as the current bindery context used by Directory Services when it does bindery emulation. The bindery emulation context currently in use can be found by checking the bindery emulation setting on the NetWare 4.x file server (for example, by using the *SERVMAN* utility). For more information, refer to your IPX/SPX documentation.

Configuring APPC on the Server

This section describes how to set up APPC communications on a DB2 Connect or DB2 Universal Database server for inbound client connections. To access a remote server through APPC, you must first have installed and configured communication software on both the client and server workstations. These connections can be from workstations running DB2 clients, host database clients, or AS/400 database clients.

See “Software Requirements” on page 4 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the supported communication protocols for your particular client and server.

Before you complete the steps in this section, ensure that you have completed the instructions in “Setting the DB2COMM Registry Parameter” on page 137.

The following steps are required to set up APPC communications:

Step 1. Identify and record parameter values.

Step 2. Configure the server:

- a. Update the database manager configuration file.
- b. Configure the APPC communications subsystem.

1. Identify and Record Parameter Values

Before you configure the server, fill in copies of the *Your Value* column of the worksheet in Table 25. You can then use the worksheet to configure APPC communications for inbound connections. During the configuration process, replace the sample values in the instructions with your values from the worksheet, using the boxed numbers (for example, **1**) as reference.



The worksheet and configuration instructions supply suggested or sample values for required configuration parameters. For other parameters, use the communications program's default values. If your network configuration is different from what is used in the instructions, consult your Network Administrator for the appropriate network values.

Table 25. APPC Values Required at the Server

Ref. #	Name on the Server	Sample Value	Your Value
1	Network ID	SPIFNET	
2	Local Control Point name	NYX1GW	
3	Local node or Node ID	071 27509	
4	Local LU name	NYX1GW0A	
5	Local LU alias	NYX1GW0A	
6	Mode name	IBMRDB	
7	Service TP name	X'07'6DB	
8	Application TP name	DB2DRDA	

For each server that you are connecting to, fill in a copy of the worksheet as follows:

1. For Network ID (**1**), determine the network name of the server workstation.
2. Determine the Local Control Point name or Control Point LU (**2**) to be used for the server workstation. This value is usually the same as the Local node or Physical Unit name for the system.
3. For Local node or Node ID (**3**), determine the IDBLK and IDNUM of the server workstation. The default value should be correct, or you can obtain the required values from your Network or System Administrator.
4. Determine the Local LU name (**4**) to be used by the server. If you use a Syncpoint Manager (SPM) to manage multisite updates (two-phase

commit), the Local LU should be the LU used for the SPM. In this case, that LU cannot also be the Control Point LU.

5. For Local LU alias (**5**), you usually use the same value as for the Local LU name (**4**).
6. For Mode name (**6**), usually the default IBMDRB is sufficient.
7. For Service TP name (**7**) and Application TP name (**8**), choose a name of up to 64 characters, or use the default transaction programs, X'07'6DB and DB2DRDA.



Record the following entries from the worksheet so that they can be used to configure communications and directories for each client from which you will be accepting connections:

- Network ID (**1**)
- Local Control Point name (**2**)
- Local LU name (**4**)
- Mode name (**6**)
- Transaction Program (TP) name (**7**) or (**8**).

2. Configure the Server

The following section describe how to configure APPC on the server. Replace the sample value with your value as recorded on your worksheet.

A. Update the Database Manager Configuration File



If you want to use only the default Transaction Programs (TPs) for a single DB2 instance, you do not need to configure the *tpname* database manager configuration parameter. Skip this step and go to “B. Configure the APPC Communications Subsystem” on page 156.

The default Transaction Program (TP) names are DB2DRDA and X'07'6DB. To configure a DB2 instance to listen for a TP other than, or in addition to, the default TPs, you must configure the *tpname* database manager configuration parameter. You must also configure a unique TP name if you have multiple instances on the server.

To update the database manager configuration file with the TP name (**8**), perform the following steps:

- Step 1. Log on to the system as a user with System Administrative (SYSADM) authority. For more information, see “Working with the System Administrative Group” on page 451.
- Step 2. For UNIX servers, set up the instance environment and invoke the DB2 command line processor as follows:
 - a. Run `db2profile` or `db2cshrc` as follows:


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

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

- b. Start the DB2 command line processor by entering the **db2** command.

Step 3. Update the database manager configuration file with the server's TP name (*tpname*) by entering the following commands:

```
update dbm cfg using tpname tpname
db2stop
db2start
```

For example, if the server's transaction program name is DB2DRDA, enter the following commands:

```
update dbm cfg using tpname DB2DRDA
db2stop
db2start
```

If you are configuring the Administration Server to use APPC, you must also update the admin server configuration file. If the Administration Server's TP name is DB2ADMIN, enter the following command:

```
update admin configuration using tpname DB2ADMIN
db2admin stop
db2admin start
```

If your server contains multiple instances, each accepting connections using APPC, then each instance requires a unique TP on which to listen. Only one of these instances can listen for the default TPs. The DB2SERVICETPINSTANCE registry variable controls which instance listens for the default TPs (on OS/2, Windows NT, Windows 2000 or AIX). This variable must be set to the name of the instance you want to listen for these default TPs, and set globally, so that all instances on a given machine can obtain its value.

For example, to set DB2SERVICETPINSTANCE globally, and to ensure that the instance MYINST1 listens for default TPs, enter the following command:

```
db2set -g DB2SERVICETPINSTANCE=MYINST
```

Consider the following server configuration:

- Two instances are defined on the server, MYINST1 and MYINST2.
- For MYINST1, the *tpname* parameter is set to MYTP1.
- For MYINST2, the *tpname* parameter is set to MYTP2.
- DB2SERVICETPINSTANCE is set to MYINST1.

In this configuration, MYINST1 instance will listen for the MYTP1 *in addition to* the default TPs DB2DRDA and X'07'6DB. The MYINST2 instance will continue to listen for the MYTP2 TP.

If this registry value is not set globally, and multiple instances are configured to support APPC connections, then unpredictable results will occur; that is, it will not be possible to predict which instance listens for the default TPs.

B. Configure the APPC Communications Subsystem

To configure your DB2 server to accept remote clients using APPC, you need to update the APPC communications subsystem to support the TP name that the server will use.



Go to the section that provides the APPC inbound client connection instructions for your operating system:

- “Configuring IBM eNetwork Communications Server for AIX”
 - “Configuring IBM eNetwork Communications Server for Windows NT” on page 162
 - “Configuring IBM eNetwork Communications Server for OS/2” on page 166
 - “Configuring IBM eNetwork Personal Communications for Windows NT” on page 171
 - “Configuring Microsoft SNA Server for Windows NT” on page 175
 - “Configuring SunLink SNA for Solaris” on page 179
-

Configuring IBM eNetwork Communications Server for AIX: This section describes how to configure your IBM eNetwork Communications Server for AIX (CS/AIX) to accept inbound APPC client connections. CS/AIX is the only product supported for this purpose. Before you begin, ensure that your workstation has CS/AIX installed.

For more information on setting up your environment, refer to the online help supplied with CS/AIX.

The following assumptions have been made:

- The basic installation of the CS/AIX package has already been completed.
- DB2 Connect or DB2 Universal Database for AIX has been installed.
- The user is logged on as root.

Use the *Your Value* entries from the worksheet in Table 25 on page 153 to complete the steps below.



DB2 automatically configures CS/AIX with the TP parameters required to accept inbound APPC connections. These instructions describe the remaining steps required to configure your APPC communications for inbound connections.

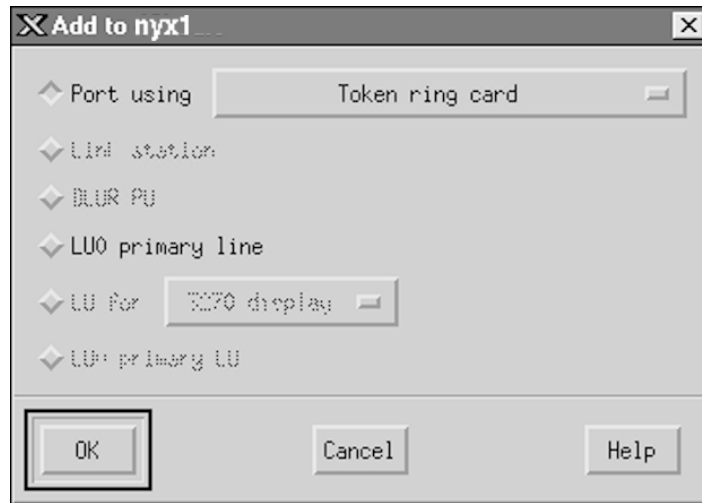
To configure CS/AIX to accept inbound APPC connections, perform the following steps:

- Step 1. Log on to the system as a user with root authority and start the `/usr/bin/X11/xsnaadmin` program. The Node window for the server opens.
- Step 2. Define a node.
 - a. Select **Services->Configure Node Parameters** from the menu bar. The Node Parameters window opens.

The screenshot shows the 'Node parameters' dialog box. The 'APPN support' dropdown is set to 'End node'. The 'SNA addressing' section contains the following fields: 'Control point name' with the value 'SPIFNET.NYX1GM', 'Control point alias' with the value 'NYX1GM', and 'Node ID' with the values '071' and '27509'. The 'Description' field is empty. The bottom of the dialog features buttons for 'OK', 'Advanced...', 'Cancel', and 'Help'.

- b. Click on the **APPN support** drop down box and select the **End node** option.
 - c. Enter your Network ID (**1**) and Control Point name (**2**).
 - d. Enter your Control Point name (**2**) in the **Control point alias** fields.
 - e. Enter your Node ID (**3**) in the **Node ID** fields.
 - f. Click **OK**.
 - Step 3. Define a port.
 - a. Select the **Connectivity and Dependent LUs** window.

- b. Click on the **Add** push button. The Add to node window opens.



- c. Select the **Port using** radio button.
- d. Click on the **Port using** drop down box and select the appropriate port. For the purpose of our example, we will select the **Token ring card** option.

- e. Click **OK**. The Port window for the chosen port type opens.

Token ring SAP

SNA port name: TRSAPO

Token ring card: 0

Local link name:

Local SAP number: 04

Initially active

HPR

Use HPR on implicit links

Use HPR link-level error recovery

Connection network

Define on connection network

CN name: SPIFNET + NYX1GW

Description:

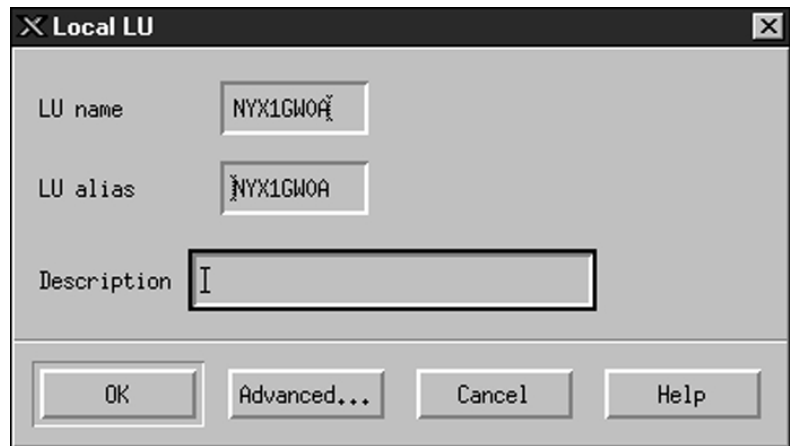
OK Advanced... Cancel Help

- f. Enter a name for the port in the **SNA port name** field.
- g. Select the **Initially active** checkbox.
- h. From the **Connection network** box, select the **Define on a connection network** checkbox.
- i. Enter your Network ID (**1**) and Control Point name (**2**) in the **CN name** fields.
- j. Click **OK**. The Token Ring SAP window closes and a new port appears in the **Connectivity and Dependent LUs** window.

Step 4. Define a local LU.

- a. Select the Independent local LUs window.

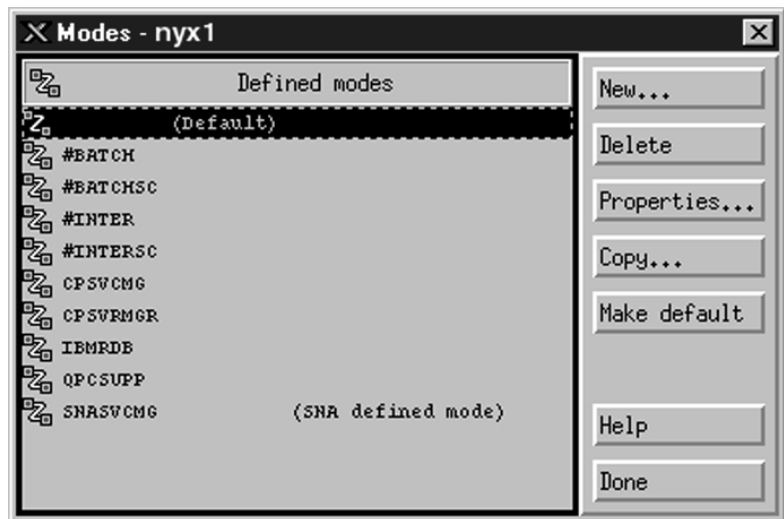
- b. Click on the **Add** push button. The Local LU window opens.



- c. Enter your independent Local LU name (**4**) in the **LU name** field.
- d. Enter the same name (**5**) in the **LU alias** field.
- e. Click **OK**. The new LU appears in the Independent local LUs window.

Step 5. Define a mode.

- a. Select **Services->APPC->Modes** from the menu bar. The Modes window opens.



- b. Click on the **New** push button. The Mode window opens.

The screenshot shows a dialog box titled "Mode" with the following fields and options:

- Name:** IBMRED
- Session limits:**
 - Initial: 20
 - Maximum: 32767
 - Min con. winner sessions: 10
 - Min con. loser sessions: 10
 - Auto-activated sessions: 04
- Receive pacing window:**
 - Initial: 8
 - Maximum: (Optional)
- Specify timeout
- Restrict max RU size
- Description:** (empty)
- Buttons:** OK, Cancel, Help

- c. Enter your Mode name (**6**) in the **Name** field.
- d. The configuration values below are suggested for the following fields:

- **Initial Session limits:** 20
- **Maximum Session limits:** 32767
- **Min con. winner sessions:** 10
- **Min con. loser sessions:** 10
- **Auto-activated sessions:** 4
- **Initial Receive pacing window:** 8

These values are suggested because they are known to work. You will need to tailor these values so that they are optimized for your particular application environment.

- e. Click **OK**. The new mode appears in the Mode window.

- f. Click **Done**.
- Step 6. Close the CS/AIX administration program.
- Step 7. Test the APPC connection.
- a. Start the SNA subsystem by entering the `/usr/bin/sna start` command. You can enter the `/usr/bin/sna stop` command to stop the SNA subsystem first, if required.
 - b. Start the SNA administration program. You can enter either the `/usr/bin/snaadmin` command or the `/usr/bin/X11/xsnaadmin` command.
 - c. Start the subsystem node. Select the appropriate node icon in the button bar, and click on **Start**.

You have now finished setting up your workstation for inbound APPC communications.

Configuring IBM eNetwork Communications Server for Windows NT: This section describes how to configure IBM eNetwork Communications Server for Windows NT (CS/NT) to accept inbound APPC client connections.

Before you begin, ensure that the IBM Communications Server for Windows NT you installed complies with the following restrictions:

- ___ 1. APAR fixes JR11529 and JR11170. These fixes are required to enable cancelling of queries in progress by using **Ctrl-Break** or issuing the SQLCancel ODBC/CLI call.
- ___ 2. IBM Communications Server IEEE 802.2 LAN interface (an installation option for Communications Server) or LLC2 driver installed from the IBM Communications Server installation directory. During installation CS/NT asks if you want to install LLC2. If you are not sure whether LLC2 was installed with your copy of CS/NT, you can find out as follows:
 - Step a. Click on **Start** and select **Settings->Control Panel**.
 - Step b. Double-click on the **Network** icon.
 - Step c. In the Network window, click on the **Protocols** tab. IBM LLC2 Protocol must be one of the protocols listed. If it is not, you need to install this protocol from your IBM Communications Server for Windows NT software. Refer to its documentation for instructions.

For more information on setting up your environment, refer to the online help supplied with CS/NT, or to the following publications:

- *Connectivity Supplement*
- *DRDA Connectivity Guide*

The following assumption has been made:

- The basic installation of the IBM eNetwork Communication Server for Windows NT package has already been completed.

Use the *Your Value* entries from the worksheet in Table 25 on page 153 to complete the steps below.

To configure CS/NT to accept inbound APPC connections, perform the following steps:

Step 1. Start IBM Communications Server for Windows NT.

- a. Click **Start** and select **Programs->IBM Communications Server->SNA Node Configuration**. The IBM Communications Server SNA Node Configuration window opens.
- b. Select **File->New->Advanced**.

Step 2. Configure the node.

- a. In the **Configuration options** box, select **Configure Node**, then click on the **New** push button. The Define the Node window opens.
- b. Enter your Network ID (**1**) and local Control Point name (**2**) in the **Fully qualified CP name** fields.
- c. Enter the same name (**2**) in the **CP alias** field .
- d. Enter your Node ID (**3**) in the **Local Node ID** field.
- e. Select the **End Node** radio button.
- f. Click **OK**.

Step 3. Configure devices.

- a. In the **Configuration options** field, select **Configure devices**.
- b. Select the appropriate DLC from **DLCs** field. These instructions use the **LAN** DLC.
- c. Click on the **New** button. The appropriate window opens with default values displayed. In this case, the Define a LAN Device window opens.
- d. Click **OK** to accept the default values.

Step 4. Configure the gateway.



Perform this step only if you are setting up Communications Server to accept requests from Communications Server for Windows NT SNA API Client.

- a. In the **Configuration options** field, select **Configure the Gateway**, then click on the **New** button. The Define Gateway window opens.
- b. Select the **SNA Clients** tab.

- c. Select the **Enable SNA API Client Services** check box.
 - d. Click **OK** to accept the default values.
- Step 5. Configure modes.
- a. In the **Configuration options** field, select **Configure modes**, then click on the **New** button. The Define a Mode window opens.
 - b. Enter your Mode name (**6**) in the **Mode name** field.
 - c. Select the **Advanced** tab.
 - d. Select **#CONNECT** from the **Class of Service Name** field.
 - e. Click **OK**.
- Step 6. Configure Local LU 6.2.
- a. In the **Configuration options** field, select **Configure local LU 6.2**, then click on the **New** push button. The Define a Local LU 6.2 window opens.
 - b. Enter your Local LU name (**4**) in the **Local LU name** field.
 - c. Enter a value for the **LU session limit** field. The default, 0, specifies the maximum allowed value.
 - d. Accept the defaults for the other fields and click **OK**.
- Step 7. Create a service Transaction Program (TP).
- a. In the **Configuration options** field, select **Configure Transaction Programs**.
 - b. Click on the **New** push button. The Define a Transaction Program window opens.
 - c. Select the **Basic** tab.
 - d. Select the **Service TP** check box.
 - e. Specify a service TP (**7**) in the **TP name** field.
 - f. Select the **Background Process** check box.
 - g. Select the **Advanced** tab.
 - h. Change the default in the **Receive Allocate timeout** field to 0 (no timeout).
 - i. If you are configuring Communications Server for use with Communication Server SNA Client, select the **For SNA API Client use** check box.
 - j. Accept the defaults for the other fields.
 - k. Click **OK**.
- Step 8. Create an Application Transaction Program.
- a. In the **Configuration options** field, select **Configure Transaction Programs**, then click on the **New** push button. The Define a Transaction Program window opens.
 - b. Select the **Basic** tab.

- c. Clear the **Service TP** check box.
- d. Specify an application TP name (**8**) in the **TP name** field.
- e. Select the **Background Process** check box.
- f. Select the **Advanced** tab.
- g. Change the default in the **Receive Allocate timeout** field to 0 (no timeout).
- h. If you are configuring Communications Server for use with Communication Server SNA Client, select the **For SNA API Client use** check box.
- i. Accept the defaults for the other fields and click **OK**.

Step 9. Save the configuration.

- a. Select **File->Save As**. The Save As window opens.
- b. Type in a file name, for example *ny3.acg*, and click **OK**.
- c. In the window that opens, you are asked if you want this configuration to be the default. Click on the **Yes** button.

Step 10. Update the environment.

IBM Communications Server uses an environment variable called APPCLLU to set the default Local LU used for APPC communications. You may set this variable on a per-session basis by opening a command window and entering `set appcllu=local_lu_name`, where *local_lu_name* represents the name of the Local LU you want to use.

However, you will probably find it more convenient to permanently set the variable. To permanently set the variable in Windows NT, perform the following steps:

- a. Click **Start** and select **Settings->Control Panel**.
- b. Double-click on the System icon. The System Properties window opens.
- c. Select the **Environment** tab.
- d. Type APPCLLU in the **Variable** field.
- e. Type your Local LU name (**4**) in the **Value** field.
- f. Click on the **Set** push button to accept the changes.
- g. Click **OK** to exit the System Properties window. The environment variable will now remain set for future sessions.

Step 11. Start SNA Node Operations.

- a. Click **Start** and select **Programs->IBM Communication Server->SNA Node Operations**. The **SNA Node Operations** window opens.
- b. Select **Operations->Start Node** from the menu bar.

- c. In the window that opens, select the configuration file you saved in the previous step (for example, ny3.acg) and click **OK**.
- Step 12. After installing Communications Server you should register it as a Windows NT Service. This will automatically start Communications Server when the machine is booted.

To register Communications Server as an NT service enter one of the following commands:

```
csstart -a
```

to register Communications Server with the default configuration, or:

```
csstart -a c:\ibmcs\private\your.acg
```

where `c:\ibmcs\private\your.acg` represents the name of the non-default Communications Server configuration file that you want to use.

Whenever your machine is booted in the future, Communications Server will be started automatically with the required configuration file.



Now that you have configured the server, you are ready to install a DB2 client. For more information, go to “Chapter 2. Installing DB2 Clients” on page 11.

Configuring IBM eNetwork Communications Server for OS/2: This section describes how to configure IBM eNetwork Communications Server for OS/2 V5 (CS/2) to accept inbound APPC client connections.

Before you begin, ensure that your workstation has CS/2 V5 or later for OS/2 installed.

For more information on setting up your environment, refer to the online help supplied with CS/2, or to the following publications:

- *Connectivity Supplement*
- *DRDA Connectivity Guide*

The following assumptions have been made:

- The basic installation of the IBM eNetwork Communication Server V5 for OS/2 package has already been completed.
- DB2 Connect or DB2 Universal Database for OS/2 has been installed.

Use the *Your Value* entries from the worksheet in Table 25 on page 153 to complete the following steps.

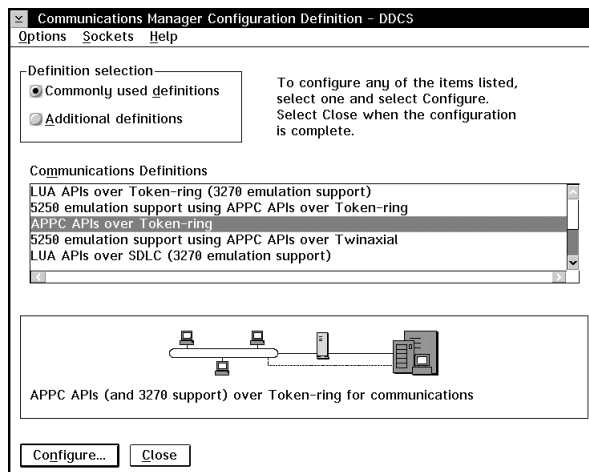


These instructions describe how to create new profiles within a new configuration. If you are modifying an existing configuration, you may need to delete some profiles before you can verify the configuration.

To configure your system, perform the following steps:

Step 1. Start a new configuration.

- a. Double-click on the **IBM eNetwork Communications Server** icon.
- b. Double-click on the **Communications Manager Setup** icon.
- c. On the Communications Manager Setup panel, click on the **Setup** push button.
- d. On the Open Configuration panel, provide a name for a new configuration file and click **OK**. The Communications Manager Configuration Definition window opens.



Step 2. Configure the protocol.

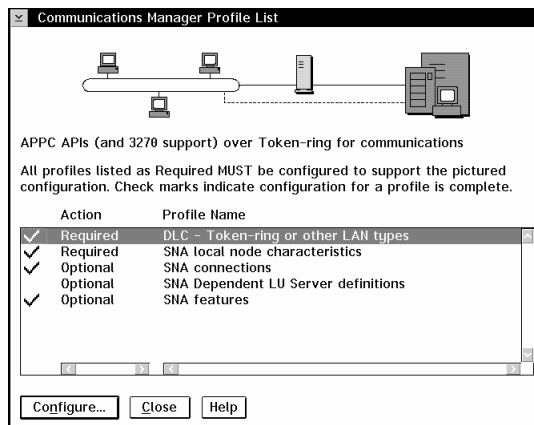
- a. Select the **Commonly used definitions** radio button.
- b. In the **Communications Definitions** box, select the protocol that you want to use.



These instructions use APPC APIs over Token-Ring.

- c. Click on the **Configure** push button. The APPC APIs Over Token-Ring window opens.

- d. Enter your Network ID (**1**) and local Control Point name (**2**) in the corresponding fields.
- e. Click on the **End node** push button that your network administrator advises you to use.
You can select either the **End node - to a network node server** radio button or the **End node - no network node server** radio button. A network node server is used when many users are routed through the same connection. This example assumes no network node server is used.
- f. Click on the **Advanced** push button. The Communications Manager Profile List window opens. Subsequent steps begin from this window. You will return to this window when each of the following steps are completed.



Step 3. Prepare a LAN DLC profile.

- a. On the Profile List window, select the **DLC - Token ring or other LAN Types Adapter Parameters** option and click on the **Configure** push button. The Token Ring or Other Lan Types Adapter Parameters window opens.
- b. Enter your Network ID (**1**) in the **Network ID** field.
- c. Click **OK**.

Step 4. Update the SNA local node characteristics.

- a. On the Profile List window, select the **SNA local node characteristics** option and click on the **Configure** push button. The Local Node Characteristic window opens.
- b. Enter your Network ID (**1**) in the **Network ID** field.
- c. The Local node name (**2**) was probably set when CS/2 was installed. If you are not sure, consult your local network administrator.

- d. Enter your Node ID (**3**) in the **Local node ID (hex)** field.

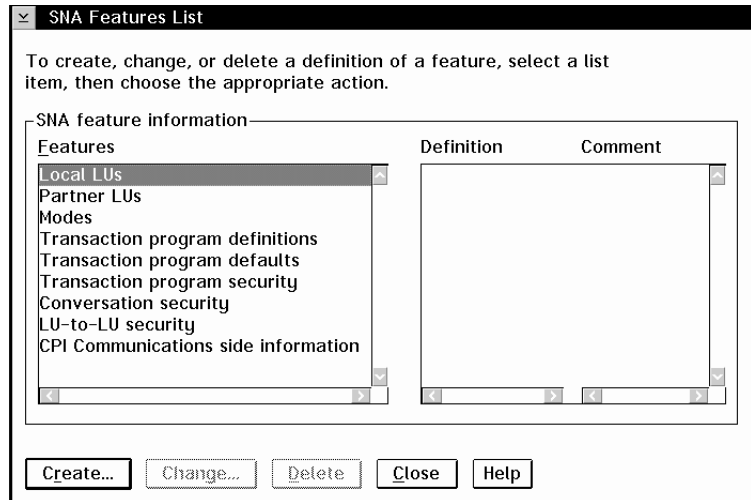


The first part should be filled in for you already when you display the profile. You only need to complete the second part.

- e. Click **OK**.

Step 5. Set SNA features.

- a. In the Profile List window, select the **SNA features** option and click on the **Configure** push button. The SNA Features List window opens. Subsequent steps begin from this window.



Step 6. Prepare a local LU profile.

If the DB2 workstation is defined as an independent LU, prepare a Local LU Profile by performing the following steps:

- a. In the SNA Features List window, select **Local LUs->Create** from the **action** menu bar.
- b. Enter your Local LU name (**4**) in the **LU name** field.
- c. Enter your Local LU alias (**5**) in the **alias** field.
- d. Select the **Independent LU** radio button in the **NAU address** box.
- e. Click **OK**.

Step 7. Prepare a mode definition.

- a. From the SNA Features List box, select the **Modes** option and click on the **Create** push button. The Mode Definition window opens.

Mode Definition

Mode name: IBMRDB

Class of service: #CONNECT

Mode session limit: 8 (0 - 32767)

Minimum contention winners: 0 (0 - 32767)

Receive pacing window: 4 (0 - 63)

Pacing type: Adaptive

Compression and session-level encryption support: Setup...

RU size

Default RU size

Maximum RU size (256 - 16384)

Optional comment

OK Cancel Help

- b. Enter your Mode name (**6**) in the **mode name** field.
- c. For the other fields, you can either specify values that match the mode profile defined on your server systems, or tune the parameters.
- d. Click **OK** to finish the creation of the mode and to return to the SNA Features List panel.

Step 8. Define a transaction program name.

- a. From the SNA Features List panel, double-click on **Transaction Program Definitions**. The Transaction Program Definition window opens.
- b. Specify your transaction program name (**8**) in the **Transaction program (TP) name** field.

- c. Enter any string, for example, notused, in the **OS/2 program path and file name** field. This field will not be used to determine the actual transaction program location, but must be filled to continue with the configuration steps.
- d. Select the **Conversation security required** check box.
- e. Click on the **Continue** push button. The Additional TP Parameters window opens.
- f. Select the **Background** radio button from the **Presentation type** group.
- g. Select the **Queued, operator preloaded** radio button from the **Operation type** group.
- h. Click **OK** to finish the TP name definition and return to the SNA Features List panel.

Step 9. Define conversation security.

- a. From the SNA Features List panel, double-click on **Security**. The Conversation Security window opens.
- b. Select the **Utilize User Profile Management** check box.
- c. Click on the **Add** push button.
- d. Click **OK** to finish the conversation security definition and return to the SNA Features List panel.

Step 10. Save the configuration.

- a. Click on the **Close** button to return to the Communication Server Configuration Definition window.
- b. Click on the **Close** button to automatically verify and save the new configuration file, and leave the configuration windows.
- c. Stop and start Communications Server. Select **Stop Communications Normally->Start Communications**.

You have now finished setting up your workstation for inbound APPC communications.

Configuring IBM eNetwork Personal Communications for Windows NT:

This section describes how to configure IBM eNetwork Personal Communications for Windows NT (PCOMM/NT) to accept inbound APPC client connections.

Before you begin, ensure that the IBM eNetwork Personal Communications software you installed:

- ___ 1. Is Version 4.30 or higher.
- ___ 2. Has the LLC2 driver installed from the IBM Communications Server installation directory. To confirm this, perform the following steps:

Step a. Click on **Start** and select **Settings->Control Panel**.

- Step b. Double-click on the Network icon. The Network window opens.
- Step c. In the Network window, click on the **Protocols** tab.
- Step d. Verify that IBM LLC2 Protocol is one of the protocols listed. If it is not, you need to install this protocol from your IBM Personal Communications for Windows NT software. Refer to its documentation for instructions.

For more information on setting up your environment, refer to the online help supplied with PCOMM/NT or to the following publications:

- *Connectivity Supplement*
- *DRDA Connectivity Guide*

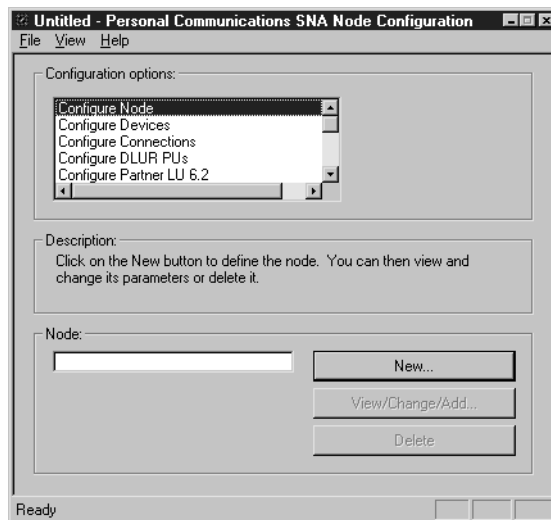
The following assumptions have been made:

- The basic installation of the PCOMM/NT package has already been completed.
- DB2 Connect or DB2 Universal Database has been installed.

Use the *Your Value* entries from the worksheet in Table 25 on page 153 to complete the steps below.

To start IBM Personal Communications, complete the following steps:

- Step 1. Click **Start** and select **Programs->IBM Communications Server->SNA Node Configuration**. The Personal Communications SNA Node Configuration window opens.



Step 2. Select **File->New** from the menu bar. The Define the Node window opens. Subsequent steps will begin from this window.

To configure APPC communications, perform the following steps:

Step 1. Configure the Node.

- a. In the **Configuration options** box, select **Configure Node**, then click on the **New** push button. The Define the Node window opens.
- b. In the **Fully qualified CP name** fields, type in your Network ID (**1**) and Local Control Point name (**2**).
- c. Optionally, in the **CP alias** field, type in a CP alias. If you leave this blank the Local Control Point name (**2**) will be used.
- d. Enter your Node ID (**3**) in the **Local Node ID** fields.
- e. Click on **OK**.

Step 2. Configure the device.

- a. In the **Configuration options** box, select **Configure devices**.
- b. Select the appropriate DLC from **DLCs** field. These instructions use the **LAN** DLC.
- c. Click on the **New** button. The appropriate window opens with default values displayed. In this case, the Define a LAN device window opens.
- d. Click **OK** to accept the default values.

Step 3. Configure Modes.

- a. In the **Configuration options** box, select **Configure modes**, then click on the **New** push button. The Define a Mode window opens.
- b. Enter your Mode name (**6**) in the **Mode name** field of the **Basic** tab.
- c. Select the **Advanced** tab.
- d. Select **#CONNECT** from the **Class of Service Name** field.
- e. Click **OK**.

Step 4. Configure Local LU 6.2.

- a. In the **Configuration options** box, select **Configure Local LU 6.2**, then click on the **New** button. The Define a Local LU 6.2 window opens.
- b. Enter your Local LU name (**4**) in the **Local LU name** field.
- c. Type in a value for the **LU session limit** field. The default, 0, specifies the maximum allowed value.
- d. Accept the defaults for the other fields and click on **OK**.

Step 5. Create a service Transaction Program.

- a. In the **Configuration options** field, select **Configure Transaction Programs**.
 - b. Click on the **New** push button. The Define a Transaction Program window opens.
 - c. Select the **Basic** tab.
 - d. Specify a service TP (**7**) in the **TP name** field.
 - e. Select the **Advanced** tab.
 - f. Change the default in the **Receive Allocate timeout** field to 0 (no timeout).
 - g. Accept the defaults for the other fields and click on **OK**.
- Step 6. Create an Application Transaction Program.
- a. In the **Configuration options** field, select **Configure Transaction Programs**, then click on the **New** button. The Define a Transaction Program window opens.
 - b. Select the **Basic** tab.
 - c. Clear the **Service TP** check box.
 - d. Specify an application TP name (**8**) in the **TP name** field.
 - e. Select the **Background Process** check box.
 - f. Select the **Advanced** tab.
 - g. Change the default in the **Receive Allocate timeout** field to 0 (no timeout).
 - h. Accept the defaults for the other fields.
 - i. Click **OK**.
- Step 7. Save the configuration.
- a. Select **File->Save As**. The Save As window opens.
 - b. Type in a file name, for example `ny3.acg`, and click **OK**.
 - c. In the window that opens, you are asked if you want this configuration to be the default. Click on the **Yes** push button.

Step 8. Update the environment.

IBM Personal Communications uses an environment variable called APPCLU to set the default Local LU used for APPC communications. You may set this variable on a per-session basis by opening a command window and entering `set appclu=local_lu_name`, where *local_lu_name* represents the name of the local LU you want to use.

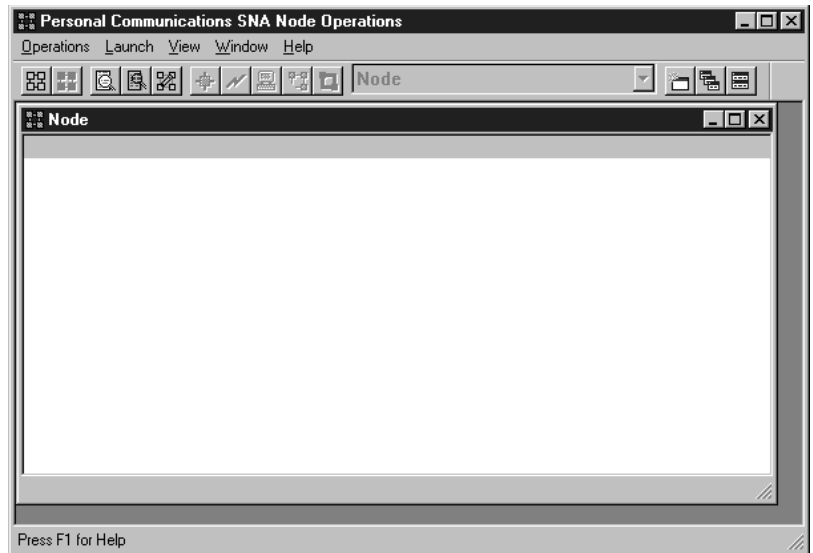
However, you will probably find it more convenient to permanently set the variable. To permanently set the variable in Windows NT, complete the following steps:

- a. Click on **Start** and select **Settings->Control Panel**.

- b. Double-click on the **System** icon. The System Properties window opens.
- c. Select the **Environment** tab.
- d. Enter `appc11u` in the **Variable** field.
- e. Enter your local LU name (**4**) in the **Value** field.
- f. Click on the **Set** push button to accept the changes.
- g. Click **OK** to exit the System Properties window. The environment variable will now remain set for future sessions.

Step 9. Start SNA Node Operations.

- a. Click **Start** and select **Programs->IBM Personal Communications->Administrative and PD Aids->SNA Node Operations**. The Personal Communications SNA Node Operations window opens.



- b. From the menu bar, select **Operations->Start Node**.
- c. In the window that opens, select the configuration file you saved in the previous step (for example, `ny3.acg`) and click **OK**.

You have now finished setting up your workstation for inbound APPC communications.

Configuring Microsoft SNA Server for Windows NT: This section describes how to configure your Microsoft SNA Server 4.0 to accept inbound APPC client connections.

If you wish to use DB2's Multisite Update feature with this product, your minimum requirement is Microsoft SNA Server Version 4 Service Pack 3. For more information about multisite updates, refer to *DB2 Connect User's Guide*.

For more information on setting up your environment, refer to the online help supplied with SNA Server, or to the following publications:

- *Connectivity Supplement*
- *DRDA Connectivity Guide*

The following assumptions have been made:

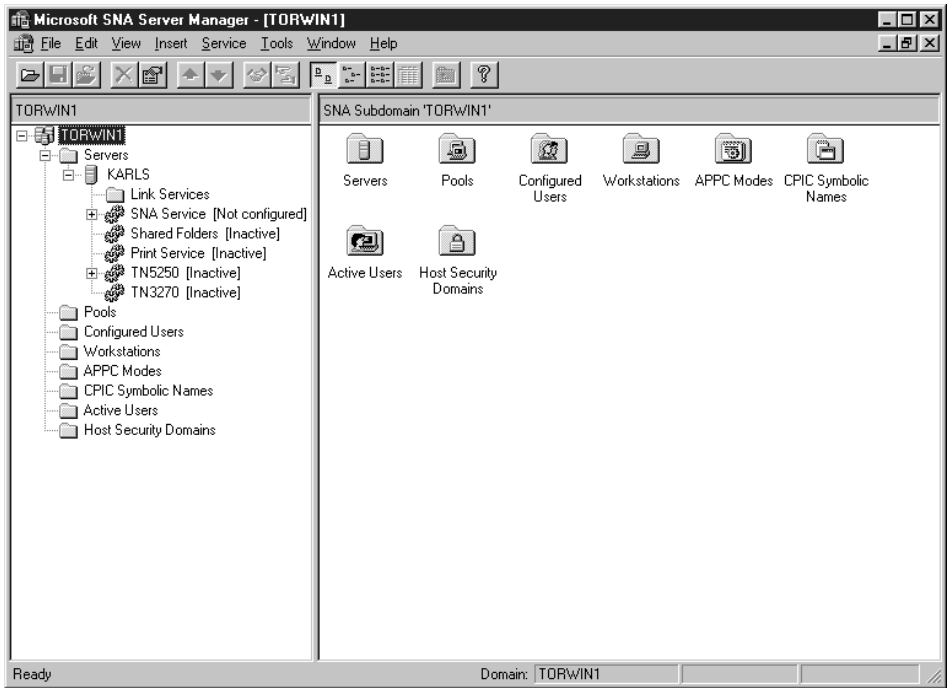
- The basic installation of the Microsoft SNA Server 4.0 for Windows NT package has already been completed.
- DB2 Connect or DB2 Universal Database has been installed.

Use the *Your Values* entries from the worksheet in Table 25 on page 153 to complete the steps below.

The transaction program (TP) name required for inbound APPC connections is automatically defined for the workstation when the SNA Server is installed. These instructions describe the remaining steps required to configure your APPC communications for inbound connections.

You can define the properties of your SNA connections in the Microsoft SNA Server Manager. The Server Manager uses an interface similar to that of the Windows NT. The following figure shows the interface. There are two panes in the main window of the Manager. All the required configuration options can be accessed by right-clicking on objects in the left-hand pane of the window. Every object has a context menu that you can access by right-clicking

on the object.

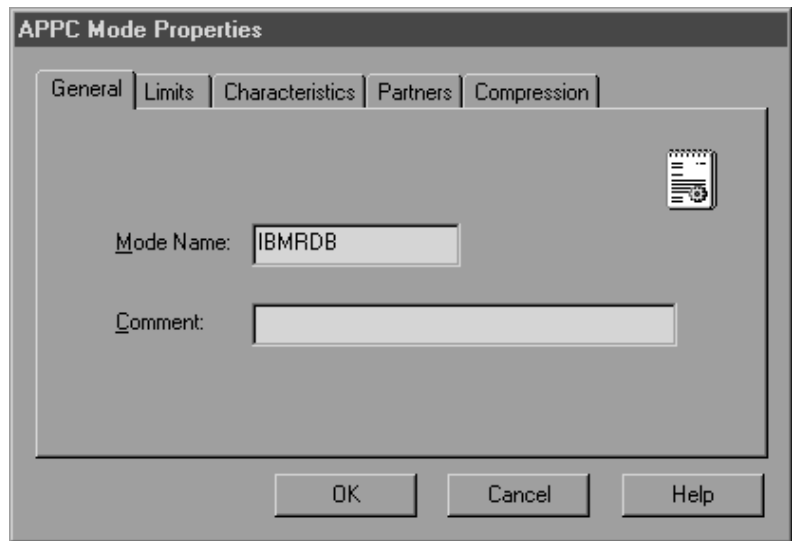


To configure APPC communications for inbound connections using Microsoft SNA Server Manager, perform the following steps:

- Step 1. Start the Server Manager by clicking on **Start** and selecting **Programs->Microsoft SNA Server->Manager**.
- Step 2. Define the control point name.
 - a. Click on the [+] sign beside the **Servers** folder.
 - b. Right-click on the **SNA Service** folder and select the **Properties** option. The Properties window opens.
 - c. Enter your Network ID (**1**) in the **NETID** field.
 - d. Enter your local Control Point name (**2**) in the **Control Point Name** field.
 - e. Click **OK**.
- Step 3. Define a local LU.
 - a. Right-click on the **SNA Service** icon and select the **Insert->APPC->Local LU** option. The Local APPC LU Properties window opens.
 - b. Enter the following information:
 - The **LU alias** (**5**).

- The **NETID**(**1**).
 - The **LU name** (**4**).
- c. Select the **Advanced** tab.
 - d. Select the **Member of Default Outgoing Local APPC LU Pool** option. Accept the other defaults.
 - e. Click **OK**.
- Step 4. Define a mode.
- a. Right-click on the **APPC Modes** folder, and select the **Insert->APPC->Mode Definition** option.

The APPC Mode Properties window opens.



- b. Enter the Mode Name **6** in the **Mode Name** field.
- c. Select the **Limits** tab.
- d. Enter appropriate numbers in the **Parallel Session Limit** and **Minimum Contention Winner Limit** fields. Your Network Administrator can supply you with the numbers if you do not know the limits you should place here.
- e. Accept the other defaults, and click **OK**.

Step 5. Save the configuration.

- a. Select **File->Save File** in the Server Manager window. The Save File window opens.
- b. Type a unique name for your configuration into the **File Name** field.
- c. Click on the **Save** button. Your configuration is now saved.

You have now finished setting up your workstation for inbound APPC communications.

Configuring SunLink SNA for Solaris: This section describes how to configure your Solaris server to accept inbound APPC client connections. Before you begin, ensure that your workstation has SunLink SNA PU 2.1 Server for Solaris installed. For more information on setting up your environment, refer to:

- *Connectivity Supplement*
- *DRDA Connectivity Guide*

- *SunLink PU 2.1 Server Configuration and Administrator's Manual.*

The following assumptions have been made:

- The basic installation of the SunLink SNA PU 2.1 Server for Solaris package has already been completed.
- DB2 Connect or DB2 Universal Database has been installed.
- The user is logged on as root.

Use the *Your Value* entries from the worksheet in Table 25 on page 153 to complete the steps below.

To configure the SunLink SNA PU 2.1 Server to accept inbound APPC connections, log on as root and edit or create the Server configuration file. This file is called `sunpu2.config`, and must be placed in `/opt/SUNWpu21`, or the directory where SunLink SNA PU 2.1 Server is installed.

The example below shows the configuration file sections required for configuring the server to accept inbound APPC client connections. Other sections that are not shown are required for establishing server to host connections.

```
// SunLink SunLU6.2/SunPU2.1 SNA Server Sample Configuration
// Token Ring Peer-to-Peer System A @(#)sunlu62.a.tr
//
// The physical connection is a Token Ring interface adapter.

CP      NAME=NYX1GW           // Local name (8 char max)
        NQ_CP_NAME=SPIFNET.NYX1GW // Network Qualified Name
        ;

TRLINE  NAME=MAC1            // SunLink specific name
        SOURCE_ADDRESS=x'400011527509' // sysA_mac_addr for Sun machine
        ;

LU      NAME=NYX1GW0A        // Local name (8 char max)
        NQ_LU_NAME=SPIFNET.NYX1GW0A // Network Qualified Name
        SESS_LMT=50           // Max LU sessions
        LUTYPE=6.2
        ;

MODE    NAME=IBMRDB          // Mode Name (8 char max)
        DLC_NAME=NYX2        // Associated DLC
        PTNR_LU_NAME=NYX2    // Associated Local LU
        LCL_MAX_SESS_LMT=30 // Max Session Limit
        MIN_CW_SESS=15       // Min Conwinners
        MIN_CL_SESS=15       // Min Conlosers
        ;
// This section adds DLC for the inbound APPC client NYX2

DLC     NAME=NYX2,           // User defined name (8 char max)
        LINK_NAME=MAC1,     // Line name this station is on
```

```

LCLLSAP=x'04',           // Local Link Service Access Point
RMTLSAP=x'04',           // Remove Link Service Access Point
RMTMACADDR=x'400011528901', // sysB_mac_addr
TERMINID=x'05d27510',   // IDNUM and IDBLK = XID
MAXDATA=4096,
ACTIVITY_TIMEOUT=0,
RETRIES=20,
REPLY_TIMEOUT=20,
RESPONSE_TIMEOUT=20,
ACTPU_SUPPRESS=yes
;

// This section defines the partner LU NYX2

PTNR_LU NAME=NYX2,           // Partner LU name (8 char max)
        LOC_LU_NAME=NYX1GW0A, // Associated Local LU
        NQ_LU_NAME=SPIFNET.NYX2 // Network Qualified Name
        SEC_ACCEPT=ALREADY_VERIFIED // Accept client as already verified
;

// This section adds the TP name NYSERVER
// and associates it with the local LU NYX1GW01

TP      TP_NAME=DB2DRDA,           // TP Name
        LOC_LU_NAME=NYX1GW0A,     // Associated Local LU
        CONV_TYPE=BASIC,         // Conversation Type
;

//

SECURITY LOC_LU_NAME=NYX1GW0A,     // Local LU Alias
        USER_ID=USERID,          // User id
        PASSWORD=PASSWORD,       // Password (since UNIX_SEC=NO)
;

```

When you have finished editing and saving the server configuration file, perform the following steps:

Step 1. Start and stop the SunLINK subsystem

- a. Change to the SunLink directory, which is usually /opt/SUNWpu21.
- b. Set up environment variables for FlexLM licensing. For example:

```

export LD_LIBRARY_PATH=/usr/openwin/lib:/usr/lib
export LM_LICENSE_FILE=/etc/opt/licenses/licenses_combined

```

For more information, refer to the SunLink documentation.

- c. Ensure you have created the SNA server configuration file in the /opt/SUNWpu21 directory.
- d. Use the sunop utility to check the status of SunLink SNA if it is already started.

Check to see if the PU and/or DLC status is connected. Refer to the SunLink documentation for details of the sunop utility.

e. Stop SunLink if it is active. For example, enter the following command:

```
kill -9 sunpu2.pid
```

f. Start SunLink. For example, enter the following command:

```
sunpu2.1
```

Step 2. Set the following environment variables:

APPC_GATEWAY

Name of the DB2 for Solaris server (usually the TCP/IP hostname).

APPC_LOCAL_LU

Name of the local LU name provided in the SNA configuration file (**4**).

Export these on the server machine.

You have now finished setting up your workstation for inbound APPC communications.

Part 3. CLI/ODBC Enablement

Chapter 12. Running Your Own Applications

Various types of applications can access DB2 databases:

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

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

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

Step 1. Ensure the server is configured and running.

Ensure that the database manager is started on the database server to which the application program is connecting. If it is not, you must issue the **db2start** command at the server before starting the application.

Step 2. Ensure that you can connect to the database that the application uses.

Step 3. Bind the utilities and the applications to the database. See “Binding Database Utilities” for more information.

Step 4. Run the application program.

Binding Database Utilities

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

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

The bind files are grouped together in different .lst files in the bnd directory, under the installation directory (typically sql11ib for OS/2, and Windows 32-bit operating systems). Each file is specific to a server.

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

- On OS/2 and Windows 32-bit operating systems, you can use the Client Configuration Assistant by performing the following steps:
 - Step 1. Start the Client Configuration Assistant (CCA).
 - Step 2. Select the database to which you want to bind the utilities.
 - Step 3. Click **Bind**.
 - Step 4. Select the **Bind DB2 Utilities** radio button.
 - Step 5. Click **Continue**.
 - Step 6. Enter a user ID and password to connect to the database. The user ID must have the authority to bind new packages against the database. Select the utilities you want to bind and click **OK**.
- On all operating systems, you can use the command line processor by performing the following steps:

Step 1. Change to the bnd directory in the install path. For example:

For UNIX platforms

INSTHOME/sql11ib/bnd, where INSTHOME represents the home directory of the instance that you want to use.

For all other platforms

x:\sql11ib\bnd, where x: represents the drive where you installed DB2

Step 2. To connect to the database, enter the following commands in the Command Center or the Command Line Processor:

```
connect to database_alias
```

where *database_alias* represents the name of the database to which you want to connect.

Step 3. Enter the following commands in the Command Center or the Command Line Processor:

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

In this example, bind.msg and clibind.msg are the output message files, and EXECUTE and BINDADD privileges are granted to *public*.

Step 4. Reset the connection to the database by entering the following command:

```
connect reset
```


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

Notes:

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



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

Running CLI/ODBC Programs

The DB2 Call Level Interface (CLI) run-time environment and the DB2 CLI/ODBC driver are included with DB2 clients as optional components during install.

This support enables applications developed using ODBC and DB2 CLI APIs to work with any DB2 server. DB2 CLI application development support is provided by the DB2 Application Development Client which is packaged with your DB2 server.

Before DB2 CLI or ODBC applications can access DB2, the DB2 CLI packages must be bound on the server. Although this will occur automatically on the first connection if the user has the required authority to bind the packages, it is recommended that the administrator do this first with each version of the client on each platform that will access the server. See "Binding Database Utilities" on page 185 for specific details.

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

- Step 1.** Use the Client Configuration Assistant (CCA) to add the database (if you have separate client and server machines) so that its instances and databases can be made known to the Control Center, then add

the instances and databases for that system. If you do not have access to this program you can use the **catalog** command in the command line processor.

- Step 2. The DB2 CLI/ODBC driver is an optional component during the DB2 client install on Windows platforms. Be sure it is selected at that point. On OS/2 you must use the **Install ODBC Driver** icon to install both the DB2 CLI/ODBC driver and the ODBC driver manager. On UNIX platforms the DB2 CLI/ODBC driver is automatically installed with the client.
- Step 3. To access the DB2 database from ODBC:
- a. The ODBC Driver Manager (From Microsoft or other vendor) must already be installed (this is done by default during the installation of DB2 only on 32-bit Windows systems).
 - b. The DB2 databases must be registered as ODBC data sources. The ODBC driver manager does not read the DB2 catalog information; instead it references its own list of data sources.
 - c. If a DB2 table does not have a unique index then many ODBC applications will open it as read-only. A unique index should be created for each DB2 table that is to be updated by an ODBC application. Refer to the **CREATE INDEX** statement in the *SQL Reference*. Using the Control Center you would alter the settings of the table, then select the **Primary Key** tab and move one or more columns from the available columns list over to the primary key columns list. Any column you select as part of the primary key must be defined as NOT NULL.
- Step 4. If necessary, you can set various CLI/ODBC Configuration Keywords to modify the behavior of DB2 CLI/ODBC and the applications using it.

If you followed the above steps to install ODBC support, and added DB2 databases as ODBC data sources, your ODBC applications will now be able to access them.

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

- “How to Bind the DB2 CLI/ODBC Driver to the Database” on page 195
- “How to Set CLI/ODBC Configuration Keywords” on page 195
- “Configuring db2cli.ini” on page 196

Platform Specific Details for CLI/ODBC Access



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

- “Windows 32-bit operating systems Client Access to DB2 using CLI/ODBC”
 - “OS/2 Client Access to DB2 using CLI/ODBC” on page 191
 - “UNIX Client Access to DB2 using CLI/ODBC” on page 193
-

Windows 32-bit operating systems Client Access to DB2 using CLI/ODBC

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

Step 1. The DB2 database (and node if the database is remote) must be cataloged. To do so, use the CCA (or the command line processor). For more information refer to the online help in the CCA (or the **CATALOG DATABASE** and **CATALOG NODE** commands in the *Command Reference*).

Step 2. Verify that the Microsoft ODBC Driver Manager and the DB2 CLI/ODBC driver are installed. On Windows 32-bit operating systems they are both installed with DB2 unless the ODBC component is manually unselected during the install. DB2 will not overwrite a newer version of the Microsoft ODBC Driver Manager if one is found.

To verify that they both exist on the machine:

- a. Start the Microsoft ODBC Data Sources icon in the Control Panel, or run the **odbcad32.exe** command from the command line.
- b. Click on the **Drivers** tab.
- c. Verify that “IBM DB2 ODBC DRIVER” is shown in the list.

If either the Microsoft ODBC Driver Manager or the IBM DB2 CLI/ODBC driver is not installed, then rerun the DB2 install and select the ODBC component on Windows 32-bit operating systems.

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

- Using the CCA:
 - a. Select the DB2 database alias that you want to add as a data source.
 - b. Click on the **Properties** push button. The Database Properties window opens.

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

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

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

Step 4. Configure the DB2 CLI/ODBC driver using the CCA: (Optional)

- a. Select the DB2 database alias you want to configure.
- b. Click on the **Properties** push button. The Database Properties window opens.
- c. Click on the **Settings** push button. The CLI/ODBC Settings window opens.
- d. Click on the **Advanced** push button. You can set the configuration keywords in the window that opens. These keywords are associated with the database *alias name*, and affect all DB2 CLI/ODBC applications that access the database. The online help explains all of the keywords, as does “Configuration Keyword Descriptions” on page 205.

For information on manually editing this file (`db2cli.ini`), see “Configuring `db2cli.ini`” on page 196.

Step 5. If you have installed ODBC access (as described above), you can now access DB2 data using ODBC applications. Start the ODBC application and go to the Open window. Select the **ODBC databases** file type. The DB2 databases that you added as ODBC data sources

will be selectable from the list. Many ODBC applications will open the table as read-only unless a unique index exists.



If you require additional information at this point you can refer to the following topics in “Detailed Configuration Information” on page 195:

- “How to Bind the DB2 CLI/ODBC Driver to the Database” on page 195
 - “How to Set CLI/ODBC Configuration Keywords” on page 195
 - “Configuring db2cli.ini” on page 196
-

OS/2 Client Access to DB2 using CLI/ODBC

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

1. The DB2 database (and node if the database is remote) must be cataloged. To do so, use the CCA (or the command line processor).
For more information, see the online help in the CCA. (or the **CATALOG DATABASE** and **CATALOG NODE** commands in the *Command Reference*).
2. If you are using ODBC applications to access DB2 data, perform the following steps. (If you are using only CLI applications, skip this step and go to the next step.)

- a. Check that there is an ODBC Driver Manager installed. The ODBC Driver Manager is not installed with DB2; we suggest you use the Driver Manager that was shipped with your ODBC application. Also ensure that the DB2 CLI/ODBC driver is installed:
 - 1) Run the ODBC Administration tool as described in its documentation. This is usually done in one of two ways:
 - Double-click on the **ODBC** Folder in OS/2, and double-click on the **ODBC Administrator** icon.
 - Run **odbcadm.exe** from the command line.

The Data Sources window opens.

- 2) Click on the **Drivers** push button. The Drivers window opens.
- 3) Verify that “IBM DB2 ODBC DRIVER” is shown in the list.

If the ODBC Driver Manager is not installed then follow the installation instructions that came with your ODBC application. If the IBM DB2 CLI/ODBC driver is not installed then double-click on the **Install ODBC Driver** icon in the DB2 folder to install the DB2 CLI/ODBC driver.

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

- 1) Select the DB2 database alias that you want to add as a data source.
- 2) Click on the **Properties** push button.
- 3) Select the **Register this database for ODBC** check box.
- Using the ODBC Driver Manager:
 - 1) Run the ODBC Driver Manager, as described in its documentation. This is usually done in one of two ways:
 - Double-click on the **ODBC** Folder in OS/2, and double-click on the **ODBC Administrator** icon.
 - Run **odbcadm.exe** from the command line.
 - 2) Click on the **Add** push button from the Data Sources window. The Add Data Source Window opens.
 - 3) Double-click on the IBM DB2 ODBC DRIVER in the list.
 - 4) Select the DB2 database to add and click on **OK**.
3. Configure the DB2 CLI/ODBC driver using the CCA: (Optional)
 - a. Select the DB2 database alias you want to configure.
 - b. Click on the **Properties** push button. The Database Properties window opens.
 - c. Click on the **Settings** push button. The CLI/ODBC Settings window opens.
 - d. Click on the **Advanced** push button. You can set the configuration keywords in the window that appears. These keywords are associated with the database *alias name*, and affect all DB2 CLI/ODBC applications that access the database. The online help explains all of the keywords, as does “Configuration Keyword Descriptions” on page 205.
For information on manually editing this file (db2cli.ini), see “Configuring db2cli.ini” on page 196.
4. If you have installed ODBC access (as described above), you can now access DB2 data using ODBC applications. Start the ODBC application and go to the Open window. Select the **ODBC databases** file type. The DB2 databases that you added as ODBC data sources will be selectable from the list. Many ODBC applications will open the table as read-only unless a unique index exists.



If you require additional information at this point you can refer to the following topics in “Detailed Configuration Information” on page 195:

- “How to Bind the DB2 CLI/ODBC Driver to the Database” on page 195
 - “How to Set CLI/ODBC Configuration Keywords” on page 195
 - “Configuring db2cli.ini” on page 196
-

UNIX Client Access to DB2 using CLI/ODBC

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

1. The DB2 database (and node if the database is remote) must be cataloged. To do so, use the command line processor.
For more information see “Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor” on page 41 or the **CATALOG DATABASE** and **CATALOG NODE** commands in the *Command Reference*.
2. The DB2 CLI/ODBC driver is installed during the DB2 client install. Be sure it is selected at that point.
3. If you are using ODBC applications to access DB2 data, perform the following steps. (If you are using only CLI applications, skip this step and go to the next step.)

- a. When using an ODBC application you must ensure that an ODBC Driver Manager is installed and that each user that will use ODBC has access to it. DB2 does not install an ODBC Driver Manager, you must use the ODBC Driver Manager that was supplied with your ODBC client application or ODBC SDK in order to access DB2 data using that application.
- b. The Driver Manager uses two initialization files.

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

.odbc.ini End-user’s data source configuration. Each user ID has a separate copy of this file in their home directory. Note that the file starts with a dot.

Setting up odbcinst.ini

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

Use an ASCII editor to update this file. It must have a stanza (section) called [IBM DB2 ODBC DRIVER], with a line starting with “Driver” indicating the full path to the DB2 ODBC driver, called db2.o on AIX, and libdb2 on other UNIX platforms (the file extension depends on the platform; libdb2.so on Solaris Operating Environment, and so on). For example, on AIX, if the home directory of your end user is /u/thisuser/ and the sqllib directory is installed there, then the correct entry would be:

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

Setting up .odbc.ini

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

The .odbc.ini file must exist in the end user's home directory (note the dot at the start of the file name). Update this file, using an ASCII editor, to reflect the appropriate data source configuration information. To register a DB2 database as an ODBC data source there must be one stanza (section) for each DB2 database.

The .odbc.ini file must contain the following lines:

- in the [ODBC Data Source] stanza:

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

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

- in the [SAMPLE] stanza (on AIX, for example):

```
[SAMPLE]
Driver=/u/thisuser/sql1lib/lib/libdb2.a
Description=Sample DB2 ODBC Database
```

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

- in the [ODBC] stanza:

```
InstallDir=/u/thisuser/sql1lib/odbc1ib
```

Indicates that /u/thisuser/sql1lib/odbc1ib should be treated as the location where ODBC is installed.

- Ensure that the InstallDir correctly points to the ODBC Driver Manager location.

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

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

See "How to Configure ODBC.INI" on page 196 for more detailed information.

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

4. Configure the DB2 CLI/ODBC driver (Optional).

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

For information on manually editing this file (`db2cli.ini`), see “Configuring `db2cli.ini`” on page 196. For information about the specific keywords see the *CLI Guide and Reference*.



If you require additional information at this point you can refer to the following topics in “Detailed Configuration Information”:

- “How to Bind the DB2 CLI/ODBC Driver to the Database”
 - “How to Set CLI/ODBC Configuration Keywords”
 - “Configuring `db2cli.ini`” on page 196
-

Detailed Configuration Information

The section “Platform Specific Details for CLI/ODBC Access” on page 189 should provide you with all of the information you require. The following additional information is useful where DB2 tool support is not available, and for administrators who require more detailed information.

The following topics are covered in this section:

- “How to Bind the DB2 CLI/ODBC Driver to the Database”
- “How to Set CLI/ODBC Configuration Keywords”
- “Configuring `db2cli.ini`” on page 196

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

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

See “Binding Database Utilities” on page 185 for more information.

How to Set CLI/ODBC Configuration Keywords

DB2 CLI can be configured further by using either the CCA or the DB2 Client Setup administration tool, whichever is applicable for your platform, or by manually editing the `db2cli.ini` file.

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

The following is an example `odbc.ini` file:

```
[ODBC Data Sources]
MS Access Databases=Access Data (*.mdb)

[MS Access Databases]
Driver=D:\WINDOWS\SYSTEM\simba.dll
FileType=RedISAM
SingleUser=False
UseSystemDB=False
```

The [ODBC Data Sources] section lists the name of each available data source and the description of the associated driver.

For each data source listed in the [ODBC Data Sources] section, there is a section that lists additional information about that data source. These are called the *Data Source Specification* sections.

Step 2. Under the [ODBC DATA SOURCE] entry, add the following line:

```
database_alias=IBM DB2 ODBC DRIVER
```

where *database_alias* is the alias of the database cataloged in the database directory (the database name used by the command line processor `CONNECT TO` statement).

Step 3. Add a new entry in the Data Source Specification section to associate the data source with the driver:

```
[database_alias]
Driver=x:\windows\system\db2cliw.dll
```

where:

- *database_alias* is the alias of the database cataloged in the database directory, and listed under the Data Source Specification section.
- *x*: is the drive where the Windows operating system is installed.

The following shows the example file with the IBM data source entries added:

```
[ODBC Data Sources]
MS Access Databases=Access Data (*.mdb)
SAMPLE=IBM DB2 ODBC DRIVER

[MS Access Databases]
Driver=D:\WINDOWS\SYSTEM\simba.dll
FileType=RedISAM
SingleUser=False
UseSystemDB=False

[SAMPLE]
Driver=D:\WINDOWS\SYSTEM\db2cliw.dll
Description=Sample DB2 Client/Server database
```

UNIX Configuration of .ini files

The section “UNIX Client Access to DB2 using CLI/ODBC” on page 193 contains detailed steps on how to update both the `odbc.ini` and `odbcinst.ini` files.

Running Java Programs

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

For DB2 JDBC support, you must include the DB2 Java Enablement component when you install the DB2 client. With DB2 JDBC support you can build and run JDBC applications and applets. These contain dynamic SQL only, and use a Java call interface to pass SQL statements to DB2.

The DB2 Application Development Client provides support for Java embedded SQL (SQLJ). With DB2 SQLJ support and DB2 JDBC support you can build and run SQLJ applications and applets. These contain static SQL and use embedded SQL statements that are bound to the DB2 database.

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

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

- To build JDBC applications, you must install a DB2 client with the DB2 Java Enablement component. To run JDBC applications, your DB2 client with the DB2 Java Enablement component must connect to a DB2 server.
- To build SQLJ applications, you must install the DB2 Application Development Client and a DB2 Administrative Client with the DB2 Java Enablement component. To run SQLJ applications, your DB2 client with the DB2 Java Enablement component must connect to a DB2 server.
- To build JDBC applets, you must install a DB2 client with the DB2 Java Enablement component. To run JDBC applets, the client machine does not require any DB2 components.
- To build SQLJ applets, you must install the DB2 Application Development Client and a DB2 Administrative Client with the DB2 Java Enablement component. To run SQLJ applets, the client machine does not require any DB2 components.

For detailed information on building and running JDBC and SQLJ programs refer to the *Application Building Guide*. For more information on DB2 programming in Java, refer to the *Application Development Guide*. This covers creating and running JDBC and SQLJ applications, applets, stored procedures and UDFs.

For the latest, updated DB2 Java information, visit the Web site at:
<http://www.ibm.com/software/data/db2/java>

Configuring the Environment

To build DB2 Java programs, you need to install and configure the appropriate version of a Java Development Kit (JDK) on your development machine. To run DB2 Java applications, you must install and configure the appropriate version of either a Java Runtime Environment (JRE) or JDK on your development machine. The following table lists the version of the JDK that is appropriate for your development machine:

AIX The IBM AIX Developer Kit, Java Technology Edition, Version 1.1.8. On AIX systems that do not have a JDK installed, this JDK is automatically installed with the DB2 Application Development Client.

HP-UX

The HP-UX Developer's Kit for Java, Release 1.1.8, from Hewlett-Packard.

Linux The IBM Developer Kit for Linux, Java Technology Edition, Version 1.1.8.

OS/2 The IBM Java Development Kit for OS/2, version 1.1.8, which is available on the product CD-ROM.

PTX The ptx/JSE, Version 1.2.1, from IBM.

SGI IRIX

The Java 2 Software Development Kit for SGI IRIX, version 1.2.1, from SGI.

Solaris Operating Environment

The Java Development Kit for Solaris, version 1.1.8, from Sun Microsystems.

Windows 32-bit operating systems

The IBM Developer Kit for Windows 32-bit operating systems, Java Technology Edition, Version 1.1.8. When you install the DB2 Application Development Client, this JDK is automatically installed in the `sql1lib\java\jdk` directory.

For information on installing and configuring any of the above JDKs, please refer to: <http://www.ibm.com/software/data/db2/java>

For all supported platforms, you must also install and configure a DB2 client with the DB2 Java Enablement component. To bind SQLJ programs to a database, you must install and configure a DB2 Administrative Client with the DB2 Java Enablement component.

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

On UNIX platforms:

```
db2 update dbm cfg using JDK11_PATH /usr/jdk
```

where /usr/jdk represents the path where the JDK is installed.

On Windows and OS/2 platforms:

```
db2 update dbm cfg using JDK11_PATH C:\sql11ib\java\jdk
```

where C:\sql11ib\java\jdk represents the path where the JDK is installed.

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

```
db2 get dbm cfg
```

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



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

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

For more information on this and other workarounds available, please visit: <http://www.ibm.com/software/data/db2/java>

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

On UNIX platforms:

- CLASSPATH includes "." and the file sqllib/java/db2java.zip
- On AIX, Linux, PTX, Silicon Graphics IRIX, and Solaris Operating Environment: LD_LIBRARY_PATH includes the directory sqllib/lib
- On HP-UX: SHLIB_PATH includes the directory sqllib/lib
- On Solaris Operating Environment only: THREADS_FLAG is set to "native"

On Windows and OS/2 platforms:

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

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

On UNIX platforms:

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

On Windows and OS/2 platforms:

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

Java Applications

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

```
java prog_name
```

where prog_name is the name of the program.

The DB2 JDBC driver handles the JDBC API calls from your application and uses the DB2 client to communicate the requests to the server and receive the results. An SQLJ application must be bound to the database before it is run.

Java Applets

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

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

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

then you would enter:

```
db2jstrt 6789
```

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

Copy the `sqllib/java/db2java.zip` file into the same directory as these other files. For SQLJ applets, also copy the `sqllib/java/runtime.zip` file into this directory. Then on your client machine start your web browser (which supports JDK 1.1) and load the `.html` file.

When your applet calls the JDBC API to connect to DB2, the JDBC driver establishes separate communications with the DB2 database through the JDBC applet server residing on the DB2 server. An SQLJ applet must be bound to the database before it is run.

Chapter 13. DB2 CLI/ODBC Configuration Keyword Listing

The keywords are listed in alphabetical order starting with "APPENDAPINAME." They are also divided into categories. In the ODBC Data Source Configuration tool (not available on UNIX platforms) each of these categories is presented on a separate tab in the notebook.

For further information about DB2 CLI/ODBC applications, see the information on your specific operating system in "Platform Specific Details for CLI/ODBC Access" on page 189.

Configuration Keywords by Category

CLI/ODBC Settings General Configuration Keywords

- "DBALIAS" on page 219
- "PWD" on page 236
- "UID" on page 250

Compatibility Configuration Keywords

The **Compatibility** set of options are used to define DB2 behavior. They can be set to ensure that other applications are compatible with DB2.

- "DEFERREDPREPARE" on page 221
- "DISABLEMULTITHREAD" on page 222
- "EARLYCLOSE" on page 222

Data Type Configuration Keywords

The **Data Type** set of options are used to define how DB2 reports and handles various data types.

- "BITDATA" on page 206
- "GRAPHIC" on page 225
- "LOBMAXCOLUMNSIZE" on page 229
- "LONGDATACOMPAT" on page 229

Enterprise Configuration Keywords

The **Enterprise** set of options are used to maximize the efficiency of connections to large databases.

- "CLISHEMA" on page 208
- "CONNECTNODE" on page 209
- "CURRENTPACKAGESET" on page 211
- "CURRENTSCHEMA" on page 212
- "CURRENTSQLID" on page 213
- "DB2CONNECTVERSION" on page 215
- "DBNAME" on page 219

- “GRANTEELIST” on page 223
- “GRANTORLIST” on page 224
- “SCHEMALIST” on page 237
- “SYSSHEMA” on page 243
- “TABLETYPE” on page 244

Environment Configuration Keywords

The **Environment** set of options are used to define the location of various files on the server and client machines.

- “CLIPKG” on page 207
- “CURRENTFUNCTIONPATH” on page 210
- “DEFAULTPROCLIBRARY” on page 220
- “QUERYTIMEOUTINTERVAL” on page 236
- “TEMPDIR” on page 245

File DSN Configuration Keywords

The **File DSN** set of options are used to set the TCP/IP settings for a file DSN connection.

- “DATABASE” on page 214
- “HOSTNAME” on page 226
- “PROTOCOL” on page 235
- “SERVICENAME” on page 238

Optimization Configuration Keywords

The **Optimization** set of options are used to speed up and reduce the amount of network flow between the CLI/ODBC Driver and the server.

- “CURRENTREFRESHAGE” on page 212
- “DB2DEGREE” on page 215
- “DB2ESTIMATE” on page 216
- “DB2EXPLAIN” on page 217
- “DB2OPTIMIZATION” on page 218
- “KEEPSTATEMENT” on page 228
- “OPTIMIZEFORNROWS” on page 232
- “OPTIMIZESQLCOLUMNS” on page 233
- “UNDERSCORE” on page 251

Service Configuration Keywords

The **Service** set of options are used to help in troubleshooting problems with CLI/ODBC connections. Some options can also be used by programmers to gain a better understanding of how their CLI programs are translated into calls to the server.

- “APPENDAPINAME” on page 205
- “IGNOREWARNINGS” on page 226
- “IGNOREWARNLIST” on page 227
- “PATCH1” on page 233
- “PATCH2” on page 234
- “POPUPMESSAGE” on page 235

- “SQLSTATEFILTER” on page 239
- “TRACE” on page 246
- “TRACECOMM” on page 246
- “TRACEFILENAME” on page 247
- “TRACEFLUSH” on page 248
- “TRACEPATHNAME” on page 249
- “WARNINGLIST” on page 252

Static SQL Configuration Keywords

The **Static SQL** set of options are used when running static SQL statements in CLI/ODBC applications.

- “STATICCAPFILE” on page 239
- “STATICLOGFILE” on page 240
- “STATICMODE” on page 240
- “STATICPACKAGE” on page 241

Transaction Configuration Keywords

The **Transaction** set of options are used to control and speed up SQL statements used in the application.

- “ASYNCENABLE” on page 206
- “CONNECTTYPE” on page 210
- “CURSORHOLD” on page 213
- “KEEPCONNECT” on page 227
- “MAXCONN” on page 230
- “MODE” on page 231
- “MULTICONNECT” on page 231
- “SYNCPOINT” on page 242
- “TXNISOLATION” on page 250

Configuration Keyword Descriptions

APPENDAPINAME

Keyword Description:

Append the CLI/ODBC function name which generated an error to the error message.

db2cli.ini Keyword Syntax:

APPENDAPINAME = 0 | 1

Default Setting:

Do NOT display DB2 CLI function name.

DB2 CLI/ODBC Settings Tab:

Service

Usage Notes:

The DB2 CLI function (API) name that generated an error is appended to the error message retrieved using `SQLGetDiagRec()` or `SQLError()`. The function name is enclosed in curly braces { }.

For example,

```
[IBM][CLI Driver]" CLIxxxx: < text >  
SQLSTATE=XXXXX {SQLGetData}"  
    0 = do NOT append DB2 CLI function name (default)  
    1 = append the DB2 CLI function name
```

This keyword is only useful for debugging.

ASYNCEENABLE

Keyword Description:

Enable or disable the ability to execute queries asynchronously.

db2cli.ini Keyword Syntax:

ASYNCEENABLE = 1 | 0

Default Setting:

Execute queries asynchronously.

DB2 CLI/ODBC Settings Tab:

Transaction

Usage Notes:

This option allows you to enable or disable the ability to execute queries asynchronously. This only benefits applications that were written to take advantage of this feature. Disable it only if your application does not function properly when enabled. It is placed in the data source specific section of the `db2cli.ini` file.

1 = Execute queries asynchronously (default)

0 = Queries not executed asynchronously

Note: The CLI/ODBC driver will act as it did with previous versions of DB2 that did not support asynchronous ODBC.

BITDATA

Keyword Description:

Specify whether binary data types are reported as binary or character data types.

db2cli.ini Keyword Syntax:

BITDATA = 1 | 0

Default Setting:

Report FOR BIT DATA and BLOB data types as binary data types.

DB2 CLI/ODBC Settings Tab:

Data Type

Usage Notes:

This option allows you to specify whether ODBC binary data types (SQL_BINARY, SQL_VARBINARY, SQL_LONGVARBINARY, and SQL_BLOB), are reported as binary type data. IBM DBMSs support columns with binary data types by defining CHAR, VARCHAR, and LONG VARCHAR columns with the FOR BIT DATA attribute. DB2 Universal Database will also support binary data via the BLOB data type (in this case it is mapped to a CLOB data type).

Users may also need to set this option if they are using a DB2 Version 1 application that retrieves (LONG) (VAR)CHAR data into SQL_C_CHAR buffer. In DB2 Version 1, data is moved into the SQL_C_CHAR buffer unchanged; starting in DB2 Version 2, the data is converted into the ASCII representation of each hexadecimal nibble.

Only set BITDATA = 0 if you are sure that all columns defined as FOR BIT DATA or BLOB contain only character data, and the application is incapable of displaying binary data columns.

1 = report FOR BIT DATA and BLOB data types as binary data types (default).

0 = report FOR BIT DATA and BLOB data types as character data types.

CLIPKG**Keyword Description:**

Number of large packages to be generated

db2cli.ini Keyword Syntax:

CLIPKG = 3 | 4 | ... | 30

Default Setting:

3

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Usage Notes:

If the value is NOT an integer between 3 and 30, the default will be used without error or warning.

This keyword is used to increase the number of sections for SQL statements in CLI/ODBC applications. If it is used, the administrator should explicitly bind the required CLI bind files with CLIPKG bind option. Also, the `db2cli.ini` file on the server (DB2 UDB V6.1 or later on UNIX or Intel platforms) must be updated with the same value of CLIPKG.

This setting only applies to large packages (containing 364 sections). The number of small packages (containing 64 sections) is 3 and cannot be changed.

It is recommended that you only increase the number of sections enough to run your application as the packages take up space in the database.

CLISCHEMA

Keyword Description:

Set the DB2 ODBC catalog view to use.

db2cli.ini Keyword Syntax:

CLISCHEMA = ODBC catalog view

Default Setting:

None - No ODBC catalog view is used

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The `db2cli.ini` file must be modified directly to make use of this keyword.

See Also:

“SYSSHEMA” on page 243

Usage Notes:

The DB2 ODBC catalog is designed to improve the performance of schema calls for lists of tables in ODBC applications that connect to host DBMSs through DB2 Connect.

The DB2 ODBC catalog, created and maintained on the host DBMS, contains rows representing objects defined in the real DB2 catalog, but these rows include only the columns necessary to support ODBC operations. The tables in the DB2 ODBC catalog are pre-joined and specifically indexed to support fast catalog access for ODBC applications.

System administrators can create multiple DB2 ODBC catalog views, each containing only the rows that are needed by a particular user group. Each end user can then select the DB2 ODBC catalog view they wish to use (by setting this keyword).

Use of the CLISHEMA setting is completely transparent to the ODBC application; you can use this option with any ODBC application.

While this keyword has some similar effects as the SYSSHEMA keyword, CLISHEMA should be used instead (where applicable).

CLISHEMA improves data access efficiency: The user-defined tables used with SYSSHEMA were mirror images of the DB2 catalog tables, and the ODBC driver still had to join rows from multiple tables to produce the information required by the ODBC user. Using CLISHEMA also results in less contention on the catalog tables.

CONNECTNODE

Keyword Description:

Specify the node to which a connect is to be made

db2cli.ini Keyword Syntax:

CONNECTNODE = integer value from 1 to 999 |
SQL_CONN_CATALOG_NODE

Default Setting:

Logical node which is defined with port 0 on the machine is used.

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Only Applicable when:

Connecting to a multi-node DB2 Extended Enterprise Edition database server.

Usage Notes:

Used to specify the target logical node of a DB2 Extended Enterprise Edition database partition server that you want to connect to. This keyword (or attribute setting) overrides the value of the environment variable DB2NODE. Can be set to:

- an integer between 0 and 999
- SQL_CONN_CATALOG_NODE

If this variable is not set, the target logical node defaults to the logical node which is defined with port 0 on the machine.

CONNECTTYPE

Keyword Description:

Remote or Distributed unit of work.

db2cli.ini Keyword Syntax:

CONNECTTYPE = 1 | 2

Default Setting:

Remote unit of work

DB2 CLI/ODBC Settings Tab:

Transaction

See Also:

“SYNCPOINT” on page 242

Usage Notes:

This option allows you to specify the default connect type.

1 = Remote unit of work. Multiple concurrent connections, each with its own commit scope. The concurrent transactions are not coordinated. (default)

2 = Distributed unit of work. Coordinated connections where multiple databases participate under the same distributed unit of work. This setting works in conjunction with the SYNCPOINT setting to determine if a Transaction Manager should be used.

CURRENTFUNCTIONPATH

Keyword Description:

Specify the schema used to resolve function references and data type references in dynamic SQL statements.

db2cli.ini Keyword Syntax:

CURRENTFUNCTIONPATH = *current_function_path*

Default Setting:

See description below.

DB2 CLI/ODBC Settings Tab:

Environment

Usage Notes:

This keyword defines the path used to resolve function references and data type references that are used in dynamic SQL statements. It contains a list of one or more schema-names, where schema-names are enclosed in double quotes and separated by commas.

The default value is "SYSIBM","SYSFUN",X where X is the value of the USER special register delimited by double quotes. The schema SYSIBM does not need to be specified. If it is not included in the function path, then it is implicitly assumed as the first schema.

This keyword is used as part of the process for resolving unqualified function references that may have been defined in a schema name other than the current user's schema. The order of the schema names determines the order in which the function names will be resolved. For more information on function resolution, refer to the *SQL Reference*.

CURRENTPACKAGESET

Keyword Description:

Issue "SET CURRENT PACKAGESET schema" after every connect.

db2cli.ini Keyword Syntax:

CURRENTPACKAGESET = *schema name*

Default Setting:

The clause is not appended.

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Usage Notes:

This option will issue the command "SET CURRENT PACKAGESET schema" after every connect to a database. By default this clause is not appended.

This statement sets the schema name (collection identifier) that will be used to select the package to use for subsequent SQL statements.

CLI/ODBC applications issue dynamic SQL statements. Using this option you can control the privileges used to run these statements:

- Choose a schema to use when running SQL statements from CLI/ODBC applications.
- Ensure the objects in the schema have the desired privileges and then rebind accordingly.
- Set the CURRENTPACKAGESET option to this schema.

The SQL statements from the CLI/ODBC applications will now run under the specified schema and use the privileges defined there.

Refer to the *SQL Reference* for more information on the SET CURRENT PACKAGESET command.

CURRENTREFRESHAGE

Keyword Description:

Set the value of the CURRENT REFRESH AGE special register.

db2cli.ini Keyword Syntax:

CURRENTREFRESHAGE = 0 | ANY | a numeric constant

Default Setting:

0 - summary tables defined with REFRESH DEFERRED will not be used to optimize the processing of a query

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Usage Notes:

For information on Summary Tables and the SET CURRENT REFRESH AGE statement, see the SQL Reference.

This keyword can be set to one of the following values:

- 0 - Indicates that summary tables defined with REFRESH DEFERRED will not be used to optimize the processing of a query (default).
- 9999999999999999 - Indicates that any summary tables defined with REFRESH DEFERRED or REFRESH IMMEDIATE may be used to optimize the processing of a query. This value represents 9999 years, 99 months, 99 days, 99 hours, 99 minutes, and 99 seconds.
- ANY - This is a shorthand for 9999999999999999.

CURRENTSCHEMA

Keyword Description:

Specify the schema used in a SET CURRENT SCHEMA statement upon a successful connect.

db2cli.ini Keyword Syntax:

CURRENTSCHEMA = *schema name*

Default Setting:

No statement is issued.

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Usage Notes:

Upon a successful connect, if this option is set, a SET CURRENT SCHEMA statement is sent to the DBMS. This allows the end user or application to name SQL objects without having to qualify them by schema name.

For more information on the SET CURRENT SCHEMA statement, see the *SQL Reference*.

CURRENTSQLID**Keyword Description:**

Specify the ID used in a SET CURRENT SQLID statement sent to the DBMS upon a successful connect.

db2cli.ini Keyword Syntax:

CURRENTSQLID = *current_sqlid*

Default Setting:

No statement is issued.

DB2 CLI/ODBC Settings Tab:

Enterprise

Only Applicable when:

connecting to those DB2 DBMS's where SET CURRENT SQLID is supported (such as DB2 for MVS/ESA).

Usage Notes:

Upon a successful connect, if this option is set, a SET CURRENT SQLID statement is sent to the DBMS. This allows the end user and the application to name SQL objects without having to qualify them by schema name.

CURSORHOLD**Keyword Description:**

Effect of a transaction completion on open cursors.

db2cli.ini Keyword Syntax:

CURSORHOLD = 1 | 0

Default Setting:

Selected--Cursors are not destroyed.

DB2 CLI/ODBC Settings Tab:

Transaction

Usage Notes:

This option controls the effect of a transaction completion on open cursors.

1 = cursor hold, the cursors are not destroyed when the transaction is committed (default).

0 = cursor no hold, the cursors are destroyed when the transaction is committed.

Note: Cursors are always destroyed when transactions are rolled back.

This option affects the result returned by `SQLGetInfo()` when called with `SQL_CURSOR_COMMIT_BEHAVIOR` or `SQL_CURSOR_ROLLBACK_BEHAVIOR`. The value of `CURSORHOLD` is ignored if connecting to DB2 for VSE & VM where cursor with hold is not supported.

You can use this option to tune performance. It can be set to cursor no hold (0) if you are sure that your application:

1. Does not have behavior that is dependent on the `SQL_CURSOR_COMMIT_BEHAVIOR` or the `SQL_CURSOR_ROLLBACK_BEHAVIOR` information returned via `SQLGetInfo()`, and
2. Does not require cursors to be preserved from one transaction to the next.

The DBMS will operate more efficiently, as resources no longer need to be maintained after the end of a transaction.

DATABASE**Keyword Description:**

Database on the server to connect to when using a File DSN.

db2cli.ini Keyword Syntax:

`DATABASE = database name`

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The `db2cli.ini` file must be modified directly to make use of this keyword.

Only Applicable when:

PROTOCOL set to TCPIP

See Also:

“HOSTNAME” on page 226, “PROTOCOL” on page 235,
“SERVICENAME” on page 238

Usage Notes:

When using a File DSN you must use this option to specify the database on the server to connect to. This value has nothing to do with any database alias name specified on the client, it must be set to the database name on the server itself.

This setting is only considered when the PROTOCOL option is set to TCPIP.

DB2CONNECTVERSION

Keyword Description:

Specify DB2 Connect or DB2 DDCS gateway version being used.

db2cli.ini Keyword Syntax:

DB2CONNECTVERSION = *gateway version*

Default Setting:

5

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Only Applicable when:

connecting to a data source through a DB2 Connect or DB2 DDCS gateway.

Usage Notes:

This option is used to indicate to the DB2 CLI driver which version of a DB2 Connect or DB2 DDCS gateway is being used. The CLI driver can then use this information to maximize its interaction with the data source (supporting stored procedures that return multiple result sets, for instance).

5 = Indicates that a version 5 DB2 Connect gateway is being used (default).

2 = Indicates that a version 2 DB2 DDCS gateway is being used.

DB2DEGREE

Keyword Description:

Set the degree of parallelism for the execution of SQL statements.

db2cli.ini Keyword Syntax:

DB2DEGREE = 0 | integer value from 1 to 32767 | ANY

Default Setting:

No SET CURRENT DEGREE statement is issued.

DB2 CLI/ODBC Settings Tab:

Optimization

Only Applicable when:

connecting to a cluster database system.

Usage Notes:

This option only applies to a DB2 Version 5.2 or later server. If the value specified is anything other than 0 (the default) then DB2 CLI will issue the following SQL statement after a successful connection:

```
SET CURRENT DEGREE value
```

This specifies the degree of parallelism for the execution of the SQL statements. The database manager will determine the degree of parallelism if you specify ANY.

For more information, see the SET CURRENT DEGREE statement in the *SQL Reference*.

DB2ESTIMATE**Keyword Description:**

Threshold for displaying CLI optimizer estimates after SQL query statement preparation.

db2cli.ini Keyword Syntax:

DB2ESTIMATE = 0 | **large positive number**

Default Setting:

Estimates are not returned.

DB2 CLI/ODBC Settings Tab:

Optimization

Only Applicable when:

a GUI application accesses a DB2 Version 2 or later server.

Usage Notes:

This option determines whether DB2 CLI will display a dialog box to report estimates returned by the DB2 optimizer at the end of SQL query statement preparation.

0 = Estimates are not returned (default).

large positive number = The threshold above which DB2 CLI will display the window to report estimates. This value is compared against the SQLERRD(4) field in the SQLCA associated with the PREPARE. If the value in SQLERRD(4) is greater than DB2ESTIMATE, the estimates window will appear.

The graphic window will display the optimizer estimates, along with push buttons to allow users to choose whether they wish to continue with subsequent execution of this query or cancel it.

The recommended value for DB2ESTIMATE is 60000.

This option is only relevant when connecting to a DB2 version 2 or later database. In order for the window to appear, the application must have a graphical interface.

If this option is used then the DB2 CLI/ODBC option DEFERREDPREPARE will be considered off.

DB2EXPLAIN

Keyword Description:

Determines whether Explain snapshot and/or Explain table information will be generated by the server.

db2cli.ini Keyword Syntax:

DB2EXPLAIN = 0 | 1 | 2 | 3

Default Setting:

Neither Explain snapshot nor Explain table information will be generated by the server.

DB2 CLI/ODBC Settings Tab:

Optimization

Usage Notes:

This keyword determines whether Explain snapshot and/or Explain table information will be generated by the server.

0 = both off (default)

A 'SET CURRENT EXPLAIN SNAPSHOT=NO' and a 'SET CURRENT EXPLAIN MODE=NO' statement will be sent to the server to disable both the Explain snapshot and the Explain table information capture facilities.

1 = Only Explain snapshot facility on

A 'SET CURRENT EXPLAIN SNAPSHOT=YES' and a 'SET CURRENT EXPLAIN MODE=NO' statement will be sent to the server to enable the Explain snapshot facility, and disable the Explain table information capture facility.

2 = Only Explain table information capture facility on

A 'SET CURRENT EXPLAIN MODE=YES' and a 'SET CURRENT EXPLAIN SNAPSHOT=NO' will be sent to the server to enable the Explain table information capture facility and disable the Explain snapshot facility.

3 = Both on

A 'SET CURRENT EXPLAIN MODE=YES' and a 'SET CURRENT EXPLAIN SNAPSHOT=YES' will be sent to the server to enable both the Explain snapshot and the Explain table information capture facilities.

Explain information is inserted into Explain tables, which must be created before the Explain information can be generated. For more information on these tables, refer to the *SQL Reference*.

The current authorization ID must have INSERT privilege for the Explain tables.

Option 1 is only valid when connecting to a DB2 Common Server version 2.1.0 or later database; options 2 and 3 when connecting to a DB2 Common Server version 2.1.1 or later database.

DB2OPTIMIZATION

Keyword Description:

Set the query optimization level.

db2cli.ini Keyword Syntax:

DB2OPTIMIZATION = *integer value from 0 to 9*

Default Setting:

No SET CURRENT QUERY OPTIMIZATION statement issued.

DB2 CLI/ODBC Settings Tab:

Optimization

Only Applicable when:

when connecting to a DB2 Version 2 server or later.

Usage Notes:

If this option is set then DB2 CLI will issue the following SQL statement after a successful connection:

```
SET CURRENT QUERY OPTIMIZATION positive number
```

This specifies the query optimization level at which the optimizer should operate the SQL queries. Refer to the *SQL Reference* for the allowable optimization levels.

DBALIAS

Keyword Description:

Enables Data Source Names greater than 8 characters.

db2cli.ini Keyword Syntax:

DBALIAS = *dbalias*

Default Setting:

Use the DB2 database alias as the ODBC Data Source Name.

DB2 CLI/ODBC Settings Tab:

CLI/ODBC Settings General

Usage Notes:

This keyword allows for Data Source Names of greater than 8 single byte characters. The Data Source Name (DSN) is the name, enclosed in square brackets, that denotes the section header in the `db2cli.ini` file (on platforms where this is an ASCII file). Typically, this section header is the database alias name which has a maximum length of 8 bytes. A user who wishes to refer to the data source with a longer, more meaningful name, can place the longer name in the section header, and set this keyword value to the database alias used on the CATALOG command. Here is an example:

```
; The much longer name maps to an 8 single byte character dbalias  
[MyMeaningfulName]  
DBALIAS=DB2DBT10
```

The end user can specify `[MyMeaningfulName]` as the name of the data source on connect while the actual database alias is `DB2DBT10`.

In a 16-bit Windows ODBC environment, under the `[ODBC DATA SOURCES]` entry in the `ODBC.INI` file, the following line must also be updated with the long alias name (*dbname*).

```
< alias >=IBM DB2 ODBC DRIVER
```

DBNAME

Keyword Description:

Specify the database name to reduce the time it takes for the application to query MVS table information.

db2cli.ini Keyword Syntax:

DBNAME = *dbname*

Default Setting:

Don't filter on the DBNAME column.

DB2 CLI/ODBC Settings Tab:

Enterprise

Only Applicable when:

connecting to DB2 for MVS/ESA.

See Also:

“SCHEMALIST” on page 237, “TABLETYPE” on page 244

Usage Notes:

This option is only used when connecting to DB2 for MVS/ESA, and only if (*base*) table catalog information is requested by the application. If a large number of tables exist in the DB2 for MVS/ESA subsystem, a *dbname* can be specified to reduce the time it takes for the application to query table information, and reduce the number of tables listed by the application.

If this option is set then the statement `IN DATABASE dbname` will be appended to various statements such as `CREATE TABLE`.

This value maps to the `DBNAME` column in the DB2 for MVS/ESA system catalog tables. If no value is specified, or if views, synonyms, system tables, or aliases are also specified via `TABLETYPE`, only table information will be restricted; views, aliases, and synonyms are not restricted with `DBNAME`. It can be used in conjunction with `SCHEMALIST`, and `TABLETYPE` to further limit the number of tables for which information will be returned.

DEFAULTPROCLIBRARY**Keyword Description:**

Set default stored procedure library.

db2cli.ini Keyword Syntax:

`DEFAULTPROCLIBRARY = < full path name >`

Default Setting:

Do not add a default stored procedure library to stored procedure calls.

DB2 CLI/ODBC Settings Tab:

Environment

Only Applicable when:

application is not using the stored procedure catalog table.

Usage Notes:

This option should only be used on a temporary basis; the stored procedure catalog table should be used instead. See the *SQL Reference* for more information.

The library pointed to by this option will be used in all stored procedure calls that do not already explicitly specify a library. Because you are specifying a location on the server machine, you must use the path format of that operating system, not of the client. For more information, see the *CALL* statement in the *SQL Reference*.

For instance, if the stored procedures are located on the server in the library file `d:\terry\proclib\comstor`, you could set `DEFAULTPROCLIBRARY` to `d:\terry\proclib\comstor`, then call the stored procedure *func* without specifying a library. The resulting SQL statement sent would be:

```
CALL d:\terry\proclib\comstor!func
```

DEFERREDPREPARE

Keyword Description:

Minimize network flow by combining the PREPARE request with the corresponding execute request.

db2cli.ini Keyword Syntax:

```
DEFERREDPREPARE = 0 | 1
```

Default Setting:

The prepare request will be delayed until the execute request is sent.

DB2 CLI/ODBC Settings Tab:

Compatibility

Not Applicable when:

DB2ESTIMATE is set.

Usage Notes:

Defers sending the PREPARE request until the corresponding execute request is issued. The two requests are then combined into one command/reply flow (instead of two) to minimize network flow and to improve performance.

The default behavior has changed from DB2 version 2. Deferred prepare is now the default and must be explicitly turned off if required.

- 0 = Disable deferred prepare. The PREPARE request will be executed the moment it is issued.
- 1 (default) = Enable deferred prepare. Defer the execution of the PREPARE request until the corresponding execute request is issued.

If the target DB2 Common Server database or the DDCS gateway does not support deferred prepare, the client disables deferred prepare for that connection.

Note: When deferred prepare is enabled, the row and cost estimates normally returned in the SQLERRD(3) and SQLERRD(4) of the SQLCA of a PREPARE statement may become zeros. This may be of concern to users who want to use these values to decide whether or not to continue the SQL statement.

This option is turned off if the CLI/ODBC option DB2ESTIMATE is set to a value other than zero.

DISABLEMULTITHREAD

Keyword Description:

Disable Multithreading.

db2cli.ini Keyword Syntax:

DISABLEMULTITHREAD = 0 | 1

Default Setting:

Multithreading is enabled.

DB2 CLI/ODBC Settings Tab:

Compatibility

Usage Notes:

The CLI/ODBC driver is capable of supporting multiple concurrent threads.

This option is used to enable or disable multi-thread support.

0 = Multithreading is enabled (default).

1 = Disable Multithreading.

If multithreading is disabled then all calls for all threads will be serialized at the process level. Use this setting for multithreaded applications that require the serialized behavior of DB2 Version 2.

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

EARLYCLOSE

Keyword Description:

Should the cursor associated with the connection be closed early by the DB2 server when it encounters the end of the result set?

db2cli.ini Keyword Syntax:

EARLYCLOSE = 1 | 0

Default Setting:

EARLYCLOSE behavior is on.

DB2 CLI/ODBC Settings Tab:

Compatibility

Usage Notes:

This option specifies whether or not the temporary cursor on the server can be automatically closed, without closing the cursor on the client, when the last record is sent to the client.

0 = Do not close the temporary cursor on the server early.

1 = Close the temporary cursor on the server early (default).

This saves the CLI/ODBC driver a network request by not issuing the statement to explicitly close the cursor because it knows that it has already been closed.

Having this option on will speed up applications that make use of many small result sets.

The EARLYCLOSE feature is not used if either:

- The statement disqualifies for blocking.
- The cursor type is anything other than SQL_CURSOR_FORWARD_ONLY.

Note: Although this option can be set at any time, the option value used is the one that exists when the statement is executed (when the cursor is opened).

GRANTEELIST

Keyword Description:

Reduce the amount of information returned when the application gets a list of table or column privileges.

db2cli.ini Keyword Syntax:

GRANTEELIST = " 'userID1', 'userID2',... 'userIDn' "

Default Setting:

Do not filter the results.

DB2 CLI/ODBC Settings Tab:

Enterprise

See Also:

"GRANTORLIST" on page 224

Usage Notes:

This option can be used to reduce the amount of information returned when the application gets a list of privileges for tables in a database, or columns in a table. The list of authorization IDs specified is used as a filter; the only tables or columns that are returned are those with privileges that have been granted *TO* those IDs.

Set this option to a list of one or more authorization IDs that have been granted privileges, delimited with single quotes, and separated by commas. The entire string must also be enclosed in double quotes. For example:

```
GRANTEELIST=" 'USER1', 'USER2', 'USER8' "
```

In the above example, if the application gets a list of privileges for a specific table, only those columns that have a privilege granted *TO* USER1, USER2, or USER8 would be returned.

GRANTORLIST

Keyword Description:

Reduce the amount of information returned when the application gets a list of table or column privileges.

db2cli.ini Keyword Syntax:

```
GRANTORLIST = " 'userID1', 'userID2',... 'userIDn' "
```

Default Setting:

Do not filter the results.

DB2 CLI/ODBC Settings Tab:

Enterprise

See Also:

“GRANTEELIST” on page 223

Usage Notes:

This option can be used to reduce the amount of information returned when the application gets a list of privileges for tables in a database, or columns in a table. The list of authorization IDs specified is used as a filter; the only tables or columns that are returned are those with privileges that have been granted *BY* those IDs.

Set this option to a list of one or more authorization IDs that have granted privileges, delimited with single quotes, and separated by commas. The entire string must also be enclosed in double quotes. For example:

```
GRANTORLIST=" 'USER1', 'USER2', 'USER8' "
```

In the above example, if the application gets a list of privileges for a specific table, only those columns that have a privilege granted *BY* USER1, USER2, or USER8 would be returned.

GRAPHIC

Keyword Description:

Controls whether DB2 CLI reports the IBM GRAPHIC (double byte character support) as one of the supported data types.

db2cli.ini Keyword Syntax:

```
GRAPHIC = 0 | 1 | 2 | 3
```

Default Setting:

GRAPHIC is not returned as a supported data type.

DB2 CLI/ODBC Settings Tab:

Data Type

Usage Notes:

This option controls how two related pieces of information are returned by the application:

- Whether DB2 CLI reports the IBM GRAPHIC (double byte character support) as one of the supported data types when `SQLGetTypeInfo()` is called. `SQLGetTypeInfo()` lists the data types supported by the DB2 database in the current connection.
- What unit is used to report the length of graphic columns. This applies to all DB2 CLI/ODBC functions that return length/precision either on the output argument or as part of the result set.
 - 0 = Do not report IBM GRAPHIC data type as a supported type. Length of graphic columns returned as number of DBCS characters. (default)
 - 1 = Report IBM GRAPHIC data type as supported. Length of graphic columns returned as number of DBCS characters.
 - 2 = Do not report IBM GRAPHIC data type as a supported type. Length of graphic columns returned as number of bytes. (This is needed for **Microsoft Access** 1.1-J** and **Microsoft Query**-J**.)
 - 3 = Settings 1 and 2 combined. IBM GRAPHIC data type reported as supported. Length of graphic columns returned as number of bytes.

The default is that GRAPHIC is not returned since many off the shelf applications do not recognize this data type and cannot provide proper handling.

HOSTNAME

Keyword Description:

The server system's host name or IP address, used with file DSN.

db2cli.ini Keyword Syntax:

HOSTNAME = *host name* | *IP Address*

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

File DSN

Only Applicable when:

PROTOCOL set to TCPIP

See Also:

"PROTOCOL" on page 235, "SERVICENAME" on page 238

Usage Notes:

Use this option in conjunction with the SERVICENAME option to specify the required attributes for a TCP/IP connection from this client machine to a server running DB2. These two values are only considered when the PROTOCOL option is set to TCPIP.

Specify either the server system's host name or its IP address.

IGNOREWARNINGS

Keyword Description:

Ignore Warnings.

db2cli.ini Keyword Syntax:

IGNOREWARNINGS = 0 | 1

Default Setting:

Warnings are returned as normal.

DB2 CLI/ODBC Settings Tab:

Service

See Also:

"WARNINGLIST" on page 252, "IGNOREWARNLIST" on page 227

Usage Notes:

On rare occasions an application will not correctly handle warning messages. This option can be used to indicate that warnings from the database manager are not to be passed on to the application.

0 = Warnings reported as usual (default).
1 = Database manager warnings are ignored, SQL_SUCCESS is returned.
Warnings from the DB2 CLI/ODBC driver are still returned; many are required for normal operation.

Although this option can be used on its own, it can also be used in conjunction with the WARNINGLIST CLI/ODBC configuration keyword.

IGNOREWARNLIST

Keyword Description:

Ignore specified sqlstates.

db2cli.ini Keyword Syntax:

```
IGNOREWARNLIST = "sqlstate1', 'sqlstate2', ..."
```

Default Setting:

Warnings are returned as normal

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

See Also:

"WARNINGLIST" on page 252, "IGNOREWARNINGS" on page 226

Usage Notes:

On rare occasions an application may not correctly handle some warning messages, but does not want to ignore all warning messages. This keyword can be used to indicate which warnings are not to be passed on to the application. The IGNOREWARNINGS keyword should be used if all database manager warnings are to be ignored.

If an sqlstate is included in both IGNOREWARNLIST and WARNINGLIST, it will be ignored altogether.

Each sqlstate must be in uppercase, delimited with single quotes and separated by commas. The entire string must also be enclosed in double quotes. For example:

```
IGNOREWARNLIST="01000', '01004', '01504'"
```

KEEPCONNECT

Keyword Description:

Number of connections to cache.

db2cli.ini Keyword Syntax:

```
KEEPCONNECT = 0 | positive integer
```

Default Setting:

Do not cache connections.

DB2 CLI/ODBC Settings Tab:

Transaction

Usage Notes:

0 = Do not cache database connections (default).

Setting this option to a value greater than zero can speed up applications that constantly connect to and disconnect from the same database using the same connection information.

Instead of closing the connection each time, then re-opening it again, the CLI/ODBC driver will keep the connection open and cache the connection information. When the request to connect to the same database occurs a second time, the existing connection is used. This saves the time, resources, and network flow to close the first connection, as well as to re-open the second connection.

The value set for this option indicates the number of database connections to cache. Although the maximum is limited only by system resources, usually a value of 1 or 2 is sufficient for applications that will benefit at all from this behavior.

KEEPSTATEMENT**Keyword Description:**

Number of statement handles to cache.

db2cli.ini Keyword Syntax:

KEEPSTATEMENT = 5 | positive integer

Default Setting:

Cache 5 statement handles.

DB2 CLI/ODBC Settings Tab:

Optimization

Usage Notes:

By default, the memory required for 5 statement handles is cached. When a statement handle is closed, the memory used for that handle is not deallocated but is instead used when the next statement handle is allocated.

The value set for this option determines how many statement handles are cached. It can be set to less than 5 to explicitly reduce the amount of memory

used by the statement cache. It can be increased above 5 to improve performance for applications that open, close, and then re-open large sets of statements.

The maximum number of cached statement handles is determined by system resources.

LOBMAXCOLUMNSIZE

Keyword Description:

Override default COLUMN_SIZE for LOB data types.

db2cli.ini Keyword Syntax:

LOBMAXCOLUMNSIZE = *integer greater than zero*

Default Setting:

2 Gigabytes (1G for DBCLOB)

DB2 CLI/ODBC Settings Tab:

Data Type

Only Applicable when:

LONGDATACOMPAT option is used.

See Also:

"LONGDATACOMPAT"

Usage Notes:

This will override the 2 Gigabyte (1G for DBCLOB) value that is returned by SQLGetTypeInfo() for the COLUMN_SIZE column for SQL_CLOB, SQL_BLOB, and SQL_DBCLOB SQL data types. Subsequent CREATE TABLE statements that contain LOB columns will use the column size value you set here instead of the default.

LONGDATACOMPAT

Keyword Description:

Report LOBs as long data types or as large object types.

db2cli.ini Keyword Syntax:

LONGDATACOMPAT = 0 | 1

Default Setting:

Reference LOB data types as large object types.

DB2 CLI/ODBC Settings Tab:

Data Type

See Also:

"LOBMAXCOLUMNSIZE"

Usage Notes:

This option indicates to DB2 CLI what data type the application expects when working with a database with large object (LOB) columns.

Database data type	Large Objects (0--Default)	Long Data Types (1)
CLOB	SQL_CLOB	SQL_LONGVARCHAR
BLOB	SQL_BLOB	SQL_LONGVARBINARY
DBCLOB	SQL_DBCLOB	SQL_LONGVARGRAPHIC

This option is useful when running ODBC applications that cannot handle the large object data types.

The DB2 CLI/ODBC option LOBMAXCOLUMNSIZE can be used in conjunction with this option to reduce the default size declared for the data.

MAXCONN

Keyword Description:

Maximum number of connections allowed for each application.

db2cli.ini Keyword Syntax:

MAXCONN = 0 | **positive number**

Default Setting:

As many connections as permitted by system resources.

DB2 CLI/ODBC Settings Tab:

Transaction

Usage Notes:

This option is used to specify the maximum number of connections allowed for each CLI/ODBC application. This can be used as a governor for the maximum number of connections an administrator may wish to restrict each application to open. A value of 0 may be used to represent *no limit*; that is, an application is allowed to open up as many connections as permitted by the system resources.

On OS/2 and WIN32 platforms (Windows NT and Windows 95), if the NetBIOS protocol is in use, this value corresponds to the number of connections (NetBIOS sessions) that will be concurrently set up by the application. The range of values for OS/2 NetBIOS is 1 to 254. Specifying 0 (the default) will result in 5 *reserved* connections. *Reserved NetBIOS sessions* cannot be used by other applications. The number of connections specified by

this parameter will be applied to any adapter that the DB2 NetBIOS protocol uses to connect to the remote server (adapter number is specified in the node directory for a NetBIOS node).

MODE

Keyword Description:

Default connect mode.

db2cli.ini Keyword Syntax:

MODE = SHARE | EXCLUSIVE

Default Setting:

SHARE

DB2 CLI/ODBC Settings Tab:

Transaction

Not Applicable when:

connecting to a DRDA database.

Usage Notes:

Sets the CONNECT mode to either SHARE or EXCLUSIVE. If a mode is set by the application at connect time, this value is ignored. The default is SHARE.

Note: EXCLUSIVE is not permitted for DRDA connections. Refer to the *SQL Reference* for more information on the CONNECT statement.

MULTICONNECT

Keyword Description:

How SQLConnect() requests are mapped to physical database connections.

db2cli.ini Keyword Syntax:

MULTICONNECT = 0 | 1

Default Setting:

Each SQLConnect() request by the application will result in a physical database connection.

DB2 CLI/ODBC Settings Tab:

Transaction

Usage Notes:

This option is used to specify how SQLConnect() requests are mapped to physical database connections.

1 = Connections are not shared, multiple connections are used (default) -- Each SQLConnect() request by the application will result in a physical database connection.

0 = Connections are mapped to one physical connection, one connection is used -- All connections for the application are mapped to one physical connection. This may be useful if:

- the ODBC application runs out of file handles because it uses so many connections.
- the application only reads data from the database
- the application uses autocommit (in some cases)
- the application opens multiple connections instead of using multiple statements on one connection. The use of multiple connections in this case may cause locking contention between connections.

If MULTICONNECT is set to 0 then multithreading must be disabled using the keyword DISABLEMULTITHREAD

Note: If MULTICONNECT is set off then all statements are executed on the same connection and therefore in the same transaction. This means that a rollback will roll back ALL statements on all connections. Be sure that the application is designed to work with MULTICONNECT off before doing so or the application may not operate correctly.

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

OPTIMIZEFORNROWS

Keyword Description:

Append "OPTIMIZE FOR n ROWS" clause to every select statement.

db2cli.ini Keyword Syntax:

OPTIMIZEFORNROWS = *integer*

Default Setting:

The clause is not appended.

DB2 CLI/ODBC Settings Tab:

Optimization

Usage Notes:

This option will append the "OPTIMIZE FOR n ROWS" clause to every select statement, where n is an integer larger than 0. If set to 0 (the default) this clause will not be appended.

For more information on the effect of the OPTIMIZE FOR n ROWS clause, refer to the *Administration Guide*.

OPTIMIZE SQLCOLUMNS

Keyword Description:

Optimize SQLColumns() call with explicit Schema and Table Name.

db2cli.ini Keyword Syntax:

OPTIMIZE SQLCOLUMNS = 0 | 1

Default Setting:

0 - all column information returned

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Usage Notes:

If OPTIMIZE SQLCOLUMNS is on (set to 1), then all calls to SQLColumns() will be optimized if an explicit (no wildcard specified) Schema Name, explicit Table Name, and % (ALL columns) for Column Name are specified. The DB2 CLI/ODBC Driver will optimize this call so that the system tables will not be scanned. If the call is optimized then the COLUMN_DEF information (which contains the default string for the columns) is not returned. When connecting to an AS/400 database, the information returned by SQLColumns() for columns whose data type is NUMERIC will be incorrect. If the application does not need this information then it can turn on the optimization to increase performance.

If the application needs the COLUMN_DEF information then OPTIMIZE SQLCOLUMNS should be set to 0. This is the default.

PATCH1

Keyword Description:

Use work-arounds for known problems with ODBC applications.

db2cli.ini Keyword Syntax:

PATCH1 = { 0 | 1 | 2 | 4 | 8 | 16 | ... }

Default Setting:

Use no work-arounds.

DB2 CLI/ODBC Settings Tab:

Service

See Also:

"PATCH2" on page 234

Usage Notes:

This keyword is used to specify a work-around for known problems with ODBC applications. The value specified can be for none, one, or multiple work-arounds. The patch values specified here are used in conjunction with any PATCH2 values that may also be set.

Using the DB2 CLI/ODBC Settings notebook you can select one or more patches to use. If you set the values in the `db2cli.ini` file itself and want to use multiple patch values then simply add the values together to form the keyword value. For example, if you want the patches 1, 4, and 8, then specify `PATCH1=13`.

0 = No work around (default)

The DB2 CLI/ODBC Settings notebook has a list of values. Select the Service folder in the DB2 folder for information on how to update this list of values. This information is also contained in the README file (there will be no such section in the README if there are no current patch values for that platform).

PATCH2

Keyword Description:

Use work-arounds for known problems with CLI/ODBC applications.

db2cli.ini Keyword Syntax:

`PATCH2 = "patch value 1, patch value 2, patch value 3, ..."`

Default Setting:

Use no work-arounds

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The `db2cli.ini` file must be modified directly to make use of this keyword.

See Also:

"PATCH1" on page 233

Usage Notes:

This keyword is used to specify a work-around for known problems with CLI/ODBC applications. The value specified can be for none, one, or multiple work-arounds. The patch values specified here are used in conjunction with any PATCH1 values that may also be set.

When specifying multiple patches, the values are specified in a comma delimited string (unlike the PATCH1 option where the values are added together and the sum is used).

0 = No work around (default)

To set PATCH2 values 3, 4 and 8 you would specify:

```
PATCH2="3, 4, 8"
```

The PATCH2 values are contained in the README file (there will be no such section in the README if there are no current patch values for that platform).

POPUPMESSAGE

Keyword Description:

Pop up a message box every time CLI/ODBC generates an error.

db2cli.ini Keyword Syntax:

```
POPUPMESSAGE = 0 | 1
```

Default Setting:

Do not display message box.

DB2 CLI/ODBC Settings Tab:

Service

Only Applicable when:

running OS/2 or Windows applications.

See Also:

"SQLSTATEFILTER" on page 239

Usage Notes:

Pops up a message box every time DB2 CLI generates an error that can be retrieved using SQLGetDiagRec() or SQLError(). Useful for debugging applications that do not report messages to users.

0 = do NOT display message box (default)

1 = display message box

PROTOCOL

Keyword Description:

Communications protocol used for File DSN.

db2cli.ini Keyword Syntax:

```
PROTOCOL = TCPIP
```

Default Setting:

none

DB2 CLI/ODBC Settings Tab:

File DSN

See Also:

“HOSTNAME” on page 226, “SERVICENAME” on page 238

Usage Notes:

TCP/IP is the only protocol supported when using a File DSN. Set the option to the string TCPIP (without the slash).

When this option is set then the following options must also be set:

- “DATABASE” on page 214
- “SERVICENAME” on page 238
- “HOSTNAME” on page 226

PWD**Keyword Description:**

Define default password.

db2cli.ini Keyword Syntax:

PWD = *password*

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

CLI/ODBC Settings General

Usage Notes:

This *password* value is used if a password is not provided by the application at connect time.

It is stored as plain text and is therefore not secure.

QUERYTIMEOUTINTERVAL**Keyword Description:**

Delay (in seconds) between checking for a query timeout

db2cli.ini Keyword Syntax:

QUERYTIMEOUTINTERVAL = 0 | **positive integer**

Default Setting:

5 seconds

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Usage Notes:

An application can use the SQLSetStmtAttr() function to set the SQL_ATTR_QUERY_TIMEOUT statement attribute. This indicates the number of seconds to wait for an SQL statement to execute before returning to the application

The QUERYTIMEOUTINTERVAL configuration keyword is used to indicate how long the CLI driver should wait between checks to see if the query has completed.

For instance, suppose SQL_ATTR_QUERY_TIMEOUT is set to 25 seconds (timeout after waiting for 25 seconds), and QUERYTIMEOUTINTERVAL is set to 10 seconds (check the query every 10 seconds). The query will not time out until 30 seconds (the first check AFTER the 25 second limit).

There may be cases where the SQL_ATTR_QUERY_TIMEOUT is set to a value which is too low, and the query should NOT be timed-out. If the application cannot be modified (i.e., a third party ODBC application), then the QUERYTIMEOUTINTERVAL can be set to 0, and the CLI driver will ignore the SQL_ATTR_QUERY_TIMEOUT setting.

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

SCHEMALIST**Keyword Description:**

Restrict schemas used to query table information.

db2cli.ini Keyword Syntax:

```
SCHEMALIST = " 'schema1', 'schema2',... 'schemaN' "
```

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

Enterprise

Usage Notes:

SCHEMALIST is used to provide a more restrictive default, and therefore improve performance, for those applications that list every table in the DBMS.

If there are a large number of tables defined in the database, a schema list can be specified to reduce the time it takes for the application to query table information, and reduce the number of tables listed by the application. Each schema name is case-sensitive, must be delimited with single quotes, and separated by commas. The entire string must also be enclosed in double quotes. For example:

```
SCHEMALIST="'USER1', 'USER2', 'USER3'"
```

For DB2 for MVS/ESA, CURRENT SQLID can also be included in this list, but without the single quotes, for example:

```
SCHEMALIST="'USER1', CURRENT SQLID, 'USER3'"
```

The maximum length of the string is 256 characters.

This option can be used in conjunction with DBNAME and TABLETYPE to further limit the number of tables for which information will be returned.

SERVICENAME

Keyword Description:

The server system's service name or port number, used with file DSN.

db2cli.ini Keyword Syntax:

SERVICENAME = *service name* | *port number*

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

File DSN

Only Applicable when:

PROTOCOL set to TCPIP

See Also:

"PROTOCOL" on page 235, "HOSTNAME" on page 226

Usage Notes:

Use this option in conjunction with the HOSTNAME option to specify the required attributes for a TCP/IP connection from this client machine to a server running DB2. These two values are only considered when the PROTOCOL option is set to TCPIP.

Specify either the server system's service name or its port number.

SQLSTATEFILTER

Keyword Description:

Do not pop up an error message for the defined SQLSTATES.

db2cli.ini Keyword Syntax:

SQLSTATEFILTER = " 'XXXXX', 'YYYYY', ... "

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

Service

Only Applicable when:

POPUPMESSAGE option is turned on.

See Also:

"POPUPMESSAGE" on page 235

Usage Notes:

Use in conjunction with the POPUPMESSAGE option. This prevents DB2 CLI from displaying errors that are associated with the defined states.

Each SQLSTATE must be in upper case, delimited with single quotes and separated by commas. The entire string must also be enclosed in double quotes. For example:

```
SQLSTATEFILTER=" 'HY1090', '01504', '01508' "
```

STATICCAPFILE

Keyword Description:

Specify the Capture File name and optionally the directory where it will be saved.

db2cli.ini Keyword Syntax:

STATICCAPFILE = < Full file name >

Default Setting:

None - you must specify a capture file name.

DB2 CLI/ODBC Settings Tab:

Static SQL

Only Applicable when:

STATICMODE is set to Capture or Match

See Also:

“STATICLOGFILE”, “STATICMODE”, “STATICPACKAGE” on page 241

Usage Notes:

This keyword is used to specify the Capture File name and optionally the directory where it will be saved.

For more information on running CLI/ODBC applications as static SQL, see the STATICMODE keyword.

STATICLOGFILE

Keyword Description:

Specify the Static Profiling Log File name and optionally the directory where it will be saved.

db2cli.ini Keyword Syntax:

STATICLOGFILE = < **Full file name** >

Default Setting:

No Static Profiling Log is created. If a filename is specified without a pathname then the current path will be used.

DB2 CLI/ODBC Settings Tab:

Static SQL

Only Applicable when:

STATICMODE is set to Capture or Match

See Also:

“STATICCAPFILE” on page 239, “STATICMODE”, “STATICPACKAGE” on page 241

Usage Notes:

This keyword is used to specify the Static Profiling Log File name and optionally the directory where it will be saved.

For more information on running CLI/ODBC applications as static SQL, see the STATICMODE keyword.

STATICMODE

Keyword Description:

Specify whether the CLI/ODBC application will capture SQL or use a static SQL Package for this DSN.

db2cli.ini Keyword Syntax:

STATICMODE = DISABLED | CAPTURE | MATCH

Default Setting:

0 Disabled - SQL statements are not captured and no static SQL package is used.

DB2 CLI/ODBC Settings Tab:

Static SQL

See Also:

“STATICCAPFILE” on page 239, “STATICPACKAGE”,
“STATICLOGFILE” on page 240

Usage Notes:

This option allows you to specify how the SQL issued by the CLI/ODBC application for this DSN will be processed:

- **DISABLED** = Static mode disabled. No special processing. The CLI/ODBC statements will be executed as dynamic SQL with no change. This is the default.
- **CAPTURE** = Capture Mode. Execute the CLI/ODBC statements as dynamic SQL. If the SQL statements are successful, they will be captured into a file (known as the Capture File) to be bound by the DB2CAP command later.
- **MATCH** = Match mode. Execute the CLI/ODBC statements as static SQL statements if a matching statement is found in the Capture File specified in STATICCAPFILE. The Capture File must first be bound by the DB2CAP command. See the Command Reference for details.

For more information about running CLI/ODBC applications as static SQL, see the Release Notes and the *CLI Guide and Reference*. Additional information is also available on the Internet at <http://www.ibm.com/software/data/db2/udb/staticcli>

STATICPACKAGE**Keyword Description:**

Specify the package to be used with the static profiling feature.

db2cli.ini Keyword Syntax:

STATICPACKAGE = *collection_id.package_name*

Default Setting:

None - you must specify a package name.

DB2 CLI/ODBC Settings Tab:

Static SQL

Only Applicable when:

STATICMODE is set to CAPTURE

See Also:

“STATICCAPFILE” on page 239, “STATICMODE” on page 240,
“STATICLOGFILE” on page 240

Usage Notes:

This keyword is used to specify the package to be used when the application runs in Match Mode. You first need to use Capture Mode to create the Capture File.

Only the first 7 characters of the indicated package name will be used. A one-byte suffix will be added to represent each isolation level, as follows:

- 0 for Uncommitted Read (UR)
- 1 for Cursor Stability (CS)
- 2 for Read Stability (RS)
- 3 for Repeatable Read (RR)
- 4 for No Commit (NC)

For more information on running CLI/ODBC applications as static SQL, see the STATICMODE keyword.

SYNCPPOINT

Keyword Description:

How commits and rollbacks are coordinated among multiple database (DUOW) connections.

db2cli.ini Keyword Syntax:

SYNCPPOINT = 1 | 2

Default Setting:

1 Phase commit.

DB2 CLI/ODBC Settings Tab:

Transaction

Only Applicable when:

default connect type set to Coordinated Connections
(CONNECTTYPE=2)

See Also:

“CONNECTTYPE” on page 210

Usage Notes:

Use this option to specify how commits and rollbacks will be coordinated among multiple database (DUOW) connections. It is only relevant when the default connect type is set to Coordinated connections (CONNECTTYPE = 2).

- 1 = ONEPHASE (default)

A Transaction Manager is not used to perform two phase commit but one phase commit is used to commit the work done by each database in a multiple database transaction.

- 2 = TWOPHASE

A Transaction Manager is required to coordinate two phase commits among those databases that support this.

SYSSCHEMA

Keyword Description:

Indicates an alternative schema to be searched in place of the SYSIBM (or SYSTEM, QSYS2) schemas.

db2cli.ini Keyword Syntax:

SYSSCHEMA = *syschema*

Default Setting:

No alternatives specified.

DB2 CLI/ODBC Settings Tab:

Enterprise

Usage Notes:

This option indicates an alternative schema to be searched in place of the SYSIBM (or SYSTEM, QSYS2) schemas when the DB2 CLI and ODBC Catalog Function calls are issued to obtain system catalog information.

Using this schema name the system administrator can define a set of views consisting of a subset of the rows for each of the following system catalog tables:

DB2 Universal Database	DB2 for MVS/ESA	DB2 for VSE & VM	OS/400	DB2 Universal Database for AS/400
SYSTABLES	SYSTABLES	SYSCATALOG	SYSTABLES	SYSTABLES
SYSCOLUMNS	SYSCOLUMNS	SYSCOLUMNS	SYSCOLUMNS	SYSCOLUMNS
SYSINDEXES	SYSINDEXES	SYSINDEXES	SYSINDEXES	SYSINDEXES
SYSTABAUTH	SYSTABAUTH	SYSTABAUTH		SYSCST
SYSRELS	SYSRELS	SYSKEYCOLS		SYSKEYCST
SYSDATATYPES	SYSSYNONYMS	SYSSYNONYMS		SYSCSTCOL
SYS PROCEDURES	SYSKEYS	SYSKEYS		SYSKEYS
SYS PROC PARMS	SYSCOLAUTH	SYSCOLAUTH		SYSREFCST
	SYSFOREIGNKEYS			
	SYS PROCEDURES (1)			
	SYS DATABASE			

(1) DB2 for MVS/ESA 4.1 only.

For example, if the set of views for the system catalog tables is in the ACME schema, then the view for SYSIBM.SYSTABLES is ACME.SYSTABLES; and SYSSCHEMA should then be set to ACME.

Defining and using limited views of the system catalog tables reduces the number of tables listed by the application, which reduces the time it takes for the application to query table information.

If no value is specified, the default is:

- SYSCAT or SYSIBM on DB2 Universal Database
- SYSIBM on DB2 for common server versions prior to 2.1, DB2 for MVS/ESA and OS/400
- SYSTEM on DB2 for VSE & VM
- QSYS2 on DB2 Universal Database for AS/400

This keyword can be used in conjunction with SCHEMALIST and TABLETYPE (and DBNAME on DB2 for MVS/ESA) to further limit the number of tables for which information will be returned.

TABLETYPE

Keyword Description:

Define a default list of TABLETYPES returned when querying table information.

db2cli.ini Keyword Syntax:

```
TABLETYPE = " 'TABLE' | ',ALIAS' | ',VIEW' | ',INOPERATIVE  
VIEW' | ',SYSTEM TABLE' | ',SYNONYM' "
```

Default Setting:

No default list of TABLETYPES is defined.

DB2 CLI/ODBC Settings Tab:

Enterprise

Usage Notes:

If there is a large number of tables defined in the database, a *tabletype* string can be specified to reduce the time it takes for the application to query table information, and reduce the number of tables listed by the application.

Any number of the values can be specified. Each type must be delimited with single quotes, separated by commas, and in uppercase. The entire string must also be enclosed in double quotes. For example:

```
TABLETYPE="'TABLE', 'VIEW' "
```

This option can be used in conjunction with `DBNAME` and `SCHEMALIST` to further limit the number of tables for which information will be returned.

`TABLETYPE` is used to provide a default for the DB2 CLI function that retrieves the list of tables, views, aliases, and synonyms in the database. If the application does not specify a table type on the function call, and this keyword is not used, information about all table types is returned. If the application does supply a value for the *tabletype* on the function call, then that argument value will override this keyword value.

If `TABLETYPE` includes any value other than `TABLE`, then the `DBNAME` keyword setting cannot be used to restrict information to a particular DB2 for MVS/ESA database.

TEMPDIR

Keyword Description:

Define the directory used for temporary files associated with LOB fields.

db2cli.ini Keyword Syntax:

```
TEMPDIR = < full path name >
```

Default Setting:

Use the system temporary directory.

DB2 CLI/ODBC Settings Tab:

Environment

Usage Notes:

When working with Large Objects (CLOBS, BLOBS, etc...), a temporary file is often created on the client machine to store the information. Using this option

you can specify a location for these temporary files. The system temporary directory will be used if nothing is specified.

The keyword is placed in the data source specific section of the db2cli.ini file, and has the following syntax:

- TempDir= F:\DB2TEMP

When a Large Object is accessed, an SQLSTATE of HY507 will be returned if the path name is invalid, or if the temporary files cannot be created in the directory specified.

TRACE

Keyword Description:

Turn on the DB2 CLI/ODBC trace facility.

db2cli.ini Keyword Syntax:

TRACE = 0 | 1

Default Setting:

No trace information is captured.

DB2 CLI/ODBC Settings Tab:

Service

See Also:

“TRACEFILENAME” on page 247, “TRACEFLUSH” on page 248,
“TRACEPATHNAME” on page 249

Usage Notes:

When this option is on (1), CLI/ODBC trace records are appended to the file indicated by the TRACEFILENAME configuration parameter or to files in the subdirectory indicated by the TRACEPATHNAME configuration parameter.

For example, to set up a CLI/ODBC trace file that is written to disk after each trace entry:

```
[COMMON]
TRACE=1
TRACEFILENAME=E:\TRACES\CLI\MONDAY.CLI
TRACEFLUSH=1
```

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

TRACECOMM

Keyword Description:

Include information about each network request in the trace file.

db2cli.ini Keyword Syntax:

TRACECOMM = 0 | 1

Default Setting:

0 - No network request information is captured.

DB2 CLI/ODBC Settings Tab:

This keyword cannot be set using the CLI/ODBC Settings notebook. The db2cli.ini file must be modified directly to make use of this keyword.

Only Applicable when:

the CLI/ODBC TRACE option is turned on.

See Also:

"TRACE" on page 246, "TRACEFILENAME", "TRACEPATHNAME" on page 249, "TRACEFLUSH" on page 248

Usage Notes:

When TRACECOMM is set on (1) then information about each network request will be included in the trace file.

This option is only used when the TRACE CLI/ODBC option is turned on. See the TRACE for an example.

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

TRACEFILENAME**Keyword Description:**

File used to store the DB2 CLI/ODBC trace information.

db2cli.ini Keyword Syntax:

TRACEFILENAME = < Full file name >

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

Service

Only Applicable when:

the TRACE is turned on.

See Also:

"TRACE" on page 246, "TRACEFLUSH" on page 248, "TRACEPATHNAME" on page 249

Usage Notes:

If the file specified does not exist, then it will be created; otherwise, the new trace information will be appended to the end of the file.

If the filename given is invalid or if the file cannot be created or written to, no trace will occur and no error message will be returned.

This option is only used when the TRACE option is turned on. This will be done automatically when you set this option in the CLI/ODBC Configuration utility.

See the TRACE option for an example of using the various trace settings. The TRACEPATHNAME option will be ignored if this option is set.

DB2 CLI trace should only be used for debugging purposes. It will slow down the execution of the CLI/ODBC driver, and the trace information can grow quite large if it is left on for extended periods of time.

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

TRACEFLUSH

Keyword Description:

Force a write to disk after each CLI/ODBC trace entry.

db2cli.ini Keyword Syntax:

TRACEFLUSH = 0 | 1

Default Setting:

Do not write after every entry.

DB2 CLI/ODBC Settings Tab:

Service

Only Applicable when:

the CLI/ODBC TRACE option is turned on.

See Also:

“TRACE” on page 246, “TRACEFILENAME” on page 247,
“TRACEPATHNAME” on page 249

Usage Notes:

Set this option on (TRACEFLUSH = 1) to force a write to disk after each trace entry. This will slow down the trace process, but will ensure that each entry is written to disk before the application continues to the next statement.

This option is only used when the TRACE CLI/ODBC option is turned on. See the TRACE option for an example.

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

TRACEPATHNAME

Keyword Description:

Subdirectory used to store individual DB2 CLI/ODBC trace files.

db2cli.ini Keyword Syntax:

TRACEPATHNAME = < Full subdirectory name >

Default Setting:

None

DB2 CLI/ODBC Settings Tab:

Service

Only Applicable when:

the TRACE option is turned on.

Not Applicable when:

the TRACEFILENAME option is turned on.

See Also:

“TRACE” on page 246, “TRACEFILENAME” on page 247,
“TRACEFLUSH” on page 248

Usage Notes:

Each thread or process that uses the same DLL or shared library will have a separate DB2 CLI/ODBC trace file created in the specified directory.

No trace will occur, and no error message will be returned, if the subdirectory given is invalid or if it cannot be written to.

This option is only used when the TRACE option is turned on. This will be done automatically when you set this option in the CLI/ODBC Configuration utility.

See the TRACE option for an example of using the various trace settings. It will be ignored if the DB2 CLI/ODBC option TRACEFILENAME is used.

DB2 CLI trace should only be used for debugging purposes. It will slow down the execution of the CLI/ODBC driver, and the trace information can grow quite large if it is left on for extended periods of time.

(This option is contained in the Common section of the initialization file and therefore applies to all connections to DB2.)

TXNISOLATION

Keyword Description:

Set the default isolation level.

db2cli.ini Keyword Syntax:

TXNISOLATION = 1 | 2 | 4 | 8 | 32

Default Setting:

Read Committed (Cursor Stability)

DB2 CLI/ODBC Settings Tab:

Transaction

Only Applicable when:

the default isolation level is used. This keyword will have no effect if the application has specifically set the isolation level.

Usage Notes:

Sets the isolation level to:

1 = Read Uncommitted (Uncommitted read)

2 = Read Committed (Cursor stability) (default)

4 = Repeatable Read (Read Stability)

8 = Serializable (Repeatable read)

32 = (No Commit, DATABASE 2 for AS/400 only; this is similar to autocommit)

The words in parentheses are IBM's terminology for the equivalent SQL92 isolation levels. Note that *no commit* is not an SQL92 isolation level and is supported only on DB2 Universal Database for AS/400. Refer to the *SQL Reference* for more information on isolation levels.

This keyword is only applicable if the default isolation level is used. If the application has specifically set the isolation level then this keyword will have no effect.

UID

Keyword Description:

Define default user ID.

db2cli.ini Keyword Syntax:

UID = *userid*

Default Setting:

None

DB2 CLI/ODBC Settings Tab:
CLI/ODBC Settings General

Usage Notes:

The specified *userid* value is used if a *userid* is not provided by the application at connect time.

UNDERSCORE

Keyword Description:

Specify whether or not the underscore character "_" is to be used as a wildcard character.

db2cli.ini Keyword Syntax:

UNDERSCORE = 1 | 0

Default Setting:

"_" acts as a wildcard.

DB2 CLI/ODBC Settings Tab:

Optimization

Usage Notes:

This option allows you to specify whether the underscore character "_" is to be used as a wildcard character (matching any one character, including no character), or to be used as itself. This option only affects catalog function calls that accept search pattern strings.

- 1 = "_" acts as a wildcard (default)

The underscore is treated as a wildcard matching any one character or none. For example, if two tables are defined as follows:

```
CREATE TABLE "OWNER"."KEY_WORDS" (COL1 INT)
CREATE TABLE "OWNER"."KEYWORDS" (COL1 INT)
```

The DB2 CLI catalog function call that returns table information (SQLTables()) will return both of these entries if "KEY_WORDS" is specified in the table name search pattern argument.

- 0 = "_" acts as itself

The underscore is treated as itself. If two tables are defined as the example above, SQLTables() will return only the "KEY_WORDS" entry if "KEY_WORDS" is specified in the table name search pattern argument.

Setting this keyword to 0 can result in performance improvement in those cases where object names (owner, table, column) in the database contain underscores.

Note: This keyword only has an effect on DB2 common server versions prior to Version 2.1. The ESCAPE clause for the LIKE predicate can be used for subsequent versions and all other DB2 servers. For more information on the ESCAPE clause, refer to the *SQL Reference*.

WARNINGLIST

Keyword Description:

Specify which errors to downgrade to warnings.

db2cli.ini Keyword Syntax:

WARNINGLIST = " 'xxxxx', 'yyyyy', ..."

Default Setting:

Do not downgrade any SQLSTATEs.

DB2 CLI/ODBC Settings Tab:

Service

See Also:

"IGNOREWARNLIST" on page 227, "IGNOREWARNINGS" on page 226

Usage Notes:

Any number of SQLSTATEs returned as errors can be downgraded to warnings. Each must be delimited with single quotes, separated by commas, and in uppercase. The entire string must also be enclosed in double quotes. For example:

```
WARNINGLIST=" '01S02', 'HY090' "
```

This option can be used in conjunction with the IGNOREWARNINGS CLI/ODBC configuration keyword. If you also set IGNOREWARNINGS on then any errors you downgrade to warnings will not be reported at all.

Part 4. Configuring DB2 Connect to Host or AS/400 Communications

Chapter 14. Configuring Host Communications Using the Command Line Processor

This section describes how to configure a DB2 Connect workstation to communicate with a host or AS/400 database server.



If you plan to use an OS/2 or Windows 32-bit client to communicate with a server, the Client Configuration Assistant (CCA) makes it easy to automate configuration and administration tasks. If you have installed the CCA, it is recommended that you use this tool to configure your DB2 Connect workstation for communications.

Refer to your *DB2 Connect Quick Beginnings* book for more information.

For instructions on entering DB2 commands, see “Entering Commands Using the Command Center” on page 448 or “Entering Commands Using the Command Line Processor” on page 449.



If you need to manually configure communications, go to the section that describes your communication protocol.

- TCP/IP - see “Chapter 15. Manually Configuring TCP/IP Communications on the DB2 Connect Workstation” on page 257
 - APPC - see “Chapter 16. Manually Configuring APPC Communications on the DB2 Connect Workstation” on page 267.
-

Chapter 15. Manually Configuring TCP/IP Communications on the DB2 Connect Workstation

This section describes the manual configuration instructions for TCP/IP communications on a DB2 Connect workstation.

This section assumes that TCP/IP is functional on the DB2 Connect and host systems.

The following steps provide an overview on how to set up TCP/IP communications between your DB2 Connect workstation and a host database server:

- “1. Identify and Record Parameter Values” on page 258
- “2. Configure the DB2 Connect Workstation” on page 259
- “3. Catalog the TCP/IP Node” on page 261
- “4. Catalog the Database as a Database Connection Service (DCS) Database” on page 262
- “5. Catalog the Database” on page 262
- “6. Bind Utilities and Applications to the Database Server” on page 264
- “7. Test the Host or AS/400 Connection” on page 264



Due to the characteristics of the TCP/IP protocol, TCP/IP may not be immediately notified of the failure of a partner on another host. As a result, a client application accessing a remote DB2 server using TCP/IP, or the corresponding agent at the server, may sometimes appear to be hung. DB2 uses the TCP/IP SO_KEEPALIVE socket option to detect when there has been a failure and the TCP/IP connection has been broken.

If you are experiencing problems with your TCP/IP connection, refer to the *Troubleshooting Guide* for information on how to adjust this parameter and other common TCP/IP problems.

1. Identify and Record Parameter Values

As you proceed through the configuration steps, complete the *Your Value* column in the following table. You can fill in some of the values before you start configuring this protocol.

Table 26. TCP/IP Values Required at the DB2 Connect Workstation

Parameter	Description	Sample Value	Your Value
Host name <ul style="list-style-type: none"> • Hostname (<i>hostname</i>) or • IP address (<i>ip_address</i>) 	Use the <i>hostname</i> or <i>ip_address</i> of the remote host. To resolve this parameter: <ul style="list-style-type: none"> • Contact your network administrator to obtain the <i>hostname</i>. • Contact your network administrator to obtain the <i>ip_address</i> or enter the ping hostname command. 	nyx or 9.21.15.235	
Service Name <ul style="list-style-type: none"> • Connection Service name (<i>svcname</i>) or • Port number/Protocol (<i>port_number/tcp</i>) 	Values required in the services file. The Connection Service name is an arbitrary name that represents the Connection port number (<i>port_number</i>) on the client. The port number for the DB2 Connect workstation must be the same as the port number that the <i>svcname</i> parameter maps to in the services file at the host database server. (The <i>svcname</i> parameter is located in the database manager configuration file on the host.) This value must not be in use by any other applications, and must be unique within the services file. On UNIX platforms, this value generally must be 1024 or higher. Contact your database administrator for the values used to configure the host system.	host1 or 3700/tcp	

Table 26. TCP/IP Values Required at the DB2 Connect Workstation (continued)

Parameter	Description	Sample Value	Your Value
Target database name (<i>target_dbname</i>)	The database name as it is known on the host or AS/400 system. <ul style="list-style-type: none"> • If you are connecting to a DB2 for OS/390 system, use the location name. • If you are connecting to a DB2 for AS/400 system, use the local RDB name. • If you are connecting to a DB2 for VM or DB2 for VSE system, use the dbname. 	newyork	
Local database name (<i>local_dcsname</i>)	An arbitrary local nickname for use by DB2 Connect that represents the remote host or AS/400 database.	ny	
Node name (<i>node_name</i>)	A local alias, or nickname, that describes the node to which you are trying to connect. You can choose any name you want; however, all node name values within your local node directory must be unique.	db2node	

2. Configure the DB2 Connect Workstation

The steps in this section configure TCP/IP on the DB2 Connect Workstation. Replace the sample values with your worksheet values.

A. Resolve the Host's IP Address



If your network has a name server, or you are planning to directly specify the IP address (*ip_address*) of the server, skip this step and proceed to "B. Update the Services File" on page 260.

The DB2 Connect workstation must know the address of the host system to which it is attempting to establish communications. If a name server does not exist on your network, you may directly specify a hostname that maps to the IP address (*ip_address*) of the host system in the local hosts file. See Table 10 on page 50 for the location of the hosts file for your particular platform.



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

Table 27. Location of the Local Hosts and Services Files

Platform	Location
Windows 9x	windows directory
Windows NT and Windows 2000	winnt\system32\drivers\etc directory
UNIX	/etc directory
OS/2	Specified by the <i>etc</i> environment variable. Enter the set etc command to determine the location of your local hosts or services files. Note: For DOS and WIN-OS2 sessions, you might need to update the hosts and services files located in the tcpip_product\dos\etc directory.

Using a text editor, add an entry to the DB2 Connect workstation's hosts file for the host system's hostname. For example:

```
9.21.15.235    nyx    # host address for nyx
```

where:

9.21.15.235

represents the *ip_address*

nyx represents the *hostname*

represents a comment describing the entry

If the host system is not in the same domain as the DB2 Connect Workstation, you must provide a fully qualified domain name such as *nyx.spifnet.ibm.com*, where *spifnet.ibm.com* is the domain name.

B. Update the Services File



If you are planning to catalog a TCP/IP node using a port number (*port_number*), skip this step and go to "3. Catalog the TCP/IP Node" on page 261.

Using a text editor, add the Connection Service name and port number to the DB2 Connect workstation's services file. This file is located in the same directory as the local hosts file that you may have edited in "A. Resolve the Host's IP Address" on page 259. See Table 10 on page 50 for the location of the services file for your particular platform. For example:

```
host1 3700/tcp # DB2 connection service port
```

where:

host1 represents the Connection Service name

3700 represents the Connection port number
tcp represents the communication protocol that you are using
represents a comment describing the entry

The port number used on the DB2 Connect workstation must match the port number used on the host system. Also, ensure that you did not specify a port number that is being used by any other process.

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

3. Catalog the TCP/IP Node

You must add an entry to the DB2 Connect workstation's node directory to describe the remote node. This entry specifies the chosen alias (*node_name*), the *hostname* (or *ip_address*), and the *svcname* (or *port_number*) that the client will use to access the remote host.

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

- Step 1. Log on to the system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority.
- Step 2. If you are using DB2 Connect on a UNIX platform, set up the instance environment and invoke the DB2 command line processor. Run the start-up script as follows:

```
. INSTHOME/sql1lib/db2profile (for Bourne or Korn shell)  
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

- Step 3. Catalog the node by entering the following commands:

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

For example, to catalog the remote host *nyx* on the node called *db2node*, using the service name *host1*, enter the following:

```
catalog tcpip node db2node remote nyx server host1  
terminate
```

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

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



If you need to change values that were set with the **catalog node** command, perform the following steps:

Step 1. Run the **uncatalog node** command in the command line processor as follows:

```
db2 uncatalog node node_name
```

Step 2. Recatalog the node with the values that you want to use.

4. Catalog the Database as a Database Connection Service (DCS) Database

To catalog the remote database as a Data Connection Services (DCS) database, perform the following steps:

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

Step 2. Enter the following commands:

```
catalog dcs db local_dcsname as target_dbname
terminate
```

where:

- *local_dcsname* represents the local name of the host or AS/400 database.
- *target_dbname* represents the name of database on the host or AS/400 database system.

For example, to make *ny* the local database name for DB2 Connect, for the remote host or AS/400 database called *newyork*, enter the following commands:

```
catalog dcs db ny as newyork
terminate
```

5. Catalog the Database

Before a client application can access a remote database, the database must be cataloged on the host system node and on any DB2 Connect workstation nodes that will connect to it. When you create a database, it is automatically cataloged on the host with the database alias (*database_alias*) the same as the database name (*database_name*). The information in the database directory, along with the information in the node directory, is used on the DB2 Connect workstation to establish a connection to the remote database.

To catalog a database on the DB2 Connect Workstation, perform the following steps.

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

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

Table 28. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (<i>database_name</i>)	The local DCS database name (<i>local_dcsname</i>) of the remote database, you specified this when you catalogued the DCS database directory, for example, <i>ny</i> .	<i>ny</i>	
Database alias (<i>database_alias</i>)	An arbitrary local nickname for the remote database. If you do not provide one, the default is the same as the database name (<i>database_name</i>). This is the name that you use when connecting to the database from a client.	<i>localny</i>	
Node name (<i>node_name</i>)	The name of the node directory entry that describes where the database resides. Use the same value for the node name (<i>node_name</i>) that you used to catalog the node in the previous step.	<i>db2node</i>	

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

```
. INSTHOME/sql1lib/db2profile (for Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

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

```
catalog database database_name as database_alias at node node_name
authentication auth_value
```

For example, to catalog the DCS known database *ny* so that it has the local database alias *localny*, on the node *db2node*, enter the following commands:

```
catalog database ny as localny at node db2node
authentication dcs
```



If you need to change values that were set with the **catalog database** command, perform the following steps:

Step a. Run the **uncatalog database** command in the command line processor as follows:

```
uncatalog database database_alias
```

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

6. Bind Utilities and Applications to the Database Server

The steps you have just completed set up the DB2 Connect workstation to communicate with the host or AS/400 system. You must now bind the utilities and applications to the host or AS/400 database server. You need BINDADD authority to bind.

To bind the utilities and applications to the host or AS/400 database server, enter the following commands:

```
connect to dbalias user userid using password  
bind path@ddcsmvs.lst blocking all sqlerror continue  
messages mvs.msg grant public  
connect reset
```

For example:

```
connect to NYC3 user myuserid using mypassword  
bind path/bnd@ddcsmvs.lst blocking all sqlerror continue  
messages mvs.msg grant public  
connect reset
```

For more information about these commands, refer to the *DB2 Connect User's Guide*.

7. Test the Host or AS/400 Connection

When you have finished configuring the DB2 Connect workstation for communications, perform the following steps to test the connection on a remote database. You will need to connect to a remote database to test the connection.

Step 1. Start the database manager by entering the **db2start** command on the host database server (if it was not already started).

Step 2. Enter the following command in the DB2 Connect workstation's Command Center or command line processor to connect to the remote database:

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

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the host or AS/400 database server.

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

```
db2 "select tablename from syscat.tables"
```

When you are finished using the database connection, enter the **db2 connect reset** command to end the database connection.



You are now ready to start using DB2. For more advanced topics, refer to the *Administration Guide*.

Test the Host Connection

If the connection fails, check the following items:

At the host:

- ___ 1. The *db2comm* registry value includes the value `tcpip`.
- ___ 2. The services file was updated correctly.
- ___ 3. The service name (*svcname*) parameter was updated correctly in the database manager configuration file.
- ___ 4. The database was created and cataloged properly.
- ___ 5. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).
- ___ 6. The port number specified is not being used by any other process.

If there are problems starting a protocol's connection managers, a warning message appears and the error messages are logged in the `db2diag.log` file.

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

At the DB2 Connect workstation:

- ___ 1. If used, the services and hosts files were updated correctly.
- ___ 2. The node was cataloged with the correct hostname (*hostname*) or IP address (*ip_address*).
- ___ 3. The port number must match, or the service name must map to, the port number used on the host.

- ___ 4. The node name (*node_name*) that was specified in the database directory points to the correct entry in the node directory.
- ___ 5. The database was cataloged properly, using the hosts's database alias (*database_alias*). This database alias was cataloged when the database was created on the host, as the database name (*database_name*) on the DB2 Connect workstation.

If the connection still fails after you verify these items, refer to the *Troubleshooting Guide*.

Chapter 16. Manually Configuring APPC Communications on the DB2 Connect Workstation

This section describes how to manually configure a DB2 Connect workstation to communicate with a host or AS/400 database server using the APPC communication protocol. The instructions in this section assume that APPC is supported on the DB2 Connect and host or AS/400 machines.

You will only need to refer to the instructions in this section if you wish to manually configure your APPC connection to a host or AS/400 database. APPC can often be configured automatically using the Client Configuration Assistant (CCA). The following table lists the products that can be configured using the CCA:

Table 29. Products Configured Using the CCA

Products	Platform	Configured by the CCA?
IBM Personal Communications V4.2 and later	Windows 32-bit operating systems	Yes
IBM Communications Server (Server)	Windows NT and Windows 2000	Yes
IBM Communications Server (Client)	Windows 32-bit operating systems	No
IBM Communications Server	OS/2	Yes
RUMBA	Windows 32-bit operating systems	Yes
Microsoft SNA (Server)	Windows NT and Windows 2000	No
Microsoft SNA (Client)	Windows 32-bit operating systems	No

For more information about the communications requirements of your platform, see “Software Requirements” on page 4. See “Possible Client-to-Server Connectivity Scenarios” on page 9 for the protocols supported for communication between your particular client and server.

For more information about the CCA, the communications requirements of your platform, or the protocols supported for communication between your particular client and server, refer to your *Quick Beginnings* book.

The following steps are required to set up a DB2 Connect workstation to use APPC communications with a host or AS/400 database server:

- “1. Identify and Record Parameter Values”.
- “2. Update the APPC Profiles on the DB2 Connect Workstation” on page 271.
- “3. Catalog the APPC or APPN Node” on page 347.
- “4. Catalog the Database as a Database Connection Service (DCS) Database” on page 348.
- “5. Catalog the Database” on page 349.
- “6. Bind Utilities and Applications to the Database Server” on page 350.
- “7. Test the Host or AS/400 Connection” on page 351.

1. Identify and Record Parameter Values

Before you configure the DB2 Connect workstation, have your host-side administrator and LAN administrator fill in copies of the worksheet in Table 30 for *each* host or AS/400 database to which you want to connect.

After you fill in the entries in the *Your Value* column, you can use the worksheet to configure APPC communications for DB2 Connect. During the configuration process, replace the sample values that appear in the configuration instructions with your values from the worksheet, using the boxed numbers (for example, **1**) to relate the configuration instructions to the worksheet values.

The worksheet and configuration instructions supply suggested or sample values for required configuration parameters. For other parameters, use the communications program’s default values. If your network configuration is different from that used in the instructions, consult your Network Administrator for values that are appropriate to your network.

In the configuration instructions, the ***** symbol denotes entries that need to be changed but do not have a representation on the worksheet.

Table 30. Worksheet for Planning Host and AS/400 Server Connections

Ref.	Name at the DB2 Connect Workstation	Network or VTAM Name	Sample Value	Your Value
Network Elements at the Host				
1	Host name	Local network name	SPIFNET	
2	Partner LU name	Application name	NYM2DB2	
3	Network ID		SPIFNET	
4	Partner node name	Local CP or SSCP name	NYX	

Table 30. Worksheet for Planning Host and AS/400 Server Connections (continued)

Ref.	Name at the DB2 Connect Workstation	Network or VTAM Name	Sample Value	Your Value
5	Target database name (<i>target_dbname</i>)	OS/390 or MVS: LOCATION NAME VM/VSE: DBNAME AS/400: RDB name	NEWYORK	
6	Link name or mode name		IBMRDB	
7	Connection name (link name)		LINKHOST	
8	Remote network or LAN address	Local adapter or destination address	400009451902	
Network Elements at the DB2 Connect Workstation				
9	Network or LAN ID		SPIFNET	
10	Local control point name		NYX1GW	
11	Local LU name		NYX1GW0A	
12	Local LU alias		NYX1GW0A	
13	Local node or node ID	ID BLK	071	
14		ID NUM	27509	
15	Mode name		IBMRDB	
16	Symbolic destination name		DB2CPIC	
17	Remote Transaction program (TP) name		OS/390 or MVS: X'07'6DB (07F6C4C2') or DB2DRDA VM/VSE: AXE for VSE. The DB2 for VM db name, or X'07'6DB (07F6C4C2') for VM AS/400: X'07'6DB (07F6C4C2') or QCNTEDDM	

Table 30. Worksheet for Planning Host and AS/400 Server Connections (continued)

Ref.	Name at the DB2 Connect Workstation	Network or VTAM Name	Sample Value	Your Value
DB2 Directory Entries at the DB2 Connect Workstation				
19	Node name		db2node	
19	Security		program	
20	Local database name (<i>local_dcsname</i>)		ny	

For each server that you are connecting to, fill in a copy of the worksheet as follows:

1. For *network ID*, determine the network name of both the host and the DB2 Connect workstations (**1** , **3** , and **9**). Usually these values will be the same. For example, SPIFNET.
2. For the *partner LU name* (**2**), determine the VTAM application (APPL) name for OS/390, MVS, VSE, or VM. Determine the local CP name for AS/400.
3. For *partner node name* (**4**), determine the System Services Control Point (SSCP) name for OS/390, MVS, VM, or VSE. Determine the local control point name for an AS/400.
4. For *database name* (**5**), determine the name of the host database. This is the *LOCATION NAME* for OS/390 or MVS, the *DBNAME* for VM or VSE, or a relational database (RDB) name for AS/400.
5. For *mode name* (**6** and **15**), usually the default IBMDRB is sufficient.
6. For *remote network address* (**8**), determine the controller address or local adapter address of the target host or AS/400 system.
7. Determine the *local control point name* (**10**) of the DB2 Connect workstation. This is usually the same as the PU name for the system.
8. Determine the *local LU name* to be used by DB2 Connect (**11**). If you use a Syncpoint Manager to manage multisite updates (two-phase commit), the local LU should be the LU used for the SPM. In this case, that LU cannot also be the control point LU.
9. For *local LU alias* (**12**), you usually use the same value as for the local LU name (**11**).
10. For *local node* or *node ID* (**13** plus **14**), determine the IDBLK and IDNUM of the DB2 Connect workstation. The default value should be correct.
11. For *symbolic destination name* (**16**), choose a suitable value.
12. For (remote) *transaction program (TP) name* (**17**), use the defaults listed in the worksheet.
13. Leave the other items blank for now (**18** to **21**).

2. Update the APPC Profiles on the DB2 Connect Workstation

Use the completed worksheet in Table 30 on page 268 to configure DB2 Connect APPC communications for access to a remote host or AS/400 database server.



Go to the sections that describe how to configure APPC communications on the platforms that are present in your network:

- “Configuring IBM eNetwork Communications Server for OS/2”
 - “Configuring IBM eNetwork Communications Server for Windows” on page 295
 - “Configuring IBM eNetwork Communications Server for Windows NT SNA API Client” on page 300
 - “Configuring Microsoft SNA Server for Windows” on page 303
 - “Configuring Microsoft SNA Client” on page 312
 - “Configuring IBM eNetwork Communication Server for AIX” on page 315
 - “Configuring Bull SNA for AIX” on page 324
 - “Configuring SNAPPlus2 for HP-UX” on page 327
 - “Configuring SNAP-IX Version 6.0.1 for SPARC Solaris” on page 336
 - “Configuring SunLink 9.1 for Solaris” on page 344
-

Configuring IBM eNetwork Communications Server for OS/2

This section describes how to manually configure APPC communications between your DB2 Connect for OS/2 workstation and host or AS/400 database servers.

Before you begin, ensure that your workstation has CS/2 V5 or later for OS/2 installed. The steps in this section describe how to use IBM eNetwork Communications Server Version 5. If you have Communications Manager for OS/2 V.1.x, the steps you perform are similar, but the interface and menu names are different.

For more information on setting up your environment, refer to the online help supplied with CS/2, or to the following publications:

- *Connectivity Supplement*
- *DRDA Connectivity Guide*

The following assumptions have been made:

- The basic installation of the IBM eNetwork Communication Server V5 for OS/2 package has already been completed.
- The DB2 Client for OS/2 has been installed.

Use the *Your Value* entries from the worksheet in Table 25 on page 153 to complete the steps below.

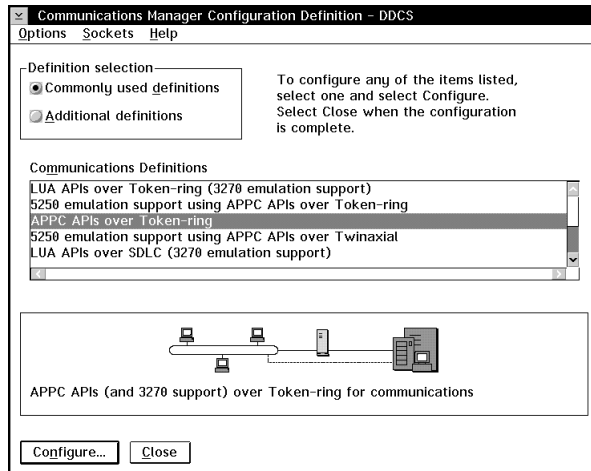


These instructions describe how to create new profiles within a new configuration. If you are modifying an existing configuration, you may need to delete some profiles before you can verify the configuration.

To configure your system, perform the following steps:

Step 1. Start a new configuration

- a. Double-click on the **IBM eNetwork Communications Server** icon.
- b. Double-click on the **Communications Manager Setup** icon.
- c. In the Communications Manager Setup window, click on the **Setup** push button.
- d. In the Open Configuration window, provide a name for a new configuration file and click on **OK**. The Communications Manager Configuration Definition window opens.



Step 2. Configure the protocol

- a. Select the **Commonly used definitions** radio button.
- b. In the Communications Definitions box, select the protocol that you want to use. These instructions use APPC APIs over Token-Ring.

- c. Click on the **Configure** push button. The APPC APIs over Token-Ring window opens.

APPC APIs over Token-ring

Network ID SPIFNET

Local node name NYX1

Local node type

Network node

End node - no network node server

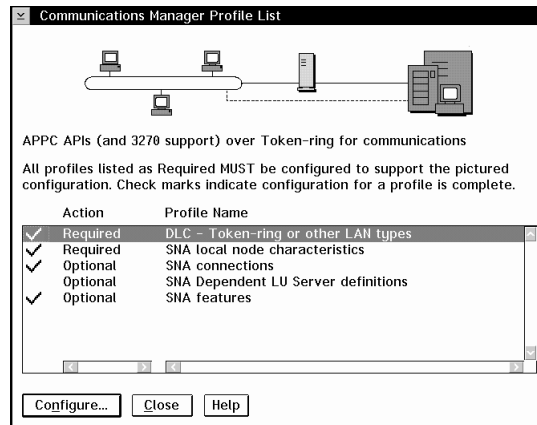
End node - to a network node server

Network node server address (hex)

OK Advanced... Cancel Help

- d. Enter your Network ID (**9**) in the **Network ID** field.
- e. Enter your local Control Point name (**10**) in the **Control point name** field.
- f. Click on the **End node** push button that your network administrator advises you to use. You can select either the **End node - to a network node server** radio button or the **End node - no network node server** radio button. A network node server is used when many users are routed through the same connection. This example assumes no network node server is used.
- g. Click on the **Advanced** push button. The subsequent steps begin from this window. You will return to this window when each of the following steps are completed. The Communication Manager

Profile List window opens.



Step 3. Prepare a LAN DLC profile

- a. On the Profile List window, select the **DLC - Token ring or other LAN Types Adapter Parameters** option and click on the **Configure** push button. The Token Ring or Other Lan Types Adapter Parameters window opens.

Token Ring or Other LAN Types DLC Adapter Parameters

Adapter (0 - 15)

Free unused links

Branch extender support

Maximum I-field size
 (265 - 16393)

Local SAP (hex)
 (04 - 9C)

Effective capacity (bits per second)

Connection network parameters (optional)
 Name . Limited resource

Additional parameters
 HPR parameters
 Link initialization parameters
 Link station protocol parameters
 Network management parameters
 Resource parameters
 Change...

OK Delete Cancel Help

- b. Enter your Network ID (**9**) in the **Network ID** field.
- c. Click on **OK**.

Step 4. Update the SNA local node characteristics

- a. On the Profile List window, select the **SNA local node characteristics** option and click on the **Configure** push button.

The Local Node Characteristic window opens.

Local Node Characteristics

Network ID SPIFNET

Local node name NYX1

Node type

End node

Network node

Branch extender support

Local node ID (hex) 05D 27509

Local node alias name NYX1

Maximum compression level NONE

Maximum compression tokens 0 (0 - 30400)

Activate Attach Manager at start up

Search required

Optional comment

OK NetWare(R)... Cancel Help

- b. Enter your Network ID (**9**) in the **Network ID** field.
- c. The Local node name was probably set when CS/2 was installed. If you are not sure, consult your local network administrator.
- d. Enter your Node ID (**13** , **14**) in the **Local node ID (hex)** field.



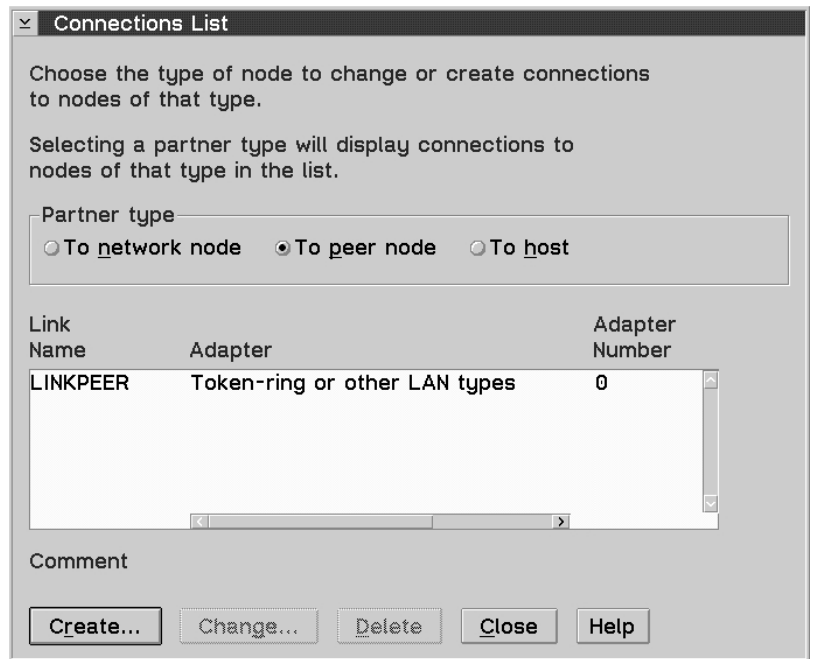
The first part should be filled in for you already when you display the profile. You only need to complete the second part.

- e. Click on **OK**.

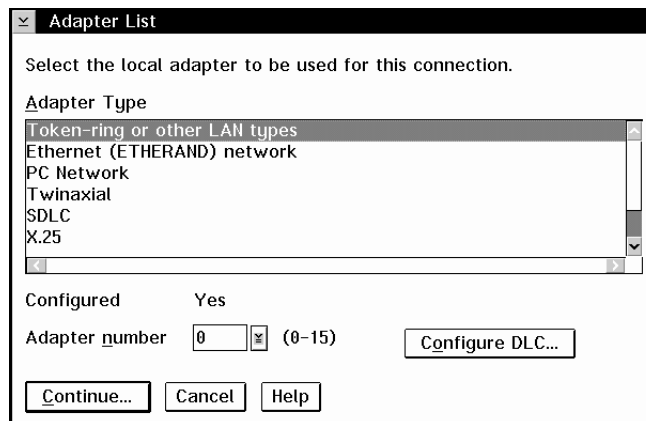
Step 5. Prepare the SNA Connection Profiles

- a. On the Profile List window, select the **SNA Connections** option and click on the **configure** push button. The Connection List

window opens.



- b. From the **Partner Type** window, select either the **To peer node** radio button (normally used for OS/400 connections), or the **To host** radio button (normally used for OS/390, MVS, VSE, and VM connections), and click on the **Create** push button. The Adapter List window opens.



- c. Select the **Token-ring, or other LAN types** Adapter Type, and specify the same adapter number that you specified previously in the DLC profile.

- d. Click on the **Continue** push button. The Connection to a Peer Node window or the Connection to a Host window opens.

- Step 6. Configure the connection in the Connection to a Peer Node or Connection to Host window
 - a. Enter the Link name (**7**) in the **Link name** field.
 - b. In the Connection window, click on the **Additional parameters** push button. The **Additional Connection Parameters** window opens.
 - c. Enter your local Control Point name (**10**) in the **Local PU name** field.
 - d. Clear the **Backup Link** check box.

- e. Enter your Node ID (**13** and **14**) in the Node ID fields.
- f. Click on **OK**.
- g. Enter the remote LAN address (**8**) in the **LAN destination address** field.
- h. Enter the Network ID (**1**) of the remote system in the **Partner network ID** field.
- i. Enter the Partner Node name (**4**) in the **Partner node name** field.
- j. Click on the **Define Partner LUs** push button. The Partner LU window opens.

Partner LUs

To add a Partner LU, enter the LU name, alias, and comment. Then select Add.

To change a Partner LU, select an LU from the list, change the LU name, alias, and/or comment fields and select Change.

To delete a Partner LU, select an LU from the list and select Delete.

Network ID: SPIFNET

LU name: NYM2DB2

Alias: NYM2DB2

LU name	Alias
SPIFNET.NYM2DB2	NYM2DB2

Change Delete

Dependent partner LU

Partner LU is dependent

Uninterpreted name: _____

Optional comment: _____

Add

OK Cancel Help

Step 7. Create a partner LUs profile

- a. Enter the Network ID (**3**) of the remote system in the **Network ID** field.
- b. Enter the Partner LU name (**2**) in the **LU name** and **Alias** fields.
- c. Click on the **Add** push button to add the partner LU profile to the connection profile.
- d. Click on **OK**.

- e. Click on the **Additional Parameters** push button. The Additional Connection Parameters window opens.

Additional Connection Parameters

Link name LINKHOST

Multiple PU parameters

Backup link PU name of primary host link

Local PU name NYX1

Local node ID (hex) 05D 27509

Host connection parameters

APPN support

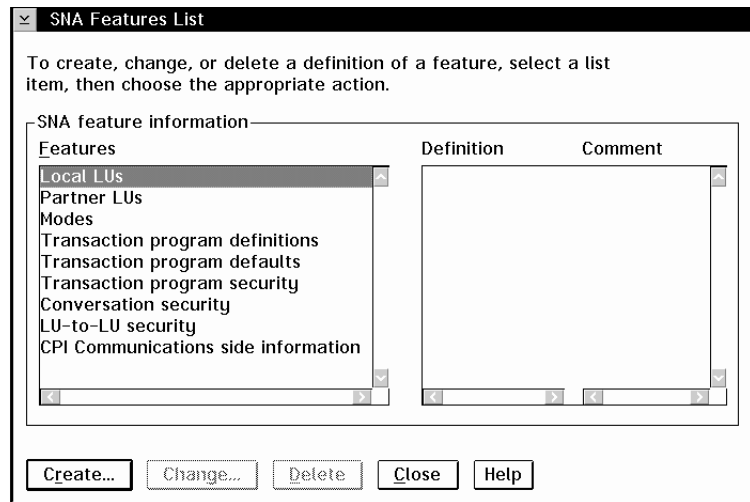
Use this host connection as your focal point support

Optional comment

OK Cancel Help

- f. Verify that the **Multiple PU Parameters** fields are filled in. This value is the Local Node ID in hex (**13** and **14**).
 - g. Click on **OK** to return to the Connection window.
 - h. Click on **OK** to return to the Connections List window.
 - i. Click on the **Close** push button to return to the Profile List window.
- Step 8. Set SNA features
- a. On the Profile List window, select the **SNA features** option and click on the **Configure** push button. The SNA Features List

window opens. Subsequent steps begin from this window.



Step 9. Prepare a local LU profile

If the DB2 workstation is defined as an independent LU, prepare a Local LU Profile by performing the following steps:

- On the SNA Features List window, select **Local LUs** → **Create** from the action menu bar.
- Enter your Local LU name (**11**) in the **LU name** field.
- Enter your Local LU alias (**13**) in the **alias** field.
- Select the **Independent LU** radio button in the **NAU address** box.
- Click on **OK**.
- To use this local LU when the DB2 Connect workstation starts the APPC connection, select the **Use this local LU as your default local LU alias** check box. By default, all APPC connections that are started from this DB2 Connect workstation will use this local LU.

Step 10. Prepare a mode definition

- a. From the SNA Features List box, select the **Modes** option and click on the **Create** push button. The Mode Definition window opens.

Mode Definition

Mode name: IBMRDB

Class of service: #CONNECT

Mode session limit: 8 (0 - 32767)

Minimum contention winners: 0 (0 - 32767)

Receive pacing window: 4 (0 - 63)

Pacing type: Adaptive

Compression and session-level encryption support: Setup...

RU size:

- Default RU size
- Maximum RU size (256 - 16384)

Optional comment:

OK Cancel Help

- b. Enter your Mode name (**6** , **15**) in the **mode name** field.
- c. For the other fields, you can either specify values that match the mode profile defined on your server systems, or tune the parameters.
- d. Click on **OK** to finish the creation of the mode and to return to the SNA Features List panel.

Step 11. Create the CPIC side information

- a. From the SNA Features List box, select the **CPI Communications Side Information** option and click the **Create** push button. The CPI Communications Side Information window opens.

- b. Enter the Symbolic destination name (**16**) in the **Symbolic destination name** field.
- c. Select the **Alias** radio button.
- d. Click on the **Alias** drop down box and select your Local LU alias (**12**).
- e. Enter the remote Transaction Program (TP) name (**17**) in the **Partner TP** field.
- f. Select the **None** radio button in the **Security type** group. You will specify the security type later when you update the DB2 directories.
- g. Enter the mode name (**6**) in the **Mode name** field.
- h. Click on **OK** to save the CPI side information profile and return to the SNA Features List panel.
- i. Click on **Close** to return to the Communications Server Profile List panel.

Step 12. Save the configuration

- a. Click on the **Close** button to return to the Communication Server Configuration Definition window.

- b. Click on the **Close** button to automatically verify and save the new configuration file, and leave the configuration windows.
- c. Stop and start Communications Server by selecting **Stop Communications Normally** → **Start Communications** from the action menu bar.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring IBM Personal Communications for Windows

This section describes how to configure IBM Personal Communications for Windows NT, Windows 2000, Windows 98 and Windows 95 on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC.

Before you begin, ensure that the IBM Personal Communications for Windows NT or Windows 9x you installed meets the following requirements:

- __ 1. It is Version 4.2 or higher
- __ 2. The IBM Personal Communications IEEE 802.2 LAN interface is installed (this is an installation option for IBM Personal Communications)
- __ 3. The LLC2 driver is installed from the IBM Communications Server installation directory. To confirm this on Windows NT or 9x:
 - a. Click on the **Start** push button, then select **Settings** → **Control Panel**.
 - b. Double-click on the **Network** icon.
 - c. On the Network window, click on the **Protocols** tab. **IBM LLC2 Protocol** must be one of the protocols listed. If it is not, you need to install this protocol from your IBM Personal Communications for Windows NT or Windows 9x software. Refer to the documentation supplied with IBM Personal Communications for instructions.

For Windows 2000:

- a. Click on the **Start** push button and then select **Settings** → **System**
- b. Click the **Network and Dial-up Connections** and select the connection you want to configure (for example, Local Area Network).

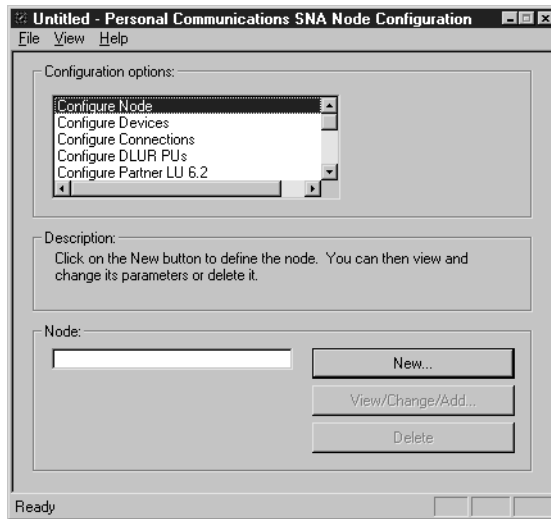
- c. From the **General** tab, select the **Properties** push button. Verify that **IBM LLC2 Protocol** is one of the protocols listed. If it is not, you need to install this protocol from your IBM Personal Communications software. Refer to its documentation for instructions.

The following assumptions are made:

- The basic installation of the IBM Personal Communication package has already been completed, with the requirements listed above met.
- DB2 Connect has been installed.

To start IBM Personal Communications, perform the following steps:

- Step 1. Click **Start** and select **Programs** —> **IBM Communications Server** —> **SNA Node Configuration**. The IBM Personal Communications SNA Node Configuration window opens.



- Step 2. Select **File** —> **New** from the menu bar. The Define the Node window opens. Subsequent steps will begin from this window.

To configure APPC communications, complete the following steps:

- Step 1. Configure the Node
 - a. In the **Configuration options** box, select **Configure Node**, then click on the **New** push button. The Define the Node window

opens.

The screenshot shows a dialog box titled "Define the Node" with a close button (X) in the top right corner. It has three tabs: "Basic", "Advanced", and "DLU Requester". The "DLU Requester" tab is selected. The dialog is divided into two main sections: "Control Point (CP)" and "Local Node ID".

Control Point (CP) section:

- Fully qualified CP name: Two text boxes containing "SPIFNET" and "NYX1" separated by a period.
- CP alias: A single empty text box.

Local Node ID section:

- Block ID: A text box containing "05D".
- Physical Unit ID: A text box containing "27509".

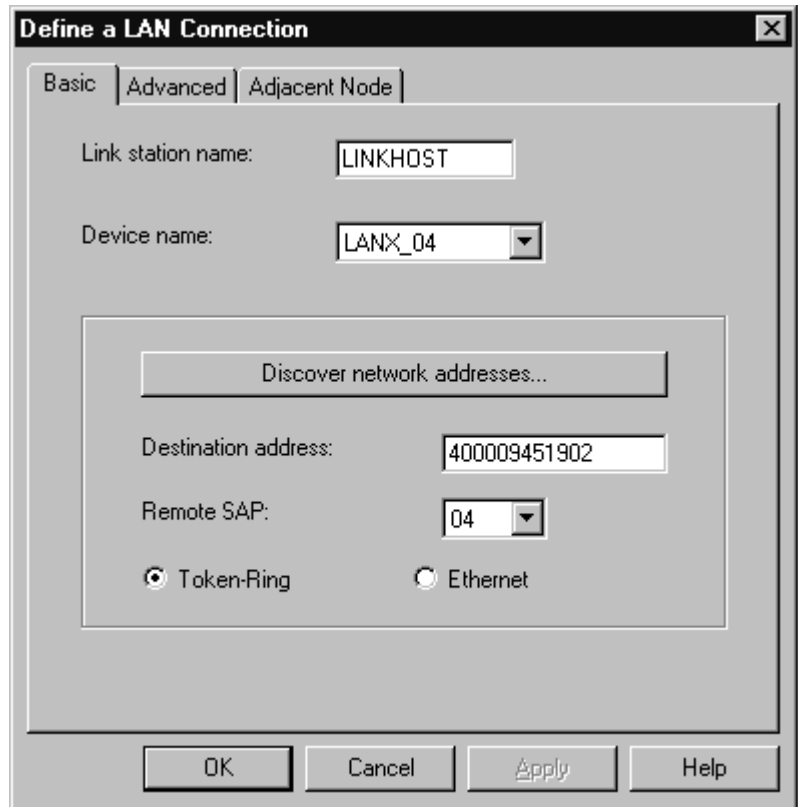
At the bottom of the dialog are four buttons: "OK", "Cancel", "Apply", and "Help".

- b. In the **Fully qualified CP name** fields, type in the network name **9** and the local control point name **10** (SPIFNET.NYX1).
 - c. Optionally, in the **CP alias** field, type in a CP alias. If you leave this blank the local control point name **10** will be used (NYX1).
 - d. In the **Local Node ID** fields, type in the block ID **13** and the physical unit ID **14** (05D 27509).
 - e. Click on **OK**.
- Step 2. Configure the device
- a. In the **Configuration options** box, select **Configure devices**.
 - b. Select the appropriate DLC from **DLCs** field. These instructions use the **LAN DLC**.

- c. Click on the **New** push button. The appropriate window opens with default values displayed. In this case, the Define a LAN device window opens.
- d. Click on **OK** to accept the default values.

Step 3. Configure Connections

- a. In the **Configuration options** box, select **Configure connections**.
- b. Ensure that in the **DLCs** field, **LAN** is highlighted.
- c. Click on the **New** push button. The Define a LAN connection window opens.



- d. On the **Basic** tab panel:
 - 1) In the **Link station name** field, type in the name **7** from the worksheet (LINKHOST).
 - 2) In the **Destination address** field, type in the address **8** from the worksheet (400009451902).

- e. On the Adjacent Node tab panel:

The screenshot shows a dialog box titled "Define a LAN Connection" with a close button (X) in the top right corner. It has three tabs: "Basic", "Advanced", and "Adjacent Node", with "Adjacent Node" being the active tab. The dialog contains the following fields and controls:

- Adjacent CP name:** Two text input fields. The first contains "SPIFNET" and the second contains "NYX".
- Adjacent CP type:** A dropdown menu currently showing "Back-Level LEN".
- TG number:** A dropdown menu currently showing "0".
- Adjacent node ID:** A large rectangular box containing two sub-fields:
 - Block ID:** An empty text input field.
 - Physical Unit ID:** An empty text input field.

At the bottom of the dialog are four buttons: "OK", "Cancel", "Apply", and "Help".

- 1) In the **Adjacent CP name** fields, type in the network ID **3** and the Control Point name **4** (SPIFNET.NYX).
- 2) In the **Adjacent CP type** field, select **Back-level LEN**.
- 3) Ensure that **TG number** is set to 0 (the default).
- 4) Click on **OK**.

Step 4. Configure Partner LU 6.2

- a. In the **Configuration options** box, select **Configure partner LU**, then click on the **New** push button. The Define a partner LU 6.2

window opens.

Define a Partner LU 6.2

Basic | Advanced

Partner LU name:
SPIFNET . NYM2DB2

Partner LU alias:
NYM2DB2

Fully qualified CP name:
SPIFNET . NYM

OK Cancel Apply Help

- b. In the **Partner LU name** fields, type in the network ID **3** and the partner LU name **2** (SPIFNET.NYM2DB2).
- c. In the **Partner LU alias** field, type in the partner LU name **2** from the worksheet (NYM2DB2).
- d. In the **Fully-qualified CP name** fields, type in the network ID **3** and the adjacent control point SSCP name **4** (SPIFNET.NYX). Accept the defaults in the **Advanced** tab panel.
- e. Click on **OK**.

Step 5. Configure Modes

- a. In the **Configuration options** box, select **Configure modes**, then click on the **New** button. The Define a mode window opens.

The screenshot shows a window titled "Define a Mode" with a close button (X) in the top right corner. It has two tabs: "Basic" (selected) and "Advanced". Inside the "Basic" tab, there are three text input fields: "Mode name:" containing "IBMRDB", "PLU mode session limit:" containing "32", and "Minimum contention winner sessions:" containing "16". At the bottom of the window are four buttons: "OK", "Cancel", "Apply", and "Help".

- b. Enter your Mode name (**15**) in the **Mode name** field of the **Basic** tab.

- c. Select the **Advanced** tab.

The screenshot shows a dialog box titled "Define a Mode" with a close button (X) in the top right corner. It has two tabs: "Basic" and "Advanced", with "Advanced" selected. The dialog contains the following fields and options:

- Maximum negotiable session limit: 128
- Receive pacing window size: 1
- Class of Service name: #CONNECT (dropdown menu)
- Use cryptography
- Use compression
- Use default RU size
- Maximum RU size: 4096

At the bottom, there are four buttons: OK, Cancel, Apply, and Help.

- d. Select **#CONNECT** from the **Class of Service Name** field.
e. Click on **OK**.

Step 6. Configure Local LU 6.2

- a. In the **Configuration options** box, select **Configure local LU 6.2**, then click on the **New** button. The Define a local LU 6.2 window opens.

The screenshot shows a dialog box titled "Define a Local LU 6.2". It has a "Basic" tab selected. The fields are as follows:

- Local LU name: NYX1G\W01
- Dependent LU:
- Local LU alias: (empty text box)
- PU name: (dropdown menu)
- NAU address: (dropdown menu)
- LU session limit: 30

Buttons at the bottom: OK, Cancel, Apply, Help.

- b. Enter your Local LU name (**11**) in the **Local LU name** field.
- c. Type in a value for the **LU session limit** field. The default, 0, specifies the maximum allowed value.
- d. Accept the defaults for the other fields and click on **OK**.

Step 7. Configure CPI-C Side Information

- a. In the **Configuration options** box, select **Configure CPI-C side information**, then click on the **New** push button. The Define

CPI-C side information window opens.



- b. In the **Symbolic destination name** field, type in the name **16** from the worksheet (DB2CPIC).
- c. In the **Mode name** field, type in the name **15** from the worksheet (IBMRDB).
- d. In the **Partner LU Name** fields, type the Net ID **3** in the first field, and the Partner LU Name **2** (SPIFNET.NYM2DB2) in the second field.
- e. Specify the TP name. In the **TP name** field:
 - To specify a non-service TP, in the **TP name** field, type in the name of the non-service TP, for example DB2DRDA, and ensure that there is **no** check mark in the **Service TP** check box.
 - To specify a service TP, in the **TP name** field, type in the name of the service TP, for example 076DB, and ensure that there is a check mark in the **Service TP** check box.
- f. Accept the defaults for the other fields and click on **OK**.

Step 8. Save the Configuration

- a. Select **File** → **Save As**. The Save As window opens.
- b. Type in a file name, for example ny3.acg, then click on **OK**.
- c. In the dialog box that appears, you are asked if you want this configuration to be the default. Click on the **Yes** button.

Step 9. Update the environment

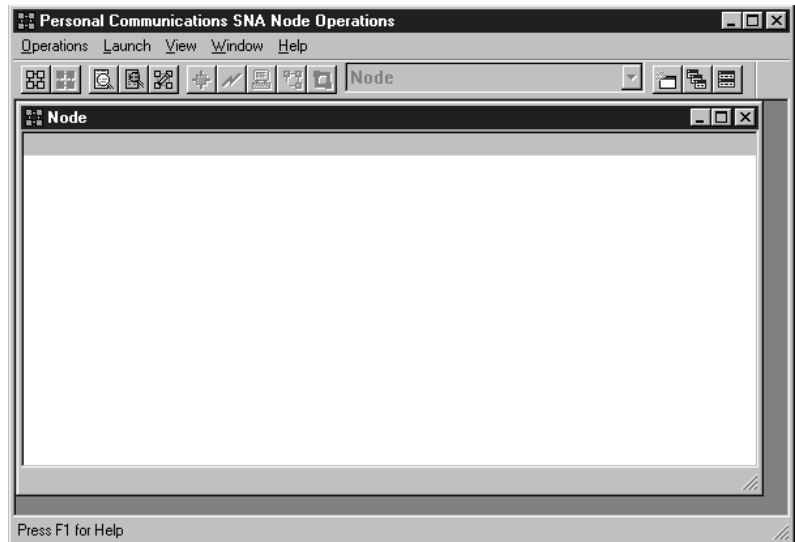
IBM Personal Communications uses an environment variable called **appc11u** to set the default Local LU used for APPC communications. You may set this variable on a per-session basis by opening a command window and typing `set appc11u=local_lu_name`, where *local_lu_name* is the name of the local LU you want to use. However, you will probably find it more convenient to permanently set the variable. To permanently set the variable in Windows NT, perform the following steps:

- a. Click on the **Start** push button and select **Settings** → **Control Panel**.
- b. Double-click on the System icon. The System Properties window opens.
- c. Select the **Environment** tab.
- d. Type `appc11u` in the **Variable** field.
- e. Type your local LU name (**4**) in the **Value** field.
- f. Click on the **Set** push button to accept the changes.
- g. Click on **OK** to exit the System Properties window.

The environment variable will now remain set for future sessions.

Step 10. Start SNA Node Operations

- a. Click on the **Start** push button and select **Programs** → **IBM Personal Communications** → **Administrative and PD Aids** → **SNA Node Operations**. The Personal Communications SNA Node Operations window opens.



- b. From the menu bar, select **Operations** → **Start Node**.

- c. In the window that opens, select the configuration file you saved in the previous step (for example, ny3.acg) and click on **OK**.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring IBM eNetwork Communications Server for Windows

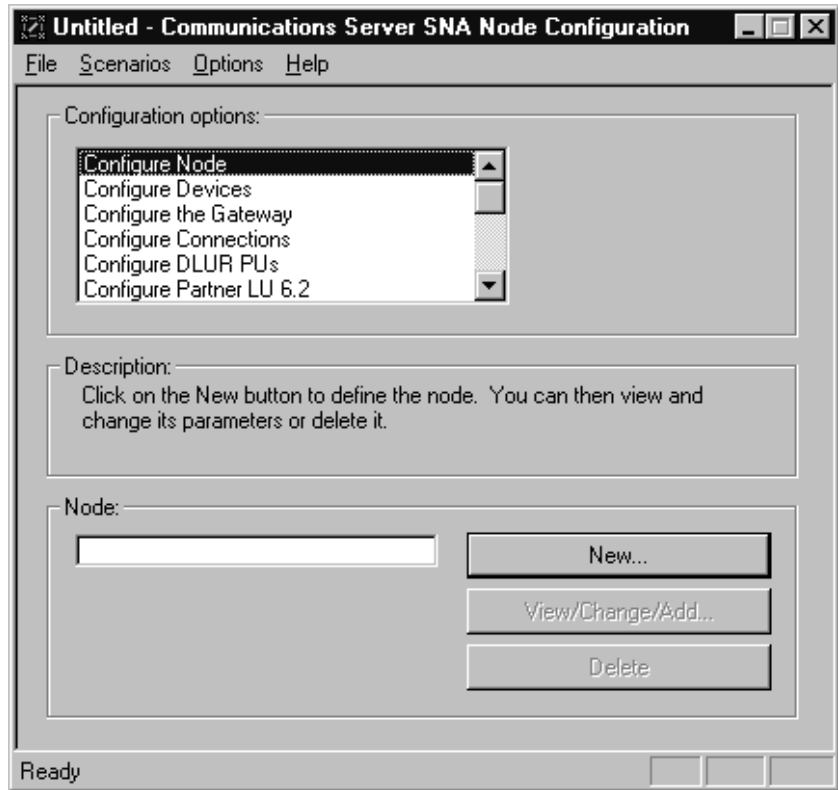
Before you begin, ensure that the IBM eNetwork Communications Server for Windows (supported on Windows NT and Windows 2000) you installed meets the following requirements:

- ___ 1. It is Version 5.0 or higher if you are planning to update multiple databases within the same transaction; if you are planning to use 2-phase commit then Version 5.01 of Communications Server for Windows is required
- ___ 2. The IBM Communications Server IEEE 802.2 LAN interface was installed (this is an installation option for Communications Server)
- ___ 3. The JR11529 and JR11170 APAR fixes have been applied. These fixes are required to enable cancelling of queries in progress by using Ctrl-BREAK or issuing the SQLCancel ODBC/CLI call.
- ___ 4. The LLC2 driver was installed from the Communications Server for Windows installation directory. During the installation, Communications Server for Windows asks if you want to install LLC2. If you are not sure whether LLC2 was installed with your copy of Communications Server for Windows, you can find out as follows:
 - a. Click **Start** and select **Settings** —> **Control Panel**.
 - b. Double-click on the **Network** icon.
 - c. On the Network window, click on the **Protocols** tab. **IBM LLC2 Protocol** must be one of the protocols listed. If it is not, you need to install this protocol from your IBM Communications Server for Windows software. For more information, refer to documentation supplied with Communications Server for Windows.

To start IBM eNetwork Personal Server, perform the following steps:

- Step 1. Click on **Start** and select **Programs** —> **IBM Communications Server** —> **SNA Node Configuration**. The IBM Communications Server SNA Node Configuration window opens.

Step 2. Select **File** → **New** → **Advanced** from the menu bar. The **Configuration options** window opens. Subsequent steps will begin from this window.



To configure IBM eNetwork Personal Server for APPC communications, complete the perform the following steps:

Step 1. Configure Node

- a. In the **Configuration options** field, select **Configure Node**, then click on the **New** push button. The Define the Node window opens.
- b. In the **Fully qualified CP name** fields, type in the network name (**9**) and the local control point name (**10**) (SPIFNET.NYX1).
- c. Optionally, in the **CP alias** field, type in a CP alias. If you leave this blank the local control point name (**10**) will be used (NYX1).
- d. In the **Local Node ID** fields, type in the block ID (**13**) and the physical unit ID (**14**) (05D.27509).
- e. Select the appropriate node type. The default is to select the **End Node** radio button.

- f. Click on **OK**.

Step 2. Configure Devices

- a. In the **Configuration options** field, select **Configure devices**.
- b. Ensure that in the **DLCs** field, the appropriate DLC is highlighted. For example, **LAN**.
- c. Click on the **New** push button. The appropriate window opens with default values displayed. For example, the Define a LAN device window.
- d. Click on **OK** to accept the default values.

Step 3. Configure the Gateway



You need to perform this step only if you are setting up Communications Server to accept requests from Communications Server for Windows SNA Client, as described in the your *DB2 Connect Quick Beginnings* manual.

- a. In the **Configuration options** field, select **Configure the Gateway**, then click on the **New** push button. The Define Gateway window opens.
- b. Click the **SNA Clients** tab.
- c. Ensure that there is a check mark in the **Enable SNA API Client Services** check box.
- d. Click **OK** to accept the default values.

Step 4. Configure Connections

- a. In the **Configuration options** field, select **Configure connections**.
- b. Ensure that **LAN** is highlighted in the **DLCs** field.
- c. Click on the **New** push button. The Define a LAN Connection window opens.
- d. On the **Basic** tab panel:
 - 1) In the **Link station name** field, type in the name (**7**) from the worksheet (LINKHOST).
 - 2) In the **Destination address** field, type in the address (**8**) from the worksheet (400009451902).
- e. On the **Security** tab panel:
 - 1) In the **Adjacent CP name** fields, type in the network ID (**3**) and the Control Point name (**4**) (SPIFNET.NYX).
 - 2) In the **Adjacent CP type** field, select the appropriate CP type (for example, **Back-level LEN**).
 - 3) Ensure that **TG number** is set to 0 (the default).
 - 4) Click **OK**.

Step 5. Configure Partner LU 6.2

- a. In the **Configuration options** field, select **Configure partner LU**, then click on the **New** button. The Define a Partner LU 6.2 window opens.
- b. In the **Partner LU name** fields, type in the network ID (**3**) and the Partner LU name (**2**) (SPIFNET.NYM2DB2).
- c. In the **Partner LU alias** field, type in the Partner LU name (**2**) from the worksheet (NYM2DB2).
- d. If you are configuring Communications Server for SNA Clients, in the **Fully-qualified CP name** fields, type in the Network ID (**3**) and the adjacent control point SSCP name (**4**) (SPIFNET.NYX).
- e. Leave the other fields blank. and click on **OK**.

Step 6. Configure Modes

- a. In the **Configuration options** field, select **Configure modes**, then click on the **New** button. The Define a Mode window opens.
- b. In the **Mode name** field (**6**), type in the mode name (IBMRDB).
- c. Click on the **Advanced** tab and ensure that the **Class of Service Name** is set to #CONNECT.
- d. Accept the defaults for the other fields and click on **OK**.

Step 7. Configure Local LU 6.2

- a. In the **Configuration options** field, select **Configure local LU 6.2**, then click on the **New** button. The Define a Local LU 6.2 window opens.
- b. In the **Local LU name** field, type in the name (**11**) from the worksheet (NYX1GW01).
- c. Enter a value for the **LU session limit** field. The default, 0, specifies the maximum allowed value.
- d. Accept the defaults for the other fields and click on **OK**.

Step 8. Configure CPI-C Side Information

- a. In the **Configuration options** field, select **Configure CPI-C Side Information**, then click on the **New** button. The Define CPI-C Side Information window opens.
- b. In the **Symbolic destination name** field, type in the name (**16**) from the worksheet (DB2CPIC).
- c. In the **Mode name** field, enter in the name (**15**) from the worksheet (IBMRDB).
- d. Click on the radio button beside **Use Partner LU alias** and select a Partner LU alias.
- e. Specify the TP name. In the **TP name** field:
 - To specify a non-service TP, in the **TP name** field, enter the name of the non-service TP, for example DB2DRDA, and ensure that there is *no* check mark in the **Service TP** check box.

- To specify a service TP, in the **TP name** field, enter the name of the service TP, for example 076DB, and ensure that there is a check mark in the **Service TP** check box.

f. Accept the defaults for the other fields and click on **OK**.

Step 9. Save the Configuration

- Select **File** → **Save as** from the menu bar. The Save As window opens.
- Enter a file name, for example ny3.acg
- Click on **OK**.
- In the window that opens you are asked if you want this configuration to be the default. Click on the **Yes** push button.

Step 10. Update the Environment

IBM Personal Communications uses an environment variable called **appclu** to set the default APPC Local LU. You may set this variable on a per-session basis by opening a command window and typing `set appclu=local_lu_name`, however you will probably find it more convenient to permanently set the variable. In order to permanently set the variable in Windows NT, complete the following steps:

Step a. Click on **Start** and select **Settings** → **Control Panel**. Double-click on the **System** icon. When the **System Properties** window opens, select the **Environment** tab.

Step b. Enter `appclu` in the **Variable** field, and enter your Local LU name (**11**) in the **Value** field.

Step c. Click **Set** to accept the changes then click **OK** to exit the System Properties window.

The environment variable will now remain set for future sessions.

Step 11. Start SNA Node Operations

To start SNA node operations on your machine, complete the following steps:

Step a. Click on **Start** and select **Programs** → **IBM Communications Server** → **SNA Node Operations**. The **SNA Node Operations** window opens.

Step b. From the menu bar, click on **Operations** and select **Start Node**. In the dialog box that opens, select the configuration file you saved at the end of Step 2 (in our example, ny3.acg). Click **OK**.

SNA node operations will now begin running.

Step 12. Registering Communications Server as a Windows Service

To automatically start Communications Server when the machine is booted, you can register it as a Windows service.

To register Communications Server as a Windows service execute one of the following commands:

```
csstart -a
```

to register Communications Server with the default configuration, or:

```
csstart -a c:\ibmcs\private\your.acg
```

where `c:\ibmcs\private\your.acg` represents the fully qualified name of the non-default Communications Server configuration file you want to use.

Whenever your machine is booted in the future, Communications Server will be started automatically with the required configuration file.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring IBM eNetwork Communications Server for Windows NT SNA API Client

Read this section if you have a Windows NT workstation that has IBM eNetwork Communications Server for Windows NT SNA API Client Version 5.0 or higher installed and you want to connect to an IBM eNetwork Communications Server for Windows NT Server.

The Communications Server for Windows NT Server and its SNA API client act as a split client. This configuration requires that you have an APPC-enabled application (such as DB2 Connect) running on the SNA API client workstation.



The instructions in this section use a Windows NT client. The instructions for other supported operating systems are similar; refer to your Communications Server for Windows NT documentation for more information.

To configure the Windows NT SNA API client for APPC communications, complete the following steps:

- Step 1. Create a user account for the SNA API client on the Communications Server for Windows NT Server
- a. Click **Start** and select **Programs** → **Administrative Tools (Common)** → **User Manager**. The User Manager window opens.
 - b. Select **Users** → **New User** from the menu bar. The New User window opens.
 - c. Fill in the fields for the new SNA client user account. For more information, refer to the Windows NT online help.
 - d. Ensure that this user account is a member of the *Administrators*, *IBMCSADMIN*, and *IBMCSAPI* groups:
 - 1) Click **Groups**.
 - 2) Select a group from the **Not member of** box and click <- **Add**. Repeat this step for each group that your user account must belong to.
 - e. Click **OK**.
 - f. Click **Add**.
- Step 2. Start the configuration GUI for the IBM eNetwork CS/NT SNA API Client. Click **Start** and select **Programs** → **IBM Communications Server SNA Client** → **Configuration**.

The CS/NT SNA Client Configuration window opens.



Step 3. Configure Global Data

- a. In the **Configuration options** box, select the **Configure Global Data** option and click on the **New** push button. The Define Global Data window opens.
- b. Enter the user name for the SNA API client in the **User name** field. This is the user name that was defined in Step 1.
- c. Enter the password for the user account in the **Password** and **Confirm Password** fields.
- d. Click **OK**.

Step 4. Configure APPC Server List

- a. In the **Configuration options** box, select the **Configure APPC Server List** option. Click on the **New** push button. The Define APPC Server List window opens.
- b. Type in the IP address of the server (for example, 123.123.123.123).
- c. Click **OK**.

Step 5. Configure CPI-C Side Information

- a. In the **Configuration options** box, select the **Configure CPI-C side information** option and click **New**. The Define CPI-C side information window opens.
- b. Enter the symbolic destination name (**16**) in the **Symbolic destination name** field.
- c. Enter your Local LU alias (**12**) in the **Local LU alias** field.
- d. Enter the mode name (**15**) in the **Mode name** field.
- e. Enter the transaction program name (**17**) in the **TP name** field.
- f. Select the **For SNA API Client use** check box for this transaction program.
- g. Enter the network ID (**3**) and partner LU name (**2**) in the **Partner LU name** field.
- h. Click **OK**.

Step 6. Save the Configuration

- a. Select **File** → **Save As** from the menu bar. The Save As window opens.
- b. Enter a file name, and click **Save**.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring Microsoft SNA Server for Windows

This section describes how to configure Microsoft SNA Server Version 4.0 for Windows NT on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC. Although Microsoft SNA Server will run on Windows NT 4.0 Workstation, Windows NT 4.0 Server is recommended.

If you wish to use DB2’s Multisite Update feature with this product, your minimum requirement is Microsoft SNA Server Version 4 Service Pack 3. For more information, refer to “Chapter 17. Enabling Multisite Updates (Two-Phase Commit)” on page 353.

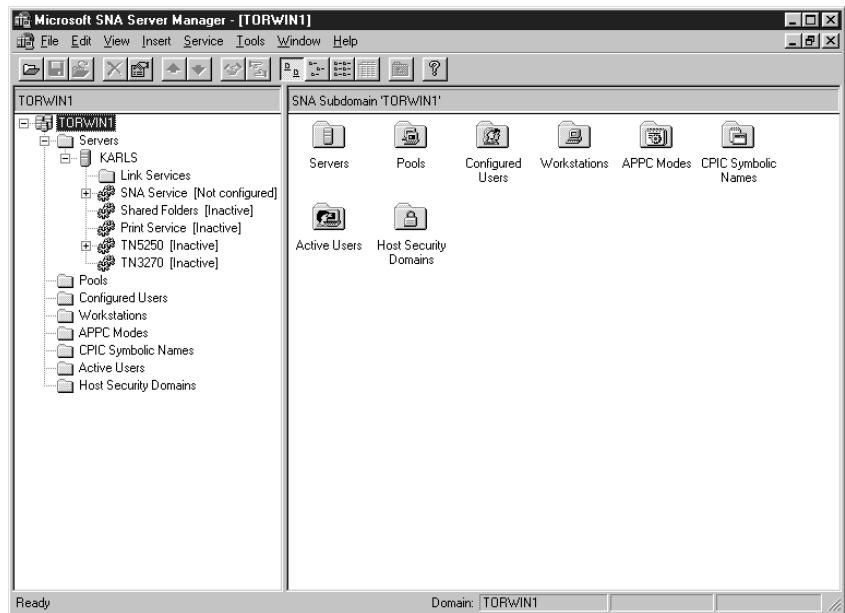


For instructions on how to configure Microsoft SNA Client for Windows, go to “Configuring Microsoft SNA Client” on page 312.

You can define the properties of your SNA connections in the Microsoft SNA Server Manager (Server Manager). The Server Manager uses an interface similar to that of the Windows NT Explorer. The following illustration shows the interface. There are two panes in the main window of the Server Manager. All the configuration options we will be using can be accessed by right-clicking on objects in the left-hand pane of the window. Every object has a *context menu* that you can access by right-clicking on the object.

To configure APPC communications for use by DB2 Connect using Microsoft SNA Server Manager, perform the following steps:

- Step 1. To start the Server Manager, click **Start** and select **Programs** —> **Microsoft SNA Server** —> **Manager**. The Microsoft SNA Server Manager window opens.

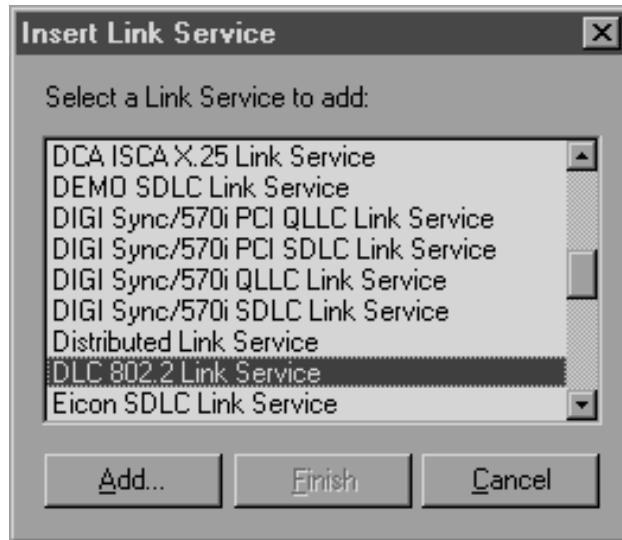


- Step 2. Define the control point name
 - a. Click on the [+] sign beside the **Servers** folder.

- b. Right-click on **SNA Service** folder and select the **Properties** option. The Properties window opens.
- c. Enter the correct **NETID** (**9**) and **Control Point Name** (**10**) in the corresponding fields.
- d. Click **OK**.

Step 3. Define the link service (802.2)

- a. Right-click on the **SNA Service** icon and select the **Insert** → **Link Service** option. The Insert Link Service window opens.



- b. Select **DLC 802.2 Link Service**.
- c. Click **Add**.
- d. Click **Finish**.

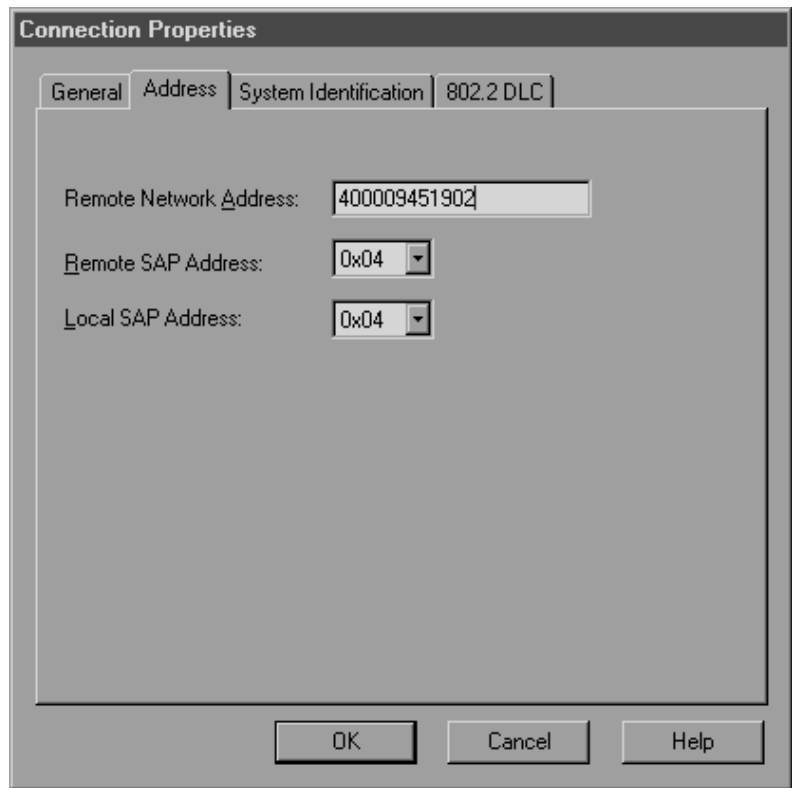
Step 4. Define the connection properties

- a. Right-click on **SNA Service** and select the **Insert** → **Connection** → **802.2** option. The Connection Properties window opens.

The screenshot shows the 'Connection Properties' dialog box with the '802.2 DLC' tab selected. The 'Name' field contains 'LINKHOST'. The 'Link Service' dropdown is set to '<None>'. The 'Comment' field is empty. Under 'Remote End', 'Host System' is selected. Under 'Allowed Directions', 'Both Directions' is selected. Under 'Activation', 'On Server Startup' is selected. The 'Passthrough via Connection' dropdown is set to '<None>'. The 'Supports Dynamic Remote APPC LU Definition' checkbox is unchecked. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

- b. Enter a connection name (**7**) in the **Name** field.
- c. Click on the **Link Service** drop-down box and select the **SnaDlc1** option.
- d. Select the **Remote End** radio button from the **Host System** box.
- e. Select the **Both Directions** radio button from the **Allowed Directions** box.
- f. Select the **On Server Startup** radio button from the **Activation** box.

- g. Select the **Address** tab.

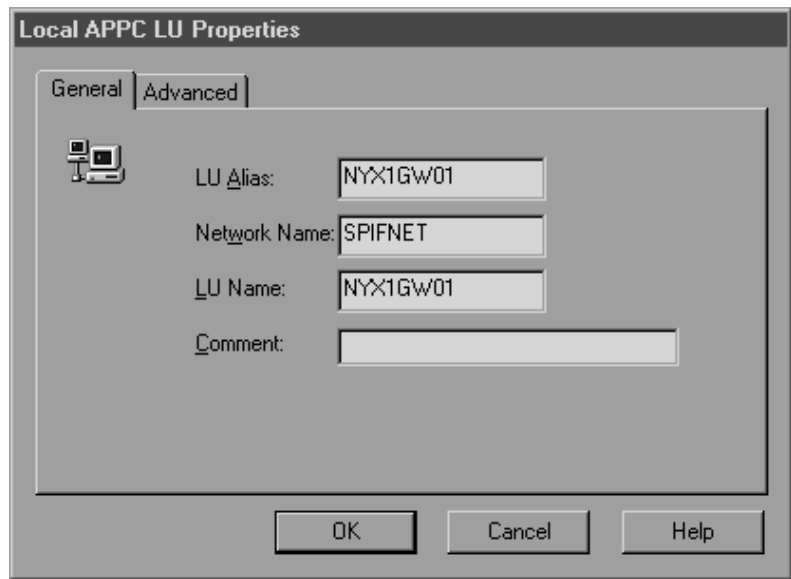


- h. Fill in the **Remote Network Address** field (**8**). Accept the default numbers in the other fields.
- i. Select the **System Identification** tab.
- j. Enter the following information:
- 1) For the **Local Node Name**, add the **Network ID** (**9**), the **Local PU Name** (**10**), and the **Local Node ID** (**1** plus **14**). Accept the **XID Type** default.
 - 2) For the **Remote Node Name**, add the **NETID** (**1**) and the **Control Point Name** (**4**).
- k. Accept the other defaults and click **OK**.

Step 5. Define a local LU

- a. Right-click on the **SNA Service** icon and select the **Insert** → **APPC** → **Local LU** option. The Local APPC LU Properties

window opens.



- b. Enter the following information:
 - The **LU Alias** (**12**).
 - The **NETID** (**9**).
 - The **LU Name** (**11**).
- c. Select the **Advanced** tab. If you are planning to use a DB2's multisite update support, ensure that you have:
 - 1) Installed Microsoft SNA Server V4 Service Pack 3
 - 2) De-selected the **Member of Default Outgoing Local APPC LU Pool** option. DB2 requires exclusive use of this LU for multisite update.
 - 3) From the **SyncPoint Support** field:
 - Select Enable.
 - Enter the SNA Server name in the **Client** field.

Syncpoint support must be enabled on this server. It is not supported on SNA clients. Therefore, the **Client** field must contain the name of the local SNA Server. Multisite update is typically required if you use Transaction Processing (TP) Monitors such as Microsoft Transaction Server, IBM TxSeries, or BEA Tuxedo.

An additional LU should be defined without Syncpoint support enabled, or if multisite update is not required. For this

LU, you should ensure that **Member of Default Outgoing Local APPC LU Pool** option is selected

d. Accept the other defaults and click **OK**.

Step 6. Define a remote LU

- a. Right-click on **SNA Services** icon and select the **Insert** → **APPC** → **Remote LU** option. The Remote APPC LU Properties window opens.
- b. Click on the **Connection** drop down box and select the appropriate connection name (**7**).
- c. Enter the partner LU name (**2**) in the **LU Alias** field.
- d. Enter the Network ID (**1**) in the **Network Name** field.



The other fields will be filled in by the program. If your LU alias is not the same as your LU Name, make sure you specify the LU Name in the appropriate field. The program will fill it in automatically, but it will be incorrect if the alias and the name are not the same.

e. Click **OK**.

Step 7. Define a mode

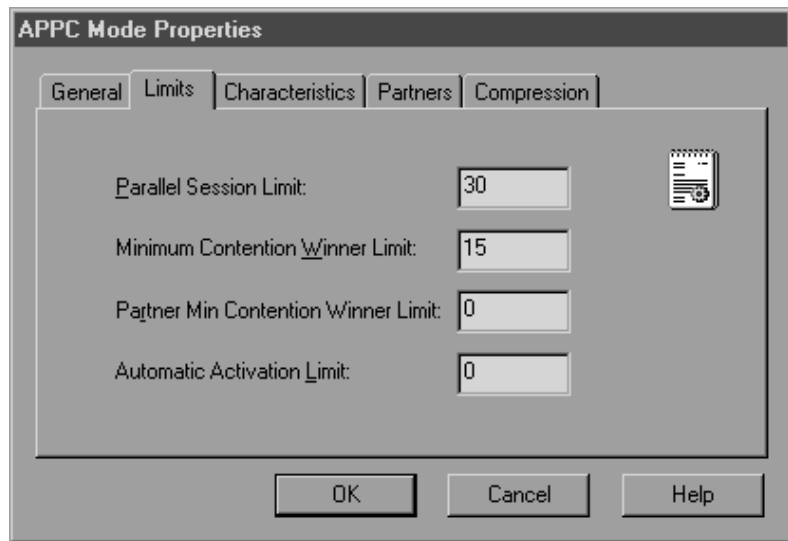
- a. Right-click on **APPC Modes** folder and select the **Insert** → **APPC** → **Mode Definition** option. The APPC Mode Properties window opens.

The screenshot shows the 'APPC Mode Properties' dialog box with the following details:

- Tab: General
- Mode Name: IBMRDB
- Comment: (empty)
- Buttons: OK, Cancel, Help

b. Enter the Mode Name **6** in the **Mode Name** field.

- c. Select the **Limits** tab.



- d. Enter appropriate numbers for the **Parallel Session Limit** and **Minimum Contention Winner Limit** fields. Your Host-Side or LAN administrator should be able to supply you with the numbers if you do not know the limits you should place here.
- e. Accept the other defaults, and click **OK**.

Step 8. Define the CPIC name properties

- a. Right-click on **CPIC Symbolic Name** folder icon and select the **Insert** → **APPC** → **CPIC Symbolic Name** option. The CPIC Name Properties window opens.

The screenshot shows the 'CPIC Name Properties' dialog box with the 'General' tab selected. The 'Name' field contains 'DB2CPIC' and the 'Comment' field is empty. Under 'Conversation Security', the 'None' radio button is selected. The 'Mode Name' dropdown menu is set to 'IBMRDB'. There is a 'User ID...' button next to the 'Program' radio button. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

- b. Enter the Symbolic Destination Name (**16**) in the **Name** field.
c. Click on the **Mode Name** drop down box and select a mode name, for example, **IBMRDB**.
d. Select the **Partner Information** tab.

The screenshot shows the 'CPIC Name Properties' dialog box with the 'Partner Information' tab selected. Under 'Partner TP Name', the 'SNA Service TP [in hex]' radio button is selected. Under 'Partner LU Name', the 'Fully Qualified' radio button is selected, with 'SPIFNET' in the first text field and 'NYM2DB2' in the second text field. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

- e. In **Partner TP Name** box, select the **SNA Service TP (in hex)** radio button and enter the Service TP name (**17**), or select the **Application TP** radio button and enter the Application TP name (**17**).
- f. In the **Partner LU Name** box, select the **Fully Qualified** radio button.
- g. Enter the fully-qualified Partner LU Name (**1** and **2**) or alias.
- h. Click **OK**.
- i. Save the configuration
 - 1) Select **File** → **Save** from the menu bar of the Server Manager window. The Save File window opens.
 - 2) Enter a unique name for your configuration into the **File Name** field.
 - 3) Click **Save**.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

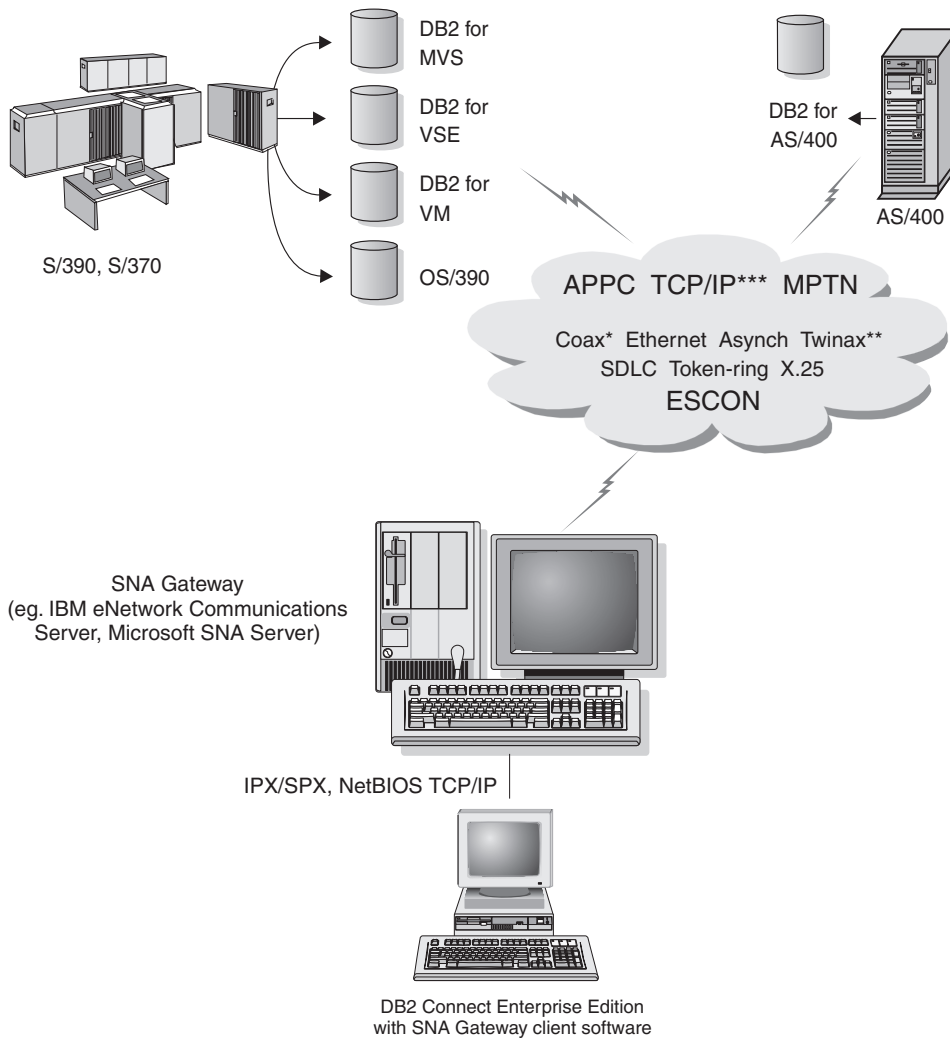
Configuring Microsoft SNA Client

Read this section if you have a Windows NT workstation that has DB2 Connect installed. This section provides step-by-step instructions for setting up the communications between your DB2 Connect workstation and a Windows NT workstation that has Microsoft SNA Server V4.0 (or later) installed.



For instructions on how to configure Microsoft SNA Server Version 4.0 for Windows NT, go to “Configuring Microsoft SNA Server for Windows” on page 303.

Figure 2 on page 313 illustrates the example DB2 Connect Server scenario.



* For Host connections only

** For AS/400

*** TCP/IP connectivity requires DB2 for OS/390 V5R1, DB2 for AS/400 V4R2, or DB2 for VM V6.1

Figure 2. Indirect Connection to host or AS/400 database server via SNA Communications Gateway

The rest of this section assumes:

1. The Microsoft SNA Server has already been configured for APPC communications with the host, and is enabled for ODBC and DRDA. Refer to the Microsoft SNA Server documentation for further information.

2. Microsoft SNA Client Version 2.11 is not already installed on your DB2 Connect workstation.

To configure the Microsoft SNA client, perform the following steps:

Step 1. Obtain Required Information

For your Microsoft SNA client software to function properly you must have access to a properly configured Microsoft SNA Server. Request that your SNA Server administrator:

1. Obtain the proper license for you to use Microsoft SNA Client on your workstation.
2. Define a user ID and password for you on the SNA Server domain.
3. Define connections to the host and AS/400 databases that you need to access, as described in “Configuring Microsoft SNA Server for Windows” on page 303.
4. Provide you with the symbolic destination name (**16**), database name (**5**), and user account to use for each database connection defined in the previous step.

If you plan to change host passwords, the SNA administrator will also need to provide you with symbolic destination names for password management tasks on each host.

5. Provide you with the Microsoft SNA Server domain name and the protocol used for communicating with the SNA server (TCP/IP, NetBEUI, IPX/SPX).

Step 2. Install the Microsoft SNA Client on the DB2 Connect Workstation

1. Obtain the Microsoft SNA Client software, and follow its instructions to start the installation program.
2. Follow the instructions on the screen to complete the installation. Choose your SNA Server domain name and communication protocol according to the instructions provided by your SNA Server administrator.
3. When you reach the Optional Components window, *deselect* Install ODBC/DRDA driver so that it will not be installed.
4. Complete the installation.

Step 3. Install DB2 Connect for Windows

1. Install DB2 Connect.
2. Click on **Start** and select **Programs** —> **DB2 for Windows NT** —> **Client Configuration Assistant**.
3. You need to provide the following information:
 - a. The Symbolic destination name (**16**) defined at the Microsoft SNA Server for the Partner LU (**2**) of the target host or AS/400 database server.

- b. The real database name (**5**).



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring IBM eNetwork Communication Server for AIX

This section describes how to configure IBM eNetwork Communication Server V5.0.3 for AIX on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC. IBM eNetwork Communication Server for AIX is the only SNA product supported for DB2 Connect running on RS/6000 machines.

Before you begin, ensure that your workstation has IBM eNetwork Communication Server V5.0.3 for AIX (CS/AIX) installed. If you need further information in order to configure your SNA environment, refer to the online help provided with CS/AIX.

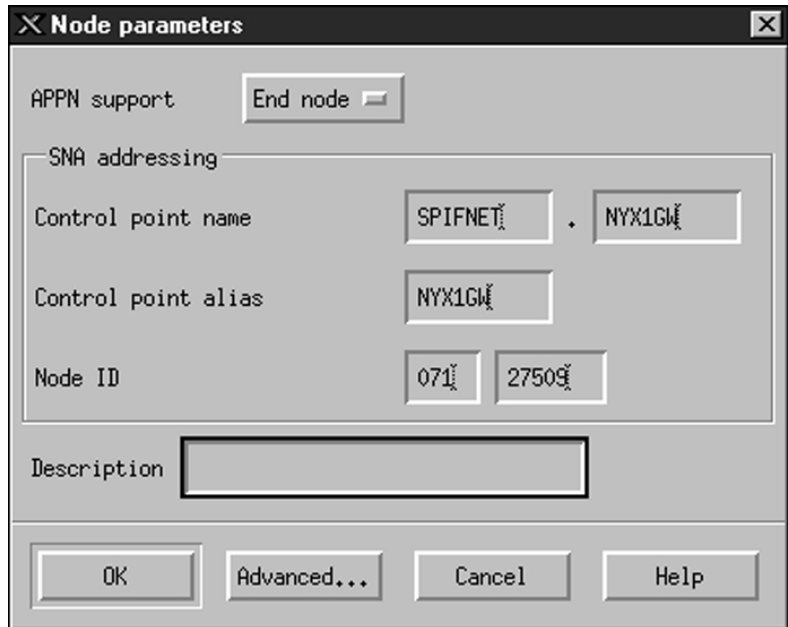
The following assumptions are made:

- The basic installation of the IBM eNetwork Communication Server V5 for AIX package has already been completed and PTF 5.0.3 has been applied.
- DB2 Connect has been installed.

To configure CS/AIX for use by DB2 Connect, log on to the system as a user with root authority and use either the `/usr/bin/snaadmin` program or the `/usr/bin/X11/xsnaadmin` program. Information about these programs can be found in the system documentation. The following steps describe how to use the `xsnaadmin` program to configure CS/AIX.

- Step 1.** Enter the command `xsnaadmin`. The Node window for the server opens.
- Step 2.** Define a node.

- a. Select **Services** → **Configure Node Parameters**. The Node Parameters window opens.



- b. Select **End node** from the **APPN support** drop-down menu.
 - c. Enter your network ID and the local PU name (**9** and **10**) in the **Control point name** fields.
 - d. Enter the local PU name (**10**) in the **Control point alias** field.
 - e. Enter your Node ID (**13** and **14**) in the **Node ID** fields.
 - f. Click **OK**.
- Step 3. Define a port.
- a. Select the Connectivity and Dependent LUs window.
 - b. Click the **Add** push button. The Add to Node window opens.
 - c. Select the **Port using** radio button.

- d. Click the **Port Using** drop down box and select the appropriate port type. For the purpose of our example, we will select the **Token ring card** option.

Token ring SAP

SNA port name: TRSAPO

Token ring card: 0

Local link name: ...

Local SAP number: 04

Initially active

HPR

Use HPR on implicit links

Use HPR link-level error recovery

Connection network

Define on connection network

CN name: SPIFNET + NYX1GW

Description:

OK Advanced... Cancel Help

- e. Click **OK**. The Port window for the chosen port type opens.
- f. Enter a name for the port in the **SNA port name** field.
- g. Select the **Initially active** check box.
- h. From the **Connection network** box, select **Define on connection network** check box.
- i. Enter your SNA Network Name (**9**) in the first part of the **CN name** field.
- j. Enter the Local PU Name (**10**) associated with your AIX computer in the second part of the **CN name** field.
- k. Click **OK**. The Port window closes and a new port opens in the Connectivity and Dependent LUs window.

Step 4. Define a link station.

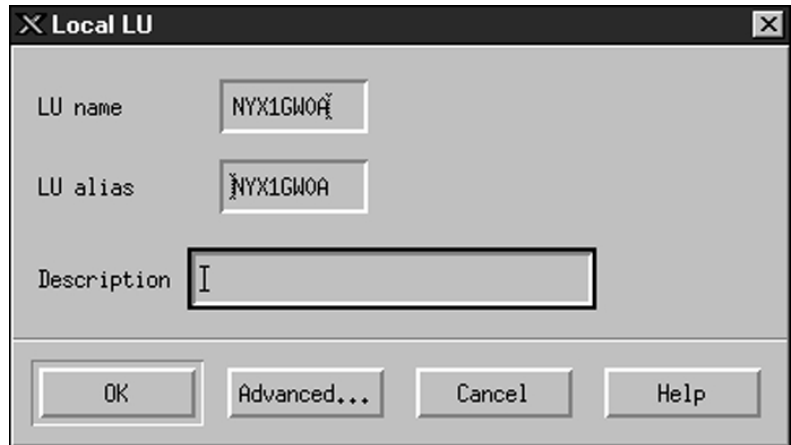
- a. In the Connectivity and Dependent LUs window, select the port that you defined in the previous step.
- b. Click the **Add** push button. The Add to Node window opens.
- c. Select the **Add a link station to port** radio button.
- d. Click **OK**. The Token ring link station window opens.

- e. Enter a name for the link in the **Name** field.
- f. Click the **Activation** drop-down box and select the **On demand** option.
- g. Select the **Independent only** option in the **LU traffic** box.
- h. In the **Independent LU traffic** box:
 - 1) Enter the network ID (**3**) and the partner LU name (**2**) in the **Remote node** fields.
 - 2) Click the **Remote node type** drop-down box and select the type of node that applies to your network.

- i. In the **Contact information** box, enter the SNA destination address (**8**) assigned for the host or AS/400 system in the **Mac address** field.
- j. Click **OK**. The Link Station window closes and a new link station appears in the Connectivity and Dependent LUs window.

Step 5. Define a local LU.

- a. Select the **Independent local LUs** window.
- b. Click the **Add** push button. The Local LU window opens.

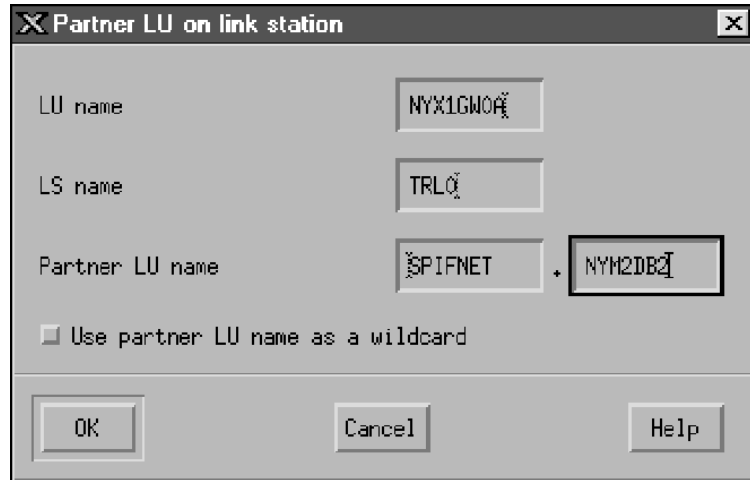


The screenshot shows a dialog box titled "Local LU". It has three input fields: "LU name" with the value "NYX1GW0A", "LU alias" with the value "NYX1GW0A", and "Description" which is currently empty. At the bottom of the dialog, there are four buttons: "OK", "Advanced...", "Cancel", and "Help".

- c. Enter your independent local LU Name (**11**) in the **LU name** field.
- d. Enter the same name in the **LU alias** field (**12**).
- e. Click **OK**. The new LU appears in the Independent Local LUs window.

Step 6. Define a partner LU over the link station.

- a. Select **Services** → **APPC** → **New PLUs** → **Over Link Station** from the menu bar. The Partner LU on link station window opens.



- b. Enter the name for the local LU (**11**) you defined previously in the **LU name** field.
- c. Enter the name for the link station you defined previously in the **LS name** field.
- d. Enter the name of the partner LU you want to connect to (**2**) in the **Partner LU name** field.
- e. Click **OK**. The Partner LU appears in the Independent Local LUs window of the Local LU that was created in the previous step.

Step 7. Define an alias for the partner LU.

- a. Select the Remote Systems window.
- b. Click the **Add** push button. The Add to Node window opens.
- c. Select the **Define partner LU alias** radio button.
- d. Click **OK**. The Partner LU window opens.
- e. Enter an alias for the partner LU in the **Alias** field.
- f. Enter the same value in the **Uninterpreted name** field.
- g. Click **OK**.

Step 8. Define a mode.

- a. Select **Services** → **APPC** → **Modes** from the menu bar. The Modes window opens.

- b. Click the **New** push button. The Mode window opens.

Mode

Name: IBMRE

Session limits:

- Initial: 20
- Maximum: 32767
- Min con. winner sessions: 10
- Min con. loser sessions: 10
- Auto-activated sessions: 04

Receive pacing window:

- Initial: 8
- Maximum: 8 (Optional)

Specify timeout

Restrict max RU size

Description:

OK Cancel Help

- c. Enter a mode name (**15**) in the **Name** field.
- d. The configuration values below are suggested for the following fields:

- **Initial session limits:** 20
- **Maximum session limits:** 32767
- **Min con. winner sessions:** 10
- **Min con. loser sessions:** 10
- **Auto-activated sessions:** 4
- **Initial receive pacing window:** 8

These values are suggested because they are known to work. You will need to tailor these values so that they are optimized for your particular application environment.

- e. Click **OK**. The new mode appears in the Modes window.

- f. Click **Done**.
- Step 9. Define the CPI-C destination name.
- Select **Services** → **APPC** → **CPI-C** from the menu bar. The CPI-C destination names window opens.
 - Click the **New** push button. The CPI-C destination window opens.

- In the **Name** field, enter the Symbolic Destination Name (**16**) you want to associate with the host or AS/400 server database. This example uses db2cpic.
- In the **Partner TP** box:

- For DB2 for MVS/ESA, DB2 for OS/390, and DB2 for AS/400, select the **Service TP (hex)** radio button, and enter the hexadecimal TP number (**17**). (For DB2 Universal Database for OS/390 or DB2/MVS, you can also use the default application TP DB2DRDA. For DB2 for AS/400 you can also use the default application TP QCNTEDDM.)
 - For DB2 for VM or VSE, select the **Application TP** radio button. For DB2 for VM, enter the DB2 for VM database name. For DB2 for VSE, enter the AXE as the application TP (**17**).
- e. In the **Partner LU and mode** box:
 - 1) Select the **Use PLU alias** radio button, and enter the partner LU alias (**2**) you created in a previous step.
 - 2) In the **Mode** field, enter the mode name (**15**) for the mode that you created in a previous step.
 - f. In the **Security** box, select the radio button that corresponds to the type of security level that you want to run on your network.
 - g. Click **OK**. The new destination name appears in the Destination Names window.
 - h. Click **Done**.

Step 10. Test the APPC connection.

- a. Start the SNA subsystem by entering the `/usr/bin/sna start` command. You can enter the `/usr/bin/sna stop` command to stop the SNA subsystem first, if required.
- b. Start the SNA administration program. You can enter either the `/usr/bin/snaadmin` command or the `/usr/bin/X11/xsnaadmin` command.
- c. Start the subsystem node. Select the appropriate node icon in the button bar, and click the **Start** push button.
- d. Start the link station. Select the link station you defined previously in the Connectivity and Dependent LUs window, and click the **Start** push button.
- e. Start the session. Select the LU you defined previously in the Independent Local LUs window, and click the **Start** push button. A session activation window opens.
- f. Select or enter a partner LU and mode.
- g. Click **OK**.



You may also need to contact your database or network administrators to have your local LU names added to the appropriate tables in order to access the host or AS/400 server database.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring Bull SNA for AIX

This section describes how to configure Bull DPX/20 SNA/20 Server on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC. If Bull DPX/20 SNA/20 Server is installed prior to installing DB2 Connect, DB2 Connect uses Bull SNA. Otherwise, you need to configure DB2 Connect to work with IBM eNetwork Communications Server V5.0.3 for AIX. See “Configuring IBM eNetwork Communication Server for AIX” on page 315 for more information.

To determine if Bull SNA is installed on your AIX 4.2 (or later) system, enter the following command:

```
lslpp -l express.exsrv+dsk
```

If Bull SNA is installed, you will see output similar to the following:

Fileset	Level	State	Description

Path: /usr/lib/objrepos			
express.exsrv+dsk	2.1.3.0	COMMITTED	EXPRESS SNA Server and Integrated Desktop

If you install Bull SNA after installing DB2 Connect and you want DB2 Connect to use Bull SNA instead of IBM eNetwork Communications Server for AIX, log on to the system as a user with root authority and enter the following command:

```
/usr/lpp/db2_06_01/cfg/db2cfgos
```

If you want to install the Bull DPX/20 SNA/20 Server, then you must have the following software:

- __ 1. AIX V4.1.4
- __ 2. Express SNA Server V2.1.3

For more information on setting up your SNA environment, refer to the Bull DPX/20 SNA/20 Server Configuration Guide.

DB2 Connect, when used with the Bull SNA server, cannot have inbound APPC connections from remote clients. The only APPC connections it can have are outbound APPC connections to the host.

To configure Bull SNA for use by DB2 Connect, enter the **express** command to configure the following SNA parameters:

Config	Express	Default configuration for EXPRESS
Node	NYX1	SPIFNET.NYX1 (HOSTNAME=NYX1)
Indep. LUs	6.2 LUs Using All Stations	
LU	NYX1	Control Point LU
Link	tok0.00001	Link (tok0)
Station	MVS	To MVS from NYX1
LU	NYX1GW01	To MVS from NYX1
LU Pair	NYM2DB2	To MVS from NYX1
Mode	IBMRDB	IBMRDB

Use default values for fields not listed.

The following example illustrates the sample configuration:

Defining hardware:

```
System (hostname) = NYX1
Adapter and Port  = NYX1.tok0
MAC Address      = 400011529778
```

Defining SNA node:

```
Name              = NYX1
Description       = SPIFNET.NYX1 (HOSTNAME=NYX1)
Network ID       = SPIFNET
Control Point    = NYX1
XID Block        = 05D
XID ID           = 29778
```

Defining token ring link:

```
Name              = tok0.00001
Description       = Link (tok0)
Connection Network name
Network ID       = SPIFNET
Control Point    = NYX
```

Defining token ring station:

```
Name              = MVS
Description       = To MVS from NYX1
Remote MAC address = 400009451902
Remote Node name
Network ID       = SPIFNET
Control Point    = NYX
```

Defining Local LU 6.2:

```
Name          = NYX1GW01
Description   = To MVS from NYX1
Network ID    = SPIFNET
LU name       = NYX1GW01
```

Defining Remote LU 6.2:

```
Name          = NYM2DB2
Description   = To MVS from NYX1
Network ID    = SPIFNET
LU name       = NYM2DB2
Remote Network ID = SPIFNET
Remote Control Point = NYX
Uninterpreted Name = NYM2DB2
```

Defining Mode:

```
Name          = IBMRDB
Description   = IBMRDB
Class of service = #CONNECT
```

Defining Symbolic Destination Info:

```
Name          = DB2CPIC
Description   = To MVS from NYX1
Partner LU    = SPIFNET.NYM2DB2
Mode          = IBMRDB
Local LU      = NYX1GW01
Partner TP    = DB2DRDA
```

After you have configured these SNA parameters, you must stop and start the SNA server. To do this, perform the following steps:

- Step 1. Log on to the system as a user with root authority.
- Step 2. Make sure your PATH contains the \$express/bin (/usr/lpp/express/bin)entry.
- Step 3. Check for active users before stopping the server by entering the following command:

```
express_adm shutdown
```
- Step 4. Stop all EXPRESS activity by entering the following command:

```
express_adm stop
```
- Step 5. Start EXPRESS by entering the following command:

```
express_adm start
```



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring SNAplus2 for HP-UX

This section describes how to configure SNAplus2 for HP-UX on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC. SNAplus2 for HP-UX is the only SNA product supported for DB2 Connect running on HP-UX V10 and V11 machines.

Before you begin, ensure that your workstation has HP-UX SNAplus2 installed. If you need more information in order to configure your SNA environment, refer to the online help provided with SNAplus2.

The following assumptions are made:

- The basic installation of the SNAplus2 for HP-UX package has already been completed.
- DB2 Connect has been installed.

To configure SNAplus2 for DB2 Connect, log on to the system as a user with root authority and use either the `/opt/sna/bin/snapadmin` program or the `/opt/sna/bin/X11/xsnapadmin` program. Information about these programs can be found in the system documentation. The following steps describe how to use the `xsnapadmin` program to configure SNAplus2.

- Step 1. Enter the command `xsnapadmin`. The Node window for the server opens.
- Step 2. Define a node.

- a. Select **Services** → **Configure Node Parameters**. The Node Parameters window opens.

The screenshot shows a dialog box titled "Node parameters". At the top left, there is a label "APPN support" and a dropdown menu currently displaying "End node". Below this is a section titled "SNA addressing" which contains three rows of input fields: "Control point name" with the value "SPIFNET . NYX1GW", "Control point alias" with the value "NYX1GW", and "Node ID" with the values "071" and "27509". Below the SNA addressing section is a "Description" label followed by an empty text box. At the bottom of the dialog are four buttons: "OK", "Advanced...", "Cancel", and "Help".

- b. Select **End node** from the **APPN support** drop-down menu.
 - c. Enter your network ID and the local PU name (**9** and **10**) in the **Control point name** fields.
 - d. Enter the local PU name (**10**) in the **Control point alias** field.
 - e. Enter your Node ID (**13** and **14**) in the **Node ID** fields.
 - f. Click **OK**.
- Step 3. Define a port.
- a. Select the Connectivity and Dependent LUs window.
 - b. Click the **Add** push button. The Add to Node window opens.
 - c. Select the **Port using** radio button.
 - d. Click the **Port Using** drop down box and select the appropriate port. For the purpose of our example, we will select the **Token ring card** option.

- e. Click **OK**. The Port window for the chosen port type opens.

Token ring SAP

SNA port name: TRSAPO

Token ring card: 0

Local SAP number: 04

Initially active

Connection network

Define on connection network

CN name: SPIFNET . NYX1GW

Description: I

OK Advanced... Cancel Help

- f. Enter a name for the port in the **SNA port name** field.
- g. Select the **Initially active** check box.
- h. Select the **Define on a connection network** check box.
- i. Enter your network ID (**9**) in the first part of the **CN name** field.
- j. Enter your local control point name (**10**) in the second part of the **CN name** field.
- k. Click **OK**. The Port window closes and a new port appears in the Connectivity and Dependent LUs window.

Step 4. Define a link station.

- a. In the Connectivity and Dependent LUs window, select the port that you defined in the previous step.
- b. Click the **Add** push button. The Add to Node window opens.
- c. Select the **Add a link station to port** radio button.

- d. Click **OK**. The Token ring link station window opens.

Token ring link station

Name: TRLO

SNA port name...: TRSAP0

Activation: On demand

LU traffic: Any Independent only Dependent only

Independent LU traffic:

Remote node...: SPIFNET . NYM

Remote node type: End or LEN node

Contact information:

MAC address: 400009451902 Flip

SAP number: 04

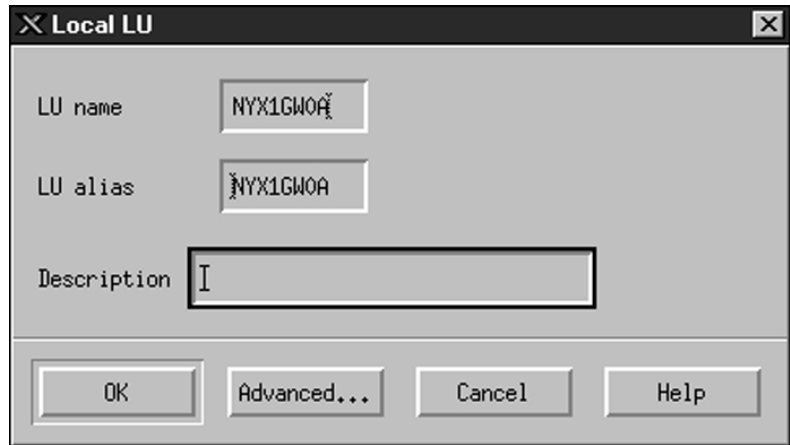
Description:

OK Advanced... Cancel Help

- e. Enter a name for the link in the **Name** field.
- f. Click the **Activation** drop-down box and select the **On demand** option.
- g. Select the **Independent only** option in the **LU traffic** box.
- h. In the **Independent LU traffic** box:
- 1) Enter the network ID (**3**) and the partner LU name (**2**) in the **Remote node** fields.
 - 2) Click the **Remote node type** drop-down box and select the type of node that applies to your network.
- i. In the **Contact information** box, enter the SNA destination address (**8**) assigned for the host or AS/400 system in the **Mac address** field.
- j. Click **OK**. The Link Station window closes and a new link station appears in the Connectivity and Dependent LUs window.

Step 5. Define a local LU.

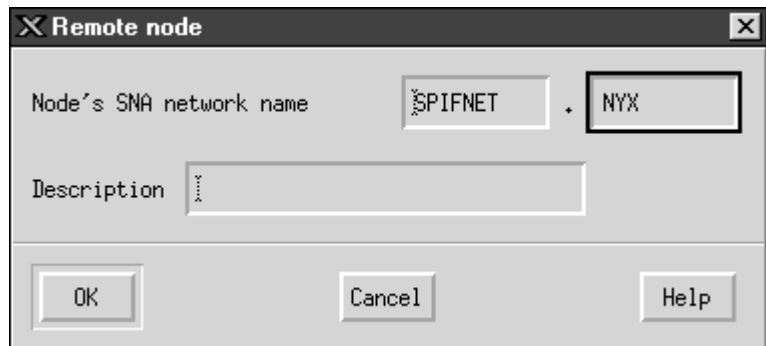
- a. Select the **Independent local LUs** window.
- b. Click the **Add** push button. The Local LU window opens.



- c. Enter your independent local LU Name (**11**) in the **LU name** field.
- d. Enter the same name in the **LU alias** field (**12**).
- e. Click **OK**. The new LU appears in the Independent Local LUs window.

Step 6. Define a remote node.

- a. Select the **Remote Systems** window.
- b. Click the **Add** push button. The Add to Node window opens.
- c. Select **Define remote node**.
- d. Click **OK**. The Remote Node configuration window opens.



- e. Enter the network ID (**3**) and the partner LU name (**2**) in the **Node's SNA network name** field.
- f. Click **OK**. The remote node appears in the Remote Systems window, and a default partner LU is defined for the node.

Step 7. Define a partner LU.

- a. In the Remote Systems window, double-click the default partner LU that was created when you defined a remote node in the previous step. The Partner LU window opens.

The screenshot shows a dialog box titled "Partner LU". It has several input fields and a checkbox. The "Partner LU name" field is split into two parts: "SPIFNET" and "NYM2DB2". Below this is a section titled "Partner LU characteristics" which contains "Alias" and "Uninterpreted name" fields, both containing "NYM2DB2" and labeled "(Optional)". There is a checkbox labeled "Supports parallel sessions" which is currently unchecked. Below this section is a "Location ..." field containing "SPIFNET" and "NYM2DB2". At the bottom is a "Description" field which is empty. The dialog box has "OK", "Cancel", and "Help" buttons at the bottom.

- b. Enter the same partner LU name (**2**) in the **Alias** and **Uninterpreted name** fields.
- c. Select **Supports parallel sessions**.
- d. Click **OK**.

Step 8. Define a mode.

- a. Select **Services** → **APPC** → **Modes** from the menu bar. The Modes window opens.

- b. Click the **New** push button. The Mode window opens.

Mode

Name: IBMRE

Session limits:

- Initial: 20
- Maximum: 32767
- Min con. winner sessions: 10
- Min con. loser sessions: 10
- Auto-activated sessions: 04

Receive pacing window:

- Initial: 8
- Maximum: (Optional)

Specify timeout

Restrict max RU size

Description:

OK Cancel Help

- c. Enter a mode name (**15**) in the **Name** field.
- d. The configuration values below are suggested for the following fields:
- **Initial session limits:** 20
 - **Maximum session limits:** 32767
 - **Min con. winner sessions:** 10
 - **Min con. loser sessions:** 10
 - **Auto-activated sessions:** 4
 - **Initial receive pacing window:** 8
- These values are suggested because they are known to work. You will need to tailor these values so that they are optimized for your particular application environment.
- e. Click **OK**. The new mode appears in the Modes window.

- f. Click **Done**.
- Step 9. Define the CPI-C destination name.
- Select **Services** → **APPC** → **CPI-C** from the menu bar. The CPI-C destination names window opens.
 - Click the **New** push button. The CPI-C destination window opens.

- In the **Name** field, enter the Symbolic Destination Name (**16**) you want to associate with the host or AS/400 server database. This example uses db2cpic.
- In the **Partner TP** box:

- For DB2 for MVS/ESA, DB2 for OS/390, and DB2 for AS/400, select the **Service TP (hex)** radio button, and enter the hexadecimal TP number (**17**). (For DB2 Universal Database for OS/390 or DB2/MVS, you can also use the default application TP DB2DRDA. For DB2 for AS/400 you can also use the default application TP QCNTEDDM.)
 - For DB2 for VM or VSE, select the **Application TP** radio button. For DB2 for VM, enter the DB2 for VM database name. For DB2 for VSE, enter the AXE as the application TP (**17**).
- e. In the **Partner LU and mode** box:
 - 1) Select the **Use PLU alias** radio button, and enter the partner LU alias (**2**) you created in a previous step.
 - 2) In the **Mode** field, enter the mode name (**15**) for the mode that you created in a previous step.
 - f. In the **Security** box, select the radio button that corresponds to the type of security level that you want to run on your network.
 - g. Click **OK**. The new destination name appears in the Destination Names window.
 - h. Click **Done**.

Step 10. Test the APPC connection.

- a. Start the SNA subsystem by entering the `/opt/sna/bin/sna start` command. You may need to enter the `/opt/sna/bin/sna stop` command to stop the SNA subsystem first.
- b. Start the SNA administration program. You can use either the `/opt/sna/bin/snaadmin` command or the `/opt/sna/bin/X11/xsnaadmin` command.
- c. Start the subsystem node. Select the appropriate node icon in the button bar, and click the **Start** push button.
- d. Start the link station. Select the link station you defined previously in the Connectivity and Dependent LUs window, and click the **Start** push button.
- e. Start the session. Select the LU you defined previously in the Independent Local LUs pane, then click the **Start** button. A session activation window opens. Select or enter a partner LU and mode.
- f. Click **OK**.



You may also need to contact your database or network administrators to have your Local LU names added to the appropriate tables in order to access the host or AS/400 server database.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Configuring SNAP-IX Version 6.0.1 for SPARC Solaris

This section describes how to configure DCL SNAP-IX V6.0.6 for SPARC Solaris on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC.

Before you begin, ensure that your workstation has DCL SNAP-IX V6.1.0 for SPARC Solaris installed. For more information on configuring your SNA environment, refer to the online help provided with DCL SNAP-IX V6.1.0 for SPARC Solaris.

The following assumptions have been made:

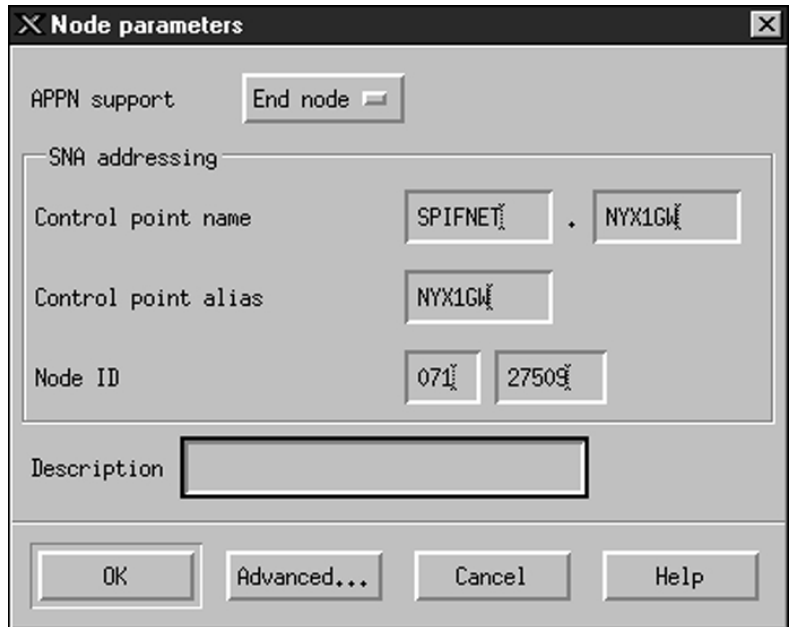
- You have completed the basic installation of the DCL SNAP-IX V6.1.0 for SPARC Solaris package.
- You have installed DB2 Connect.

To configure DCL SNAP-IX V6.1.0 for SPARC Solaris for DB2 Connect, log on to the system as a user with root authority and use either the **/opt/sna/bin/snaadmin** or the **/opt/sna/bin/X11/xsnaadmin** program. For more information about these programs, refer to the system documentation.

To configure DCL SNAP-IX V6.1.0 for SPARC Solaris using the **xsnaadmin** program, perform the following steps:

- Step 1.** Enter the **xsnaadmin** command. The Node window for the server opens.
- Step 2.** Define a node.

- a. Select **Services** → **Configure Node Parameters**. The Node Parameters window opens.



- b. Select **End node** from the **APPN support** drop-down menu.
- c. Enter your network ID and the local PU name (**9** and **10**) in the **Control point name** fields.
- d. Enter the local PU name (**10**) in the **Control point alias** field.
- e. Enter your Node ID (**13** and **14**) in the **Node ID** fields.
- f. Click **OK**.

Step 3. Define a port.

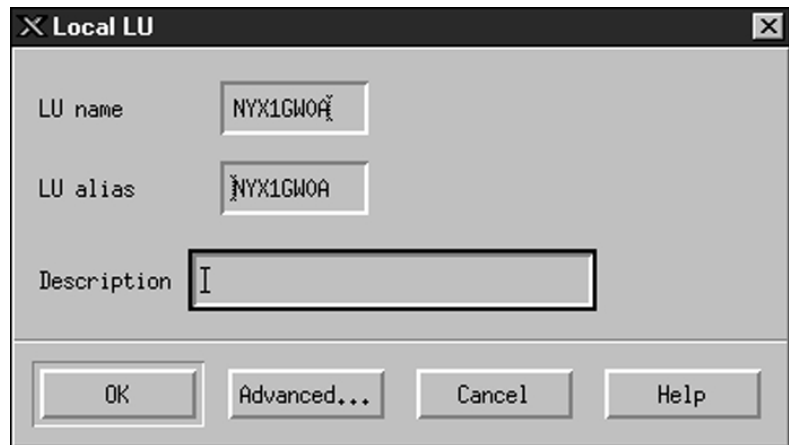
- a. Select the Connectivity and Dependent LUs window.
- b. Click **Add**. The Add to Node window opens.
- c. Select the **Port using** radio button.
- d. Click on the **Port Using** drop down box and select the appropriate port. For the purpose of our example, we will select the **SunTRI/P Adapter** option.
- e. Click **OK**. The Port window for the chosen port type opens.
- f. Enter a name for the port in the **SNA port name** field.
- g. Select the **Initially active** check box.
- h. Click **OK**. The Port window closes, and a new port appears in the Connectivity and Dependent LUs window.

Step 4. Define a link station.

- a. In the Connectivity and Dependent LUs window, select the port that you defined in the previous step.
- b. Click **Add**. The Add to Node window opens.
- c. Select the **Add a Link Station to Port** radio button.
- d. Click **OK**. The Token Ring Link Station window opens.
- e. Enter a name for the link in the **Name** field.
- f. Select the **Independent Only** option in the LU traffic box.
- g. In the Independent LU traffic box:
 - 1) Enter the Network ID (**3**) and the Partner PU Name (**2**) in the Remote Node fields.
 - 2) Click on the **Remote node type** drop down box and select the type of node that applies to your network.
- h. In the Contact information box, enter the SNA Destination Address (**8**) assigned for the host or AS/400 system in the Mac address field.
- i. Click **OK**. The Link Station window closes, and a new link station appears in the Connectivity and Dependent LUs pane.

Step 5. Define a local LU.

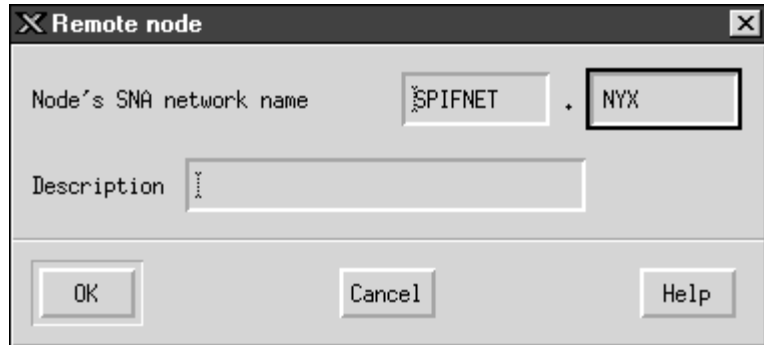
- a. Select the **Independent local LUs** window.
- b. Click the **Add** push button. The Local LU window opens.



- c. Enter your independent local LU Name (**11**) in the **LU name** field.
- d. Enter the same name in the **LU alias** field (**12**).
- e. Click **OK**. The new LU appears in the Independent Local LUs window.

Step 6. Define a remote node.

- a. Select the **Remote Systems** window.
- b. Click the **Add** push button. The Add to Node window opens.
- c. Select **Define remote node**.
- d. Click **OK**. The Remote Node configuration window opens.



- e. Enter the network ID (**3**) and the partner LU name (**2**) in the **Node's SNA network name** field.
- f. Click **OK**. The remote node appears in the Remote Systems window, and a default partner LU is defined for the node.

Step 7. Define a partner LU.

- a. Select **Services** → **APPC** → **New partner LUs** → **Partner LU on Remote Node** from the menu bar. The Partner LU window opens.
- b. Enter the Partner LU name in the **Partner LU Name** fields.
- c. Enter the same Partner LU name (**2**) in the **Alias and Uninterpreted Name** fields.
- d. Select **Supports Parallel Sessions**.
- e. Enter the Partner PU name in the **Location** fields.
- f. Click **OK**.

Step 8. Define a mode.

- a. Select **Services** → **APPC** → **Modes** from the menu bar. The Modes window opens.

- b. Click the **New** push button. The Mode window opens.

The screenshot shows a dialog box titled "Mode" with a close button in the top right corner. The dialog is divided into several sections:

- Name:** A text field containing "IBMRE".
- Session limits:** A group box containing:
 - Initial:** 20
 - Maximum:** 32767
 - Min con. winner sessions:** 10
 - Min con. loser sessions:** 10
 - Auto-activated sessions:** 04
- Receive pacing window:** A group box containing:
 - Initial:** 8
 - Maximum:** (Optional)
- Specify timeout:** An unchecked checkbox.
- Restrict max RU size:** An unchecked checkbox.
- Description:** An empty text field.

At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help".

- c. Enter a mode name (**15**) in the **Name** field.
- d. The configuration values below are suggested for the following fields:

- **Initial session limits:** 20
- **Maximum session limits:** 32767
- **Min con. winner sessions:** 10
- **Min con. loser sessions:** 10
- **Auto-activated sessions:** 4
- **Initial receive pacing window:** 8

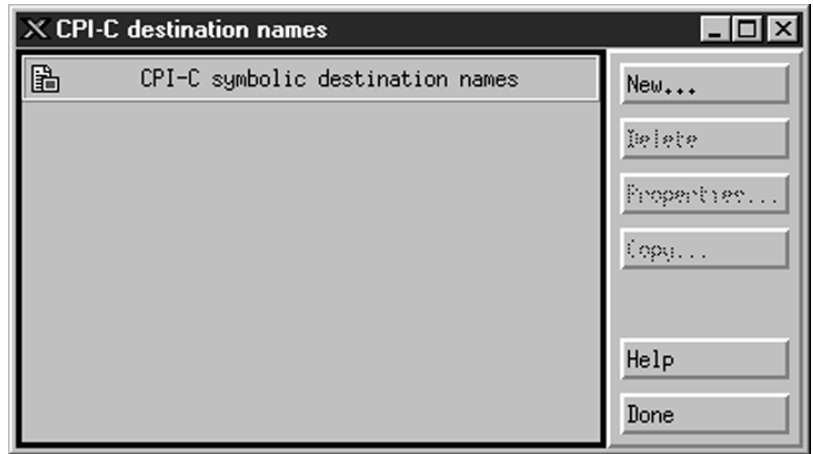
These values are suggested because they are known to work. You will need to tailor these values so that they are optimized for your particular application environment.

- e. Click **OK**. The new mode appears in the Modes window.

f. Click **Done**.

Step 9. Define the CPI-C destination name.

a. Select **Services** → **APPC** → **CPI-C** from the menu bar. The CPI-C Destination Names window opens.



- b. Click **New**. The CPI-C Destination window opens.

The screenshot shows the 'CPI-C destination' dialog box with the following fields and options:

- Name:** db2cpic
- Local LU:**
 - Specify local LU alias: NYX1GWO4
 - Use default LU
- Partner LU and mode:**
 - Use PLU alias: NYM2DB2
 - Use PLU full name
 - Mode: IBMRO
- Partner TP:**
 - Application TP
 - Service TP (Hex): 07F6C4C2
- Security:**
 - None
 - Same
 - Program
 - Program strong
- User ID: (empty)
- Password: (empty)
- Description: (empty)

Buttons: OK, Cancel, Help

- c. Enter the Symbolic Destination Name (**16**) you want to associate with the host or AS/400 server database in the **Name** field. This example uses DB2CPIC.
- d. In the **Local LU** box:
- Select the specific Local LU alias radio button and enter the Local LU alias you created previously.
- e. In the **Partner LU and mode** box:
- 1) Select the **Use PLU Alias** radio button, and enter the Partner LU Alias (**2**) you created in a previous step.

- 2) Enter the Mode name (**15**) for the mode that you created in a previous step in the **Mode** field.
- f. In the Partner TP Box:
 - 1) For DB2 for MVS/ESA, DB2 for OS/390, and DB2 for AS/400, select the **Service TP (hex)** radio button.
 - 2) Enter the hexadecimal TP number (**17**). (For DB2 Universal Database for OS/390 or DB2/MVS, you can also use the default application TP DB2DRDA. For DB2 for AS/400 you can also use the default application TP QCNTEDDM.)
 - 3) For DB2 for VM or VSE, select the **Application TP** radio button. For DB2 for VM, enter the DB2 for VM database name. For DB2 for VSE, enter the AXE as the application TP. (**17**)
 - g. In the **Security** box, select the radio button that corresponds to the type of security level that you want to run on your network.
 - h. Click **OK**. The new destination name appears in the Destination Names window.
 - i. Click **Done**.
 - j. Test the APPC connection.
 - 1) Start the SNA subsystem by entering the `/opt/sna/bin/sna start` command. You can enter the `/opt/sna/bin/sna stop` command to stop the SNA subsystem first, if required.
 - 2) Start the SNA administration program. You can use either the `/opt/sna/bin/snaadmin` or the `/opt/sna/bin/X11/xsnaadmin` command.
 - 3) Start the subsystem node. Select the node icon in the button bar, and click the **Start** push button.
 - 4) Start the link station. Select the link station that you previously defined in the Connectivity and Dependent LUs window. Click **Start**.
 - 5) Start the session. Select the LU you defined previously in the Independent Local LUs window, then click **Start**. A session activation window opens. Select or enter the Partner LU and Mode desired.
 - 6) Click **OK**.



You now need to update the DB2 directories, bind utilities and applications to the server, and test the connection.

For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29. For manual configuration instructions, and for UNIX platforms, see “3. Catalog the APPC or APPN Node” on page 347 and the sections that follow.

Step 10. Test the APPC connection.

- a. Start the SNA subsystem by entering the **/opt/sna/bin/sna start** command. You can enter the **/opt/sna/bin/sna stop** command to stop the SNA subsystem first, if required.
- b. Start the SNA administration program. You can use either the **/opt/sna/bin/snaadmin** or the **/opt/sna/bin/X11/xsnaadmin** command.
- c. Start the subsystem node. Select the node icon in the button bar, and click the **Start** push button.
- d. Start the link station. Select the link station that you previously defined in the Connectivity and Dependent LUs window. Click **Start**.
- e. Start the session. Select the LU you defined previously in the Independent Local LUs window, then click **Start**. A session activation window opens. Select or enter the Partner LU and Mode desired.
- f. Click **OK**.



You may also need to contact your database or network administrators to have your Local LU names added to the appropriate tables in order to access the host or AS/400 server database.

Configuring SunLink 9.1 for Solaris

This section describes how to configure SunLink 9.1 (SunLink SNA) for Solaris on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC. SunLink SNA for Solaris is the only SNA product supported for DB2 Connect running on Solaris.

Before you begin, ensure that your workstation has SunLink SNA 9.1 installed. The following assumptions have been made:

- The basic installation of the SunLink SNA PU 2.1 for Solaris package has already been completed.
- DB2 Connect has been installed.

To configure the SunLink SNA Server for use by DB2 Connect, log on to the system as a user with root authority and perform the following steps:

Step 1. "Start the Configuration Program"

Step 2. "Configuring for a Link Station"

Step 3. "Configuring for a Session" on page 346



You may also need to contact your database or network administrators to have your local LU names added to the appropriate tables in order to access the host or AS/400 server database.

Start the Configuration Program

The Sunlink Version 9.1 configuration tool uses a graphical user interface accessible through X Windows. To begin the configuration, do the following:

1. Log on as root.
2. To initialize the daemons:
 - a. Start the **sunsetup** program by entering:
`/opt/SUNWgman/sunsetup`
 - b. Select option 4, and then 5 (Start GMAN/PU21 Software).
3. To start the graphical interface for configuring Sunlink:
 - a. Export your display (for example, export DISPLAY=hostname:0)
 - b. Start the **sungmi** program by entering:
`/opt/SUNWgmi/sungmi`

Configuring for a Link Station

In SNA networking you must have a working link station before you can establish a session. To establish a link station with Sunlink 9.1, complete the following steps:

1. Double click on the **config1** folder in the left-hand pane of the configuration window. A hierarchical tree of icons representing resources will open. Each of the configuration actions you will perform can be launched by right-clicking on the appropriate icon in this window.
2. Begin by configuring **Systems**. Select the **Systems** icon, and right-click.
 - a. From the menu that appears, select **New** —> **System**
 - b. Set HOST = Solaris3. All other information can be left to defaults.
3. Next, you will need to configure **PU2.1 Servers**. Select the **PU2.1 Servers** icon, and right-click.
 - a. From the menu that appears, select **New PU2.1 Servers**> **PU2.1 Server**.
 - b. If you are using Solaris3, enter the following:

```
Name: OMXUF5
CP Name: CAIBMOML.OMXUF5
Command Options: -t -1
sunop service: brxadmin_pu2
lu6.2 service: brxlu62_serv
```

All other options can be left as default values.

4. The next step is to configure your LAN connections. Select the **LAN Connections** icon, and right-click.
 - a. From the menu that appears, select **New 'Lan Connections' LAN Connection**.
 - b. If you are using Solaris3, enter the following:

```
Line Name: MAC
Local Mac: 08002082611F
```
 - c. Click on **Advanced**. Ensure that the following is set:

```
Lan Speed: 16Mbs
```
5. The last item is your DLC settings. Select **MAC** icon, and right-click.
 - a. From the menu that appears, select **New —> DLC (PU2)**.
 - b. For example, If you are using Solaris3, enter the following:

```
DLC Name: Jetsons
Remote Mac: 400011529798
Remote CP: CAIBMOML.OMXR88
```

Stop the SNA daemons and start them again. You should now see an active Link Station connection. Ensure that this works before continuing.

Configuring for a Session

To configure an SNA session for database communications, double click on the **config1** folder in the left-hand pane of the configuration window. A hierarchical tree of icons representing resources will open. Each of the configuration actions you will perform can be launched by right-clicking on the appropriate icon in this window.

To configure for a session:

1. Start by configuring the independent LU. Select the **Independent LU** icon, and right-click.
 - a. From the menu that appears, select **New —> Independent LU**.
 - b. If you are using Solaris3, enter the following:

```
Name: OMXUF50A
```
 - c. Click on **Advanced**. Set the session limit as follows:

```
Session Limit: 16
Sync level : No
```


Note: Two-Phase commit is currently not supported by DB2 with SUNLINK.

2. Next, set the partner LU. Select the **Partner LU's** icon, and right-click.
 - a. From the menu that appears, select **New** → **Partner LU**.
 - b. If you are using Solaris3, enter the following:

```
Name: OMXR880A
Local LU: OMXUF50A
```
3. Finally, set the mode. Select the **OMXR880A** icon found under the **Partner LU** icon, and right-click.
 - a. From the menu that appears, select **New** → **Mode**.
 - b. For example, If you are using Solaris3, enter the following:

```
Mode Name: IBMRDB
DLC Name: Jetsons
```

To activate the session, you must stop and then restart the SNA daemons.

3. Catalog the APPC or APPN Node

You must add an entry to the DB2 Connect workstations's node directory to describe the remote node. In most cases, you will add an APPC node entry to the node directory. For OS/2 and Windows 32-bit operating systems, you can alternatively add an APPN node entry if your local SNA node has been set up as an APPN node.

To catalog the node, perform the following steps:

- Step 1.** Log on to the system as a user with System Administrative (SYSADM) or System Controller (SYSCTRL) authority.
- Step 2.** If you are using DB2 Connect on a UNIX platform, set up the instance environment and invoke the DB2 command line processor. Run the start-up script as follows:

```
. INSTHOME/sql11ib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sql11ib/db2cshrc (for C shell)
```

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

- Step 3.** To catalog an APPC node, specify the chosen alias (*node_name*), Symbolic destination name (*sym_dest_name*), and the APPC security type (*security_type*) that the client will use for the APPC connection. Enter the following commands:

```
catalog "appc node node_name remote sym_dest_name
        security security_type"
terminate
```

The *sym_dest_name* parameter is case-sensitive and *must* exactly match the case of the Symbolic destination name you defined previously.

For example, to catalog a remote database server with the Symbolic destination name *DB2CPIC* on the node called *db2node*, using APPC Security type *program*, enter the following commands:

```
catalog appc node db2node remote DB2CPIC security program
terminate
```

- Step 4. To catalog an APPN node, specify the chosen alias (*node_name*), the network ID (**9**), the remote partner LU (**4**), the transaction program name (**17**), the mode (**15**), and the security type. Enter the following commands substituting your values from the worksheet in Table 30 on page 268:

```
catalog "appn node db2node network SPIFNET remote NYM2DB2
tpname QCNTEDDM mode IBMRDB security PROGRAM"
terminate
```

Note: To connect to DB2 for MVS, it is recommended that you use security PROGRAM.



If you need to change values that were set with the **catalog node** command, perform the following steps:

Step 1. Run the **uncatalog node** command in the command line processor as follows:

```
db2 uncatalog node node_name
```

Step 2. Recatalog the node with the values that you want to use.

4. Catalog the Database as a Database Connection Service (DCS) Database

To catalog the remote database as a Data Connection Services (DCS) database, perform the following steps:

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

Step 2. Enter the following commands:

```
catalog dcs db local_dcsname as target_dbname
terminate
```

where:

- *local_dcsname* represents the local name of the host or AS/400 database.
- *target_dbname* represents the name of database on the host or AS/400 database system.

For example, to make ny the local database name for DB2 Connect, for the remote host or AS/400 database called newyork, enter the following commands:

```
catalog dcs db ny as newyork
terminate
```

5. Catalog the Database

Before a client application can access a remote database, the database must be cataloged on the host system node and on any DB2 Connect workstation nodes that will connect to it. When you create a database, it is automatically cataloged on the host with the database alias (*database_alias*) the same as the database name (*database_name*). The information in the database directory, along with the information in the node directory, is used on the DB2 Connect workstation to establish a connection to the remote database.

To catalog a database on the DB2 Connect Workstation, perform the following steps.

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

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

Table 31. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (<i>database_name</i>)	The local DCS database name (<i>local_dcsname</i>) of the <i>remote</i> database, you specified this when you catalogued the DCS database directory, for example, ny.	ny	
Database alias (<i>database_alias</i>)	An arbitrary local nickname for the remote database. If you do not provide one, the default is the same as the database name (<i>database_name</i>). This is the name that you use when connecting to the database from a client.	localny	
Node name (<i>node_name</i>)	The name of the node directory entry that describes where the database resides. Use the same value for the node name (<i>node_name</i>) that you used to catalog the node in the previous step.	db2node	

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

```
. INSTHOME/sql1lib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sql1lib/db2cshrc (for C shell)
```

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

Step 4. Catalog the database by entering the following commands:

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

For example, to catalog the DCS known database *ny* so that it has the local database alias *localny*, on the node *db2node*, enter the following commands:

```
catalog database ny as localny at node db2node
authentication dcs
terminate
```



If you need to change values that were set with the **catalog database** command, perform the following steps:

Step a. Run the **uncatalog database** command:

```
uncatalog database database_alias
```

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

6. Bind Utilities and Applications to the Database Server

The steps you have just completed set up the DB2 Connect workstation to communicate with the host or AS/400 system. You must now bind the utilities and applications to the host or AS/400 database server. You need BINDADD authority to bind.

To bind the utilities and applications to the host or AS/400 database server, enter the following commands:

```
connect to dbalias user userid using password
bind path@ddcsmvs.1st blocking all sqlerror continue
messages mvs.msg grant public
connect reset
```

For example:

```
connect to NYC3 user myuserid using mypassword
bind path/bnd@ddcsmvs.1st blocking all sqlerror continue
messages mvs.msg grant public
connect reset
```

For more information about these commands, refer to the *DB2 Connect User's Guide*.

7. Test the Host or AS/400 Connection

When you have finished configuring the DB2 Connect workstation for communications, you need to test the connection to the remote database.

Enter the following command at the DB2 Connect workstation, remembering to substitute the **database_alias** value you defined in "4. Catalog the Database as a Database Connection Service (DCS) Database" on page 348:

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

For example, enter the following command:

```
connect to nyc3 user userid using password
```

The *userid* and *password* values required are those defined at the host or AS/400 system, and must be provided to you by your DB2 Administrator. For more information, refer to the *DB2 Connect User's Guide*.

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

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

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

If the connection fails, check the following items at the DB2 Connect workstation:

- ___ 1. The node was cataloged with the correct symbolic destination name (*sym_dest_name*).
- ___ 2. The node name (*node_name*) that was specified in the database directory points to the correct entry in the node directory.
- ___ 3. The database was cataloged properly, using the correct *real_host_dbname* for the database on the host or AS/400 server.

If the connection still fails after you verify these items, refer to the *Troubleshooting Guide*.

Chapter 17. Enabling Multisite Updates (Two-Phase Commit)

This section provides an overview of the multisite update function as it applies to scenarios that involve host and AS/400 database servers. It describes the products and components needed to implement PC, UNIX, and Web applications that update multiple DB2 databases in the same transaction.

Multisite update, also known as distributed unit of work (DUOW) and two-phase commit, is a function that enables your applications to update data in multiple remote database servers with guaranteed integrity. For example, a banking transaction that involves the transfer of money from one account to another in a different database server.

In such a transaction, it is critical that updates which implement debit operations on one account do not get committed unless updates required to process credits to the other account are committed as well. The multisite update considerations apply when data representing these accounts is managed by two different database servers.

DB2 products provide comprehensive support for multisite updates. This support is available for applications developed using regular SQL as well as applications that use transaction monitor (TP monitor) products that implement the X/Open XA interface specification. Examples of such TP monitors products include IBM TxSeries (CICS and Encina), IBM Message and Queuing Series, IBM Component Broker Series, IBM San Francisco Project as well as Microsoft Transaction Server (MTS), BEA Tuxedo and several others. There are different setup requirements depending on whether native SQL multisite update or TP monitor multisite update is used.

Both the native SQL and TP monitor multisite update programs must be precompiled with the `CONNECT 2 SYNCPOINT TWOPHASE` options. Both can use the SQL Connect statement to indicate which database they want to be used for the SQL statements that follow. If there is no TP monitor to tell DB2 it is going to coordinate the transaction (as indicated by DB2 receiving the `xa_open` calls from the TP monitor to establish a database connection), then the DB2 software will be used to coordinate the transaction.

When using TP monitor multisite update, the application must request commit or rollback by using the TP monitor's API, for example CICS `SYNCPOINT`, Encina `Abort()`, MTS `SetAbort()`.

When using native SQL multisite update, the normal SQL COMMIT and ROLLBACK must be used.

TP monitor multisite update can coordinate a transaction that accesses both DB2 and non-DB2 resource managers such as Oracle, Informix or SQLServer. Native SQL multisite update is used with DB2 servers only.

For a multisite update transaction to work, each of the databases participating in a distributed transaction must be capable of supporting distributed unit of work. Currently, the following DB2 servers provided DUOW support that enabled them to participate in distributed transactions:

- DB2 UDB for UNIX, OS/2, and Windows V5 or later
- DB2 for MVS/ESA V3.1 and 4.1
- DB2 for OS/390 V5.1
- DB2 Universal Database for OS/390 V6.1 or later
- DB2/400 V3.1 or later (SNA only)
- DB2 Server for VM and VSE V5.1 or later (SNA only)
- Database Server 4

A distributed transaction can update any mix of supported database servers. For example, your application can update several tables in DB2 Universal Database on Windows NT or Windows 2000, a DB2 for OS/390 database, and a DB2/400 database, all within a single transaction.

Host and AS/400 Multisite Update Scenarios that Require SPM

Host and AS/400 database servers require DB2 Connect to participate in a distributed transaction originating from PC, UNIX, and Web applications. In addition, many of the multisite update scenarios that involve host and AS/400 database servers require that the Syncpoint Manager (SPM) component be configured. When a DB2 instance is created, the DB2 SPM is automatically configured with default settings.

The need for SPM is dictated by the choice of protocol (SNA or TCP/IP) and use of a TP monitor. The following table provides a summary of scenarios that require the use of SPM. The table also shows that DB2 Connect is required for any access to the host or AS/400 from Intel or UNIX machines. In addition, for multisite updates, the SPM component of DB2 Connect is required if the access is via SNA or uses a TP monitor.

Table 32. Host and AS/400 multisite update scenarios that require SPM

TP Monitor Used?	Protocol	SPM Needed?	Product Required (choose One)	Host and AS/400 Database Supported
Yes	TCP/IP	Yes	<ul style="list-style-type: none"> • DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise - gExtended Edition 	<ul style="list-style-type: none"> • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 or later
Yes	SNA	Yes	<ul style="list-style-type: none"> • DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise - Extended Edition* <p>Note: *AIX, OS/2, Windows NT and Windows 2000 platforms only.</p>	<ul style="list-style-type: none"> • DB2 for MVS/ESA V3.1 and 4.1 • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 or later • DB2/400 V3.1 or later • DB2 Server for VM or VSE V5.1 or later

Table 32. Host and AS/400 multisite update scenarios that require SPM (continued)

TP Monitor Used?	Protocol	SPM Needed?	Product Required (choose One)	Host and AS/400 Database Supported
No	TCP/IP	No	<ul style="list-style-type: none"> • DB2 Connect Personal Edition • DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise - Extended Edition 	<ul style="list-style-type: none"> • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 or later
No	SNA	Yes	<ul style="list-style-type: none"> • DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise - Extended Edition* <p>Note: *AIX, OS/2, Windows NT and Windows 2000 platforms only.</p>	<ul style="list-style-type: none"> • DB2 for MVS/ESA V3.1 and 4.1 • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 or later • DB2/400 V3.1 or later • DB2 Server for VM and VSE V5.1 or later

Note: A distributed transaction can update any mix of supported database servers. For example, your application can update several tables in DB2

UDB on Windows NT, a DB2 for OS/390 database and a DB2/400 database all within a single transaction.

For more information about two-phase commit, as well as instructions for setting up for several popular TP monitors, refer to:

- *Administration Guide*
- *DB2 Connect Enterprise Edition for OS/2 and Windows Quick Beginnings*
- *DB2 Connect Personal Edition Quick Beginnings* (this edition does not include the DB2 Syncpoint Manager).

You can also access the DB2 Product and Service Technical Library on the World Wide Web:

1. Go to the following Web page:
<http://www.ibm.com/software/data/db2/library/>
2. Select the **DB2 Universal Database** link.
3. Search for "Technotes" using the search keywords "DDCS", "SPM", "MTS", "CICS", and "ENCINA".

Part 5. Configuring DB2 UDB as a DRDA Application Server

Chapter 18. Accessing DB2 Universal Database Servers from Host and AS/400 Applications

Host and AS/400 applications can access DB2 Universal Database data that is stored on a DB2 Workgroup Edition, DB2 Enterprise Edition or DB2 Enterprise - Extended Edition server. The following are examples of the use of this access:

Host or AS/400 Data Migration

If you are migrating data from your host or AS/400 database to a DB2 Universal Database server, you can continue to use your existing host or AS/400 applications by having them access the data from DB2 Universal Database. This can allow for staged migration from the host or the AS/400 system.

Host or AS/400 Applications Leveraging DB2 Enterprise - Extended Edition

Your host or AS/400 application can use the parallel processing power of DB2 Universal Database for CPU intensive queries.

Access Distributed Data

Your host or AS/400 application can access distributed data stored in departmental DB2 Universal Database servers.

Supported Clients

The following database products can access DB2 Universal Database servers:

- DB2 for MVS/ESA Version 3.1 (or higher)
For information on setting up the connection from DB2 for MVS/ESA to a DB2 Universal Database server, see the instructions in "Configuration Steps for DB2 Universal Database Server" on page 362.
- DB2 for OS/390 Version 5 (or higher)
For information on setting up the connection from DB2 Universal Database for OS/390 to a DB2 Universal Database server, see the instructions in "Configuration Steps for DB2 Universal Database Server" on page 362.
- DB2 for AS/400 Version 3.1 (or higher)
For information on setting up the connection from DB2 for AS/400 to a DB2 Universal Database server, refer to the *DB2 Connectivity Supplement*.
- DB2 for VM & VSE Version 5 (or higher)
For information on setting up the connection from DB2 for VM & VSE to a DB2 Universal Database server, refer to the *DB2 Connectivity Supplement*.

For information about other IBM and non-IBM products accessing DB2 Universal Database servers, contact the software support for those products.

PTFs Required

The following PTFs are required:

DB2 for MVS/ESA Version 3: UN73393

DB2 for MVS/ESA Version 4: UN75959

DB2 for OS/390 Version 5: PQ07537

DB2 for VM/ESA Version 5: VM60922; VM61072

OS/400 Version 3 Release 2: SF23270; SF23277; SF23271; SF23721; SF23985; SF23960.

Configuration Steps for DB2 Universal Database Server

This section describes the steps and conditions required to configure a DB2 Universal Database server to accept inbound client requests from host and AS/400 database clients.

Before you begin, you must determine whether your connection will use the APPC communication protocol, or the TCP/IP communication protocol, or both.

Platform	Supported Protocols
AIX	TCP/IP, APPC, APPC Multisite Update
Linux	TCP/IP
PTX	TCP/IP
Solaris	TCP/IP, APPC
OS/2	TCP/IP, APPC, APPC Multisite Update
Windows NT and Windows 2000	TCP/IP, APPC, APPC Multisite Update

Notes:

1. The protocol you choose may depend on the host or AS/400 database client version:
 - APPC is supported by all host or AS/400 database client versions
 - TCP/IP is supported on the following host or AS/400 database client versions
 - DB2 for OS/390 Version 5 or higher
 - DB2 for AS/400 Version 4 Release 2 or higher
 - DB2 for VM Version 6 or higher.
2. Considerations for Multisite Update (Two Phase Commit)

If your host or AS/400 application required Multisite Update support (two-phase commit), you should note the following:

APPC Connections (SNA)

DB2 Universal Database Enterprise Edition for OS/2, AIX, and Windows NT and DB2 Extended Enterprise Edition for AIX, and Windows NT provide support for SNA two phase commit with host and AS/400 database clients. The supported SNA stacks for use with multisite update are:

- IBM eNetwork Communications Server for AIX Version 5.0.3
- IBM eNetwork Communications Server for Windows NT Version 5.01
- IBM eNetwork Communications Server for OS/2 Version 5
- Microsoft SNA Server Version 4 Service Pack 3

TCP/IP Connections

Multisite Update is not supported with any host or AS/400 database clients. Remote Unit of Work (RUW) is supported (one phase commit)

3. To use APPC on AIX, the optional communication support component for SNA (db2_06_01.cs.sna) must also be installed.
4. To use APPC on Solaris, the optional communication support component for SNA (db2cssna) must also be installed.

Configuring DB2 Universal Database Servers for Host or AS/400 Client Access

This section gives an overview of the steps necessary to configure DB2 Universal Database to accept inbound requests from host or AS/400 database clients. The example shows how to configure a connection from a DB2 for MVS/ESA or DB2 Universal Database for OS/390 client to a DB2 Universal Database server:

1. Ensure DB2 for MVS/ESA or DB2 Universal Database for OS/390 is installed and operational on the host.

Refer to the *Connectivity Supplement* for information on the configuration required for the DB2 for MVS/ESA or DB2 Universal Database for OS/390 AR.

2. If you use APPC, ensure VTAM is installed and operational on the host.
3. Update the tables at the DB2 for MVS/ESA or DB2 for OS/390 host.

Refer to the *Connectivity Supplement* for more information.

4. Set up the DB2 Universal Database server communications, if required. (Typically, DB2 UDB server communications are set up as part of the DB2 UDB installation. However, complete details are provided in “Chapter 11. Using the Command Line Processor to Configure Server Communications” on page 137

on page 137. If you want to use Multisite Update, see “Chapter 17. Enabling Multisite Updates (Two-Phase Commit)” on page 353.)

5. Test the connection by logging onto TSO and using DB2I/SPUFLI.

Using the DB2 Universal Database Server from Host or AS/400 Clients

Connections from host or AS/400 database clients are treated like any other connection to the DB2 Universal Database server to ensure consistency in the way the maximum number of concurrent connections to a server is measured, from host, AS/400, and Universal Database clients.

The *Administration Guide* summarizes the CCSIDs that a host or AS/400 database client must use to connect to a DB2 Universal Database server.

When using APPC, a host or AS/400 database client connects to the DB2 Universal Database server by specifying the appropriate Transaction Program Name (TPN) defined at the DB2 Universal Database server. The TPN can correspond to the *tpname* parameter value in the instance’s database manager configuration file. The TPN used by the host or AS/400 database client can also be the service transaction program **x’07’6DB**. When using this TPN and if multiple DB2 Universal Database instances exist at the server, the instance which will process the **x’07’6DB** transaction program is specified by the DB2 registry value DB2SERVICETPINSTANCE. If only a single DB2 instance will be accessed by the host or AS/400 client there is no need to specify the DB2SERVICETPINSTANCE value.

Authentication

If you choose APPC as your communication protocol, the communications subsystem may restrict the types of database manager configuration authentication settings you can use at the DB2 Universal Database server. Not all communications subsystems expose the client’s password to the DB2 Universal Database server when security (program) is used. When this is the case, the database manager configuration authentication must not be set to SERVER.

If you choose APPC as your communications protocol, you may be restricted in the types of database manager configuration authentication setting you can use at the DB2 Universal Database Server. If you have the SNA Syncpoint Manager configured, you can use any available authentication (SERVER, CLIENT, DCS).

To overcome the restriction that prevents you from using authentication SERVER, you can set the database manager authentication to DCS. This allows host or AS/400 database client connections that have been authenticated by the communication subsystem to succeed. However, with this setting the DB2

Universal Database will act as if authentication SERVER is being used for remote DB2 Universal Database client connections.

Troubleshooting

The DB2 DRDA Trace utility (**db2drdat**) is provided for tracing the data flow between a host or AS/400 database client and the DB2 Universal Database Server. For more information on setting up this trace, refer to the *Troubleshooting Guide*.

DRDA Functions Supported

DRDA functions are categorized as required or optional. Table 33 identifies which functions are implemented in the AS on the DB2 Universal Database server. The subsequent table lists the supported bind options.

Table 33. Supported DRDA Functions

Description	Required (R) Optional (O)	Supported
DRDA level 1 required function	R	Yes*
Rebind	O	Yes
Describe user privileges	O	No
Describe RDB table	O	No
Interrupt RDB request	O	No
Stored procedures return multiple row result sets	O	Yes

Note: * Certain required functions are not supported.

Bind Options Supported by the DB2 DRDA Application Server

Table 34. Bind Options Supported by the DB2 DRDA Application Server

Bind Option	Value	Sup- ported	DB2 for MVS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
Package Version Name	<u>Null</u> Any other value	Yes No	VERSION			VERSION
Bind Existence Checking	<u>Object existence optional</u> Object existence required	No Yes	VALIDATE (RUN) ^b VALIDATE (BIND) ^b	<u>NOEXIST</u> EXIST	GENLVL(10, 11-40) GENLVL(00-09)	VALIDATE <u>RUN</u> VALIDATE BIND
Package Replacement Option	<u>Replacement allowed</u> Replacement not allowed	Yes No	ACTION (REPLACE) ACTION(ADD)	<u>REPLACE</u> NEW	REPLACE(*YES) REPLACE(*NO)	ACTION <u>REPLACE</u> ACTION ADD
Package Authorization Option	<u>Keep authorizations</u> Revoke authorizations	Yes No		<u>KEEP</u> REVOKE		RETAIN <u>YES</u> RETAIN NO
Statement String Delimiter (Note 2)	Apostrophe	Yes	<u>APOSTSQL</u>	<u>SQLAPOST</u>	OPTION([...] *APOSTSQL) (Note 3)	STRDEL APOSTROPHE
Statement Decimal Delimiter (Note 5)	Double quote	No	QUOTESQL	SQLQUOTE	OPTION([...] *QUOTESQL) (Note 4)	STRDEL QUOTE
	Period	Yes	<u>PERIOD</u>	<u>PERIOD</u>	OPTION([...] *PERIOD) or OPTION([...] *SYSVAL) (Note 6)	DECDEL PERIOD
	Comma	No	COMMA	COMMA	OPTION([...] *COMMA) or OPTION([...] *SYSVAL) (Note 6)	DECDEL COMMA
Date Format (Note 7)	<u>ISO</u>	Yes	DATE(ISO) (Note 8)	DATE(<u>ISO</u>)	DATEFMT(*ISO) (Note 8)	DATE TIME <u>ISO</u> (Note 9)
	USA	Yes	DATE(USA)	DATE(USA)	DATEFMT(*USA)	DATE TIME USA
	EUR	Yes	DATE(EUR)	DATE(EUR)	DATEFMT(*EUR)	DATE TIME EUR
	JIS	Yes	DATE(JIS)	DATE(JIS)	DATEFMT(*JIS)	DATE TIME JIS
Time Format (Note 7)	<u>ISO</u>	Yes	TIME(ISO) (Note 8)	TIME(<u>ISO</u>)	TIMEFMT(*ISO) (Note 8)	DATE TIME <u>ISO</u> (Note 9)
	USA	Yes	TIME(USA)	TIME(USA)	TIMEFMT(*USA)	DATE TIME USA
	EUR	Yes	TIME(EUR)	TIME(EUR)	TIMEFMT(*EUR)	DATE TIME EUR
	JIS	Yes	TIME(JIS)	TIME(JIS)	TIMEFMT(*JIS)	DATE TIME JIS

Table 34. Bind Options Supported by the DB2 DRDA Application Server (continued)

Bind Option	Value	Sup- ported	DB2 for MV/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
Package Isolation Level (Note 10)	Repeatable read	Yes	ISOLATION(<u>RR</u>) ^b	ISOLATION(<u>RR</u>)		ISOLATION RR
	Read Stability (All)	Yes		ISOLATION(RS)	COMMIT(*ALL)	ISOLATION RS
	Cursor stability	Yes	ISOLATION(CS) ^b	ISOLATION(CS)	COMMIT(*CS)	ISOLATION <u>CS</u>
	Uncommitted Read (Change)	Yes		ISOLATION(UR)	COMMIT(* <u>CHG</u>)	ISOLATION UR
	No commit	No (Note 11)			COMMIT(*NONE)	ISOLATION NC
Bind Creation Control	<u>No errors allowed</u>	Yes	SQLERROR (<u>NOPACKAGE</u>) ^b	<u>NOCHECK</u>	OPTION([...] *GEN) GENLVL(00-09, <u>10</u> , 11-20)	SQLERROR <u>NOPACKAGE</u>
	Check only	Yes		CHECK	OPTION([...] *NOGEN)	SQLERROR CHECK
	Errors allowed	No	SQLERROR (CONTINUE) ^b	ERROR	OPTION([...] *GEN) GENLVL(21-40)	SQLERROR CONTINUE
Bind Explain Option	<u>No SQL statements</u>	Yes	EXPLAIN(<u>NO</u>) ^b	EXPLAIN(<u>NO</u>)		EXPLAIN <u>NO</u>
	All explainable SQL statements	No	EXPLAIN(YES) ^b	EXPLAIN(YES)		EXPLAIN YES
Package Owner Identifier	< <u>Authorization ID</u> >	Yes	OWNER ^b	OWNER		OWNER
	Any other value	No				
RDB Release Option	<u>Release at commit</u>	Yes	RELEASE (<u>COMMIT</u>) ^b	RELEASE (<u>COMMIT</u>)		RELEASE <u>COMMIT</u>
	Release at conversation deallocation	No	RELEASE (DEALLOCATE) ^b	RELEASE (DEALLOCATE)		RELEASE DEALLOCATE
Default RDB Collection ID	< <u>Authorization ID</u> >	Yes	QUALIFIER ^b	QUALIFIER	DFTRDBCOL	QUALIFIER
	Any other value	No				
Title (Package Description)	Any value (ignored by DB2)	Yes		LABEL	TEXT	TEXT
Query Block Protocol Control	<u>Fixed row</u>	Yes	CURRENTDATA (<u>YES</u>) ^b	SBLOCK	ALWBLK(* <u>READ</u>)	BLOCKING <u>UNAMBIG</u>
	Limited block	Yes	CURRENTDATA (NO) ^b	BLOCK	ALWBLK (*ALLREAD)	BLOCKING ALL
	Forced fixed row	Yes		<u>NOBLOCK</u>	ALWBLK(*NONE)	BLOCKING NO
Package Default Char. Subtype	<u>Use system default</u>	Yes				CHARSUB <u>DEFAULT</u>

Table 34. Bind Options Supported by the DB2 DRDA Application Server (continued)

Bind Option	Value	Sup- ported	DB2 for MVS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
If Default CCSID is SBCS	BIT	No		CHARSUB(BIT)		CHARSUB BIT
If Default CCSID is SBCS	SBCS	Yes		CHARSUB(SBCS)		CHARSUB SBCS
If Default CCSID is SBCS	MBCS	No		CHARSUB(MBCS)		CHARSUB MBCS
If Default CCSID is MBCS	BIT	No		CHARSUB(BIT)		CHARSUB BIT
If Default CCSID is MBCS	SBCS	No		CHARSUB(SBCS)		CHARSUB SBCS
If Default CCSID is MBCS	MBCS	Yes		CHARSUB(MBCS)		CHARSUB MBCS
	Any other value	No				
Package Default CCSID	<u>Value specified when DB2 database was created</u>	Yes		CCSID(SBCS) CCSID(GRAPHIC()) CCSID(MIXED())		CCSID CCSIDG CCSIDM
	Any other value	No				
Decimal Precision (Note 12)	31	Yes	DEC(31)			DEC 31
	Any other value	No	DEC(15)			DEC 15
Replaced Package Version Name	<u>Null</u>	Yes	REPLVER ^b			REPLVER
	Any other value	No				
Generic Bind Option	<u>Null</u>	No				GENERIC
	Any other value	No				
Package Authorization Rule	<u>Requester</u>	Yes				DYNAMICRULES RUN
	Owner	No				DYNAMICRULES BIND
	Creator of the user-defined function and stored procedure	No				DYNAMICRULES DEFINE
	Invoker of the user-defined function and stored procedure	No				DYNAMICRULES INVOKE
Degree of Parallelism	<u>1</u>	No				DEGREE 1
	n	No				DEGREE n
	ANY	No				DEGREE ANY

Table 34. Bind Options Supported by the DB2 DRDA Application Server (continued)

Bind Option	Value	Sup- ported	DB2 for MVS/ESA Precompile Option (Note 1)	DB2/VM Preprocessing Option	OS/400 Precompile Option	DB2 Prep or Bind Option
-------------	-------	-------------	--	-----------------------------	--------------------------	-------------------------

Note:

(*) Default values are in **bold**. (1) Most are precompile options. Bind options are indicated by ^b. (2) Defaults to what the target database supports. For DB2 the default is apostrophe. (3) Default for non-COBOL applications. (4) Default for COBOL applications. (5) Defaults to what the target database supports. For DB2 the default is period. (6) Depending on the installation, *SYSVAL is equivalent to *PERIOD or *COMMA. (7) Date and time formats must be the same for the DB2 DRDA AS. (8) Default is dependent on the installation. (9) Format applies to both date and time. If not specified, it defaults based on the country code. This default is mapped to ISO in DRDA flow. (10) Package isolation level has no default because an explicit value is always present in the DRDA datastream. (11) The isolation level will be escalated to Uncommitted Read (Change). (12) Defaults to what the target database supports. For DB2 the default is 31. (13) All variables will default to 1.

Special Considerations for DB2 for VM (SQL/DS)

Additional steps are needed to ensure the following DB2 for VM utilities work properly when accessing a DB2 Universal Database server.

- SQLDBSU
 1. Make sure a PTF for DB2 for VM APAR PN69073 is installed on your DB2 for VM system, either PTF UN91171 or PTF UN91172. (At the time of publication of this manual, there were no PTFs for DB2 for VM V4 or V5.)
 2. Set up dummy tables in your DB2 database by executing the **sqldbsu** utility provided with DB2 as `sqldbsu database_name`.
 3. Bind SQLDBSU from DB2 for VM. Refer to the “Using a DRDA Environment” section in the *SQL/DS System Administration for IBM VM Systems* manual for details. (You can skip the step on creating and populating the SQLDBA.DBSOPTIONS table because this is done by the **sqldbsu** utility in the previous step.)
- ISQL
 1. Complete the steps described above for SQLDBSU.
 2. Set up dummy tables in your DB2 database by executing the **isql** utility provided with DB2 as `isql database_name`.
 3. Bind ISQL from DB2 for VM. Refer to the “Using a DRDA Environment” in the *SQL/DS System Administration for IBM VM Systems* manual for details.

Notes:

1. On UNIX workstations, the **sqldbsu** and **isql** utilities are in `INSTHOME/sql11ib/misc` where *INSTHOME* represent the home directory of the instance owner.
2. On OS/2 and Windows, the **sqldbsu** and **isql** utilities are in `DB2PATH\misc`, for example:
`c:\SQLLIB\misc\`

If you installed DB2 on drive C, using the default directory `sql11ib`, then no special setup is required for RXSQL; refer to the *SQL/DS Procedures Language Interface Installation* manual for details.

Security and Auditability

Under APPC, DB2 Universal Database system security (authentication CLIENT, SERVER, or DCS) must be used with a APPC security SAME or PROGRAM. When these combinations are used, the user ID and password sent by the host or AS/400 is used to CONNECT to the requested database. APPC security level NONE is allowed only with DCE authentication. In this case, the encrypted DCE ticket is flown as part of the CONNECT attempt.

Under TCP/IP, all security information is flown in the CONNECT attempt.

User ID translation is not supported by DB2 Universal Database.

Configuration Considerations

Data access by host and AS/400 applications are tuned at the DB2 Universal Database server using the DB2 database manager configuration parameters. One parameter, DRDA Heap Size, is specifically for host and AS/400 database client connections. You may need to change the setting for some parameters because of the additional resources required for the DB2 Universal Database Server.

DRDA Heap Size (drda_heap_sz)

On UNIX workstations, DRDA heap size specifies the amount of memory, in pages, that is allocated for use by the DB2 Universal Database server for host and AS/400 connections.

ON OS/2 or Windows, DRDA heap size specifies the amount of memory, in segments, that is allocated for use by the DB2 Universal Database server for host and AS/400 connections.

Refer to the *Administration Guide* for more information about database manager configuration.

Part 6. Distributed Installation

Chapter 19. An Introduction to Distributed Installation

If you are planning to install DB2 products across your network, consider using a network-based distributed installation. With a network-based installation, you can roll out multiple, identical copies of DB2 products.

Types of Distributed Installation

DB2 products can be installed using systems management software, such as Microsoft Systems Management Server (SMS) on Windows NT or Windows 2000, or simply with a shared CD-ROM drive or shared network hard drive using response files.



We recommend installing from a network hard drive rather than a CD-ROM, especially if you use the CD-ROM drive for other tasks. Installing from a network CD-ROM drive will significantly increase the amount of time it will take to perform the install.

Response File

What is a response file?

The first step in any type of distributed installation is the creation of a response file. A response file is an ASCII file that can be customized with the setup and configuration data that will automate an installation. The setup and configuration data would have to be entered during a interactive install, but with a response file, the installation can proceed without any intervention.

A response file specifies such configuration and setup parameters as the destination directory, and the products and components to install. It can also be used to set up the following settings:

- Global DB2 registry variables
- Instance variables
- Instance database manager configuration settings

You can use a response file to install an identical configuration across every workstation on your network or to install multiple configurations of a DB2 product. For example, you can customize a response file which will install a DB2 Administration Client. You can then distribute this file to every workstation where you want this product to be installed.

Available Sample Response Files

The DB2 CD-ROM includes a ready-to-use sample response files with default entries. The sample response files are located in:

For Windows:

`x:\db2\common` or `x:\db2\winnt95\common` where *x* represents the CD-ROM drive.

For OS/2:

`x:\db2\[lang]` where *x* represents the CD-ROM drive and *[lang]* represents the two-character country code that represents your language (for example, EN for English).

For UNIX:

`/cdrom/db2/install/samples` where *cdrom* is the CD-ROM mount point.

You can use the following sample response files to install DB2 products on supported workstations:

- db2admcl.rsp** DB2 Administration Client
- db2sdk.rsp** Application Development Client
- db2conee.rsp** DB2 Connect Enterprise Edition
- db2conpe.rsp** DB2 Connect Personal Edition
- db2dlm.rsp** DB2 Data Links Manager
- db2wagt.rsp** Data Warehouse Agent (UNIX-based operating systems only)
- db2udbwm.rsp**
DB2 Warehouse Manager
- db2relc.rsp** DB2 Relational Connect
- db2udbpe.rsp** DB2 Universal Database Personal Edition
- db2rtcl.rsp** DB2 Run-Time Client
- db2udbse.rsp** DB2 Universal Database Satellite Edition
- db2udbwe.rsp** DB2 Universal Database Workgroup Edition
- db2eee.rsp** For Windows NT and Windows 2000, the DB2 Universal Database instance-owning database partition server response file. This response file is used to install DB2 on a machine that will be the instance-owning database partition server.
- db2udbeee.rsp**
For UNIX, the database partition server response file. This response file is used to install DB2 on a machine that will be a database partition server.

db2eeenn.rsp	The DB2 Universal Database new node response file. This response file is used to add a new node to an existing database partition system (Windows NT and Windows 2000 only).
db2eeesp.rsp	The DB2 Universal Database single partition database response file. This response file is used to migrate an existing single-partition instances to a Version 6 single-partition instances (Windows NT and Windows 2000 only).
db2osk.rsp	OLAP Starter Kit
db2qp.rsp	Query Patroller (Windows NT and Windows 2000 only).
db2qpa.rsp	Query Patroller Agent (UNIX-based operating systems only)
db2qpc.rsp	Query Patroller Client (UNIX-based operating systems only)
db2qps.rsp	Query Patroller Server (UNIX-based operating systems only)
db2gsec.rsp	Spatial Extender Client
db2gse.rsp	Spatial Extender Server

Important Response File Keywords

This section describes the most important keywords you will specify when performing a distributed installation. You can use response file keywords to specify the values of a database manager configuration parameters, installation components and the values of DB2 registry variables. The following areas are covered in this section:

- “Response File Keywords for OS/2 and Windows 32-Bit Operating Systems”
- “Response File Keywords for DB2 Satellite Edition” on page 381
- “DB2 Control Server Response File Keywords for Windows NT and Windows 2000” on page 383

Response File Keywords for OS/2 and Windows 32-Bit Operating Systems

This sections describes the most important keywords that you will specify when performing a distributed installation on OS/2 and Windows 32-bit operating systems. The following keywords are available for all DB2 products including Satellite. For keywords specific to the DB2 Satellite Edition, see “Response File Keywords for DB2 Satellite Edition” on page 381.

FILE Specifies the destination directory for a DB2 product.

REBOOT

Specifies whether to restart the system when the installation has completed (Windows 32-bit operating systems only).

For OS/2 systems, enter **/REBOOT** on the install command line.

TYPE Specifies the type of install (Windows 32-bit operating systems only).

The options are:

- 0 = Compact
- 1 = Typical (default)
- 2 = Custom

Note: A compact or typical install type will ignore any custom keywords (COMP).

KILL_PROCESSES

(Windows 32-bit operating systems only).

If you have an existing version of DB2 and it is running and this keyword is set to YES, it will terminate your running DB2 processes without prompt. For more information on killing DB2 processes, see “Killing DB2 Processes During Interactive and Response File Installations” on page 384.

PROD Specifies the product you want installed. The options are:

- ADMIN_CLIENT for DB2 Administration Client
- CONNECT_PERSONAL for DB2 Connect Personal Edition
- CONNECT_ENTERPRISE for DB2 Connect Enterprise Edition
- DATA_LINKS_MANAGER for DB2 Data Links Manager
- DB2_QP_AGENT for DB2 Query Patroller Agent (UNIX-based operating systems only)
- DB2_QP_CLIENT for DB2 Query Patroller Client (UNIX-based operating systems only)
- DB2_QUERY_PATROLLER_SERVER for DB2 Query Patroller Server (Windows 32-bit operating systems only)
- DB2_QP_SERVER for DB2 Query Patroller Server (UNIX-based operating systems only)
- OLAP_STARTER_KIT for DB2 OLAP Starter Kit
- RELATIONAL_CONNECT for DB2 Relational Connect
- RUNTIME_CLIENT for DB2 Run-Time Client
- SDK for DB2 Application Development Client
- SPATIAL_EXTENDER_CLIENT for DB2 Spatial Extender Client
- SPATIAL_EXTENDER_SERVER for DB2 Spatial Extender Server
- UDB_EEE for DB2 Enterprise - Extended Edition
- UDB_ENTERPRISE for DB2 Enterprise Edition
- UDB_PERSONAL for DB2 Personal Edition
- UDB_SATELLITE for DB2 Satellite Edition

- UDB_WORKGROUP for DB2 Workgroup Edition
- WAREHOUSE_AGENT for DB2 Data Warehouse Agent (UNIX-based operating systems only)
- WAREHOUSE_MANAGER for DB2 Data Warehouse Manager

DB2.AUTOSTART

Specifies whether or not to automatically start the DB2 instance each time the system is rebooted.

By default, the DB2 instance starts automatically unless this parameter is set to NO.

AUTOSTART_CCA

Specifies whether or not to automatically start the Client Configuration Assistant each time the system is rebooted.

By default, the Client Configuration Assistant starts automatically unless this parameter is set to NO.

AUTOSTART_CONTROL_CENTER

Specifies whether or not to automatically start the Control Center each time the system is rebooted.

By default the Control Center will be automatically started unless this parameter is set to NO.

AUTOSTART_FIRST_STEPS

Specifies whether or not to automatically start the First Steps application the first time the server system is rebooted. After the initial install, you can control First Steps through its interface and request it to come up on subsequent server system reboots.

By default, First Steps will be automatically started. When installing on remote systems, you can set the parameter to NO so that First Steps does not start.

CFGUPDATE

Specifies whether the config.sys file is automatically updated. (OS/2 operating systems only). Valid values for this keyword are:

AUTO

Automatically updates CONFIG.SYS.

MANUAL

Does not update CONFIG.SYS.

We recommend that you specify **AUTO** when performing a distributed install.

DB2SYSTEM

Specifies a name for the system which is unique within a network.

ADMIN.USERID and ADMIN.PASSWORD

Specifies the user ID and password that will be used to log on and start the Administration Server each time your system is started.

Note: Not available on Windows 9x. Available on the following DB2 products: UDBEEE, UDBEE, UDBWE, CONNEE and UDBPE.

On OS/2, if UPM already exists on your system, the user ID and password you provide must exist and have one of the following:

- UPM administrator authority on your system.
- UPM local administrator authority on your system.

If UPM is not on your system, it will be installed as part of the DB2 installation and the user ID and password you provide will be set up with the appropriate authority.

On Windows, if the username entered does not exist on the user's machine at install time, it will be created with the appropriate authorities by the install program. If the user entered exist at install time, the user ID must be part of the local administrators group. The install will verify this during the installation process, so the install will give an error message if the username does not have the required authority.

DB2.USERID and DB2.PASSWORD

Specifies the default DB2 instance's username and password. Used by the DB2 instance to logon to the system each time your system is started.

Note: Not available on Windows 9x. Available on the following DB2 products: UDBEEE, UDBEE, UDBWE, CONNEE, UDBBE and UDBSE.

On Windows, if the username entered does exist on the user's machine at install time, it will be created with the appropriate authorities by the install program. If the user entered does exist at install time, the user ID must be part of the local administrators group. The install will verify this during the installation process, so the install will give an error message if the username does not have the required authority.

DB2CTLSV.USERID and DB2CTLSV.PASSWORD

Specifies the default Control Server instance's username and password. Used by the DB2 instance to logon to the system each time your system is started.

Note: Available on UDBEE only.

On Windows, if the username entered does not exist on the user's machine at install time, it will be created with the appropriate authorities by the install program. If the user entered does exist at install time, the user ID must be part of the local administrators group. The install will verify this during the installation process, so the install will give an error message if the username does not have the required authority.

DLFM_INST_USERID and DLFM_INST_PASSWORD

Specifies the default Data Links Manager instance's username and password. Used by the DB2 instance to logon to the system each time your system is started.

Note: Available for Data Links Manager only.

On Windows, if the username entered does not exist on the user's machine at install time, it will be created with the appropriate authorities by the install program. If the user entered does not exist at install time, the user ID must be part of the local administrators group. The install will verify this during the installation process, so the install will give an error message if the username does not have the required authority.

COMP

Specifies the components you want installed. The setup program automatically installs components that are required for a product, and ignores requested components that are not available.

For Windows 32-bit operating systems, component selections have no effect unless you specify a custom installation (TYPE = 2).

Response File Keywords for DB2 Satellite Edition

This section describes the most important keywords that you will specify when performing a distributed installation of DB2 Satellite Edition on Windows 32-bit operating systems.

Note: The response file keywords listed below are unique to DB2 Satellite Edition.

DB2.AUTOSTART

Specifies whether or not to automatically start the DB2 instance each time the system is rebooted.

By default, the DB2 instance starts automatically unless this parameter is set to NO.

DB2.SATCTLDB_USERNAME and DB2.SATCTLDB_PASSWORD

Specifies the user ID and password that will be used by the satellite when connecting to the satellite control database (SATCTLDB) on the

DB2 control server. The user ID and password are used to authenticate the connection to the database. It is not mandatory that you enter this information at installation time, but it is recommended that you do so if you have the information. The user ID and password cannot be authenticated at installation time.

If you choose not to provide this information at installation time, you can do so later by running the DB2 synchronization application in test mode by issuing the **db2sync -t** command. You will then be prompted for the user ID and password required to make the connection.

DB2.DB2SATELLITEID

Specifies the unique ID for the satellite and sets the DB2SATELLITEID registry variable on the satellite. The ID must be unique across all groups that are recorded at the DB2 control server. It must match an ID defined for a satellite at the control server. The satellite ID is used during the synchronization process to identify the satellite. The ID can be a maximum of 20 characters.

It is not recommended that you provide the DB2SATELLITEID in the response file as it must be unique, unless you are customizing the value of DB2SATELLITEID for each system the response file will be used on. The DB2SATELLITEID can be set after the installation using the **db2set** command.

If not specified, the Windows login ID is used in its place during the synchronization process.

DB2.DB2SATELLITEAPPVER

Specifies the satellite's application software version. The version can be a maximum of 18 characters and numbers. The value specified must match an application version defined for the group to which the satellite belongs as defined at the satellite control server. If it does, then scripts associated with this application version will be used to maintain the satellite during the synchronization process. A default version V1R0M00 has been supplied, but this value can be changed. These values can be set or changed after installation.

DB2.USERDB_NAME

Specifies the name for the database that DB2 can create during the installation of DB2 Satellite Edition. If no value is provided, the database is not created.

DB2.USERDB_REP_SRC

Specifies that the database will be used as a DB2 replication source. DB2 will configure the database so that changes to application data can be written to change tables by the Capture program. The Apply program will then use the captured changes to synchronize application data with other systems. In addition to configuring the

database to capture data changes, you must define the application tables for which changes will be gathered. For more information on the *data capture changes* parameter of the CREATE TABLE statement, refer to the *SQL Reference*. This configuration step can be done when the installation process is complete and the application tables are defined in the database.

DB2.USERDB_RECOVERABLE

Specifies that the database on the satellite is recoverable. DB2 will configure the database for forward recovery by setting the *logretain* parameter to recovery. You will be required to manage the database log files and take backups of the database. Before the database can be used, you will have to back it up. If this keyword is not specified, the database will not be configured for forward recovery. Database log files will be managed automatically by DB2. You will not be required to take a backup copy of the database before the database can be used. However, data can be lost if a disk failure occurs.

DB2 Control Server Response File Keywords for Windows NT and Windows 2000

This section describes the most important keywords that you will specify when performing a distributed installation of the DB2 Control Server. The DB2 Control Server provides administrative and status reporting support for satellites by using the satellite control database SATCTLDDB. This database is automatically created when the Control Server component is installed. These keywords can be used to specify the values of database manager configuration parameters and the values of the DB2 registry variables.

To install the Control Server, select the CONTROL_SERVER component (COMP=CONTROL_SERVER), which is only available on UDBEE.

CTLSRV.DEDICATED_CTLSRV

Specifies whether the system on which the DB2 Control Server will be installed and will be dedicated to the DB2 Control Server. Please note, a different DB2 instance will not be created on this system.

The default is YES, the system is dedicated.

CTLSRV.AUTOSTART

Specifies whether or not to automatically start the DB2 Control Server instance (DB2CTLSV) each time the system is rebooted.

The default is YES, the DB2CTLSV instance starts automatically.

CTLSRV.SVCENAME

Specifies the DB2 Control Server instance, TCP/IP service name and can be used to override the default service name generated by the installation program. When used in conjunction with the

CTLSRV.PORT_NUMBER keyword to override the default port number, you have complete control over the TCP/IP configuration for the DB2 Control Server instance.

CTLSRV.PORT_NUMBER

Specifies the DB2 Control Server instance, TCP/IP service name and can be used to override the default service name generated by the installation program. When used in conjunction with the CTLSRV.SVCENAME keyword to override the default port number, you have complete control over the TCP/IP configuration for the DB2 Control Server instance.

Killing DB2 Processes During Interactive and Response File Installations

If any DB2 processes are running when the DB2 setup command is issued, the installation of DB2 cannot occur. For example, during an interactive installation, the following message is issued: DB2 is currently running and locked by the following process(es)

The user is then prompted to kill the DB2 processes so that the installation can proceed. You can specify that any running DB2 processes are killed when the DB2 setup command is issued (Windows 32-bit operating systems only).

To kill any running DB2 processes for an interactive installation, specify the /F option for the setup command. The /F option kills the running processes, and the message and prompt are not displayed.

For a response file installation, you can use either of the following methods to kill any active DB2 processes. If you specify either of these options, the active DB2 processes are killed before the installation proceeds.

- Specify the /F option for the setup command. You can use this option along with the /U, /L and /I options that are already available.
- Set the KILL_PROCESSES keyword to YES (the default is NO).

Note: You should exercise extreme caution when you kill active DB2 processes so that an installation can occur. The termination of a DB2 process can cause the loss of data.

Response File Generator

The response file generator utility creates a response file from an existing installed and configured DB2 product. You can use the generated response file to recreate the exact setup on other machines.

For example, you could install and configure a DB2 Run-Time client to connect to various databases across your network. Once this DB2 client is

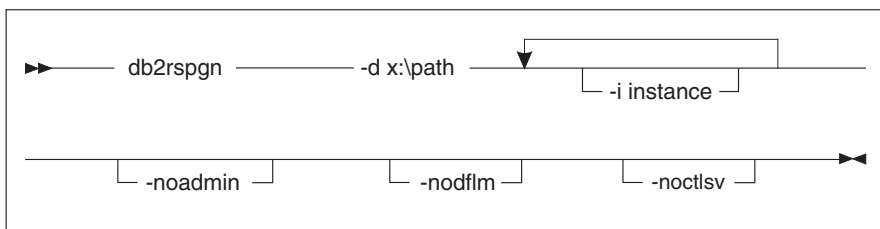
installed and configured to access all the databases that your users to have access to, you can run the response file generator to create a response file and a profile for each instance.

The response file generator creates a response file for the installation and instance profiles for each instance that you specify. You can then use the response file to create identical clients across your network.

The response file generator also gives you the option to create just the installation response file, without an instance profile, which would allow you to create identical copies of your installed client, without that configuration information.

Note: The response file generator is only available on OS/2 and Windows 32-bit operating systems.

The syntax for the **db2rspgn** command is as follows:



- d** Destination directory for a response file and any instance files. This parameter is required.
- i** A list of instances for which you want to create a profile. The administration instance (DB2DAS00) does not need to be specified. The default is to generate an instance profile file for all instances. This parameter is optional.
- noadmin**
Disables the saving of the administration instance (DB2DAS00). The administration instance will then be created with the standard defaults. The default is to save the administration instance. This parameter is optional.
- nodflm**
Disables the saving of the DLFM instance. This parameter applies only to a Data Links system. This parameter is optional.

For example, to create a directory called db2rsp at the base of the current drive, and have the response file generator place the response file and the instance profiles for all the instances in this directory, enter the following command:

```
db2rspgn -d \db2rsp
```

Each instance would have a profile created.

You can create the same directory as the first example, but only include the response files for instances `inst1`, `inst2`, and `inst3`, by entering the following command:

```
db2rspgn -d \db2rsp -i inst1 -i inst2 -i inst3
```

If you are planning to set up and configure identical DB2 products, you only need to specify the installation response file when you perform the installation. The installation response file that was created by the response file generator will automatically call each instance profile. You only need to ensure that the instance profiles are located in the same drive and directory as the installation response file.

Where Do You Go From Here?



Go to the section which details the distributed installation on your platform:

- “Chapter 20. Distributed DB2 Installation on Windows 32-Bit Operating Systems” on page 387
 - “Chapter 21. Distributed DB2 Installation on UNIX Operating Systems” on page 399
 - “Chapter 22. Distributed DB2 Installation on OS/2 Operating Systems” on page 403
-

Chapter 20. Distributed DB2 Installation on Windows 32-Bit Operating Systems

This section describes how to perform a distributed installation on Windows 32-Bit operating systems.

Before You Begin

Before you begin the installation, ensure that you have the following items and information:

- 1. Ensure that your system meets all of the memory, hardware, and software requirements to install your DB2 product. For more information, see “Chapter 1. Planning for Installation” on page 3.
- 2. You have all of the required user accounts to perform the installation. For more information, refer to the appropriate *Quick Beginnings* manual. For information on the requirements to install a DB2 Administration Client, DB2 Run-Time Client, or a Application Development Client, see “Chapter 2. Installing DB2 Clients” on page 11.

Make DB2 Files Available for Installation

The DB2 install files must be accessible across your network. To copy the required files from the CD-ROM to the shared network drive that will act as the install server, perform the following steps:

Step 1. Insert the appropriate CD-ROM into the drive.

Step 2. Create a directory by entering the following command:

```
md c:\db2prods
```

Step 3. Enter the **cpyssetup.bat** command to copy the DB2 installation files to your install server. This command is located in the *x:\db2\common* directory, where *x*: represents your CD-ROM drive.

The command syntax is as follows:

```
cpyssetup.bat directory language
```

where:

- *directory* represents the directory that you created in the previous step (for example, *c:\db2prods*).
- *language* represents the two-character country code for your language (for example, *en* for English). Table 39 on page 506 lists the keywords for each available language.

For example, to copy all of the English DB2 install files to the c:\db2prods directory, enter the following command:

```
copysetup.bat c:\db2prods en
```

Set up Shared Access

This section will allow you to grant your network workstations access to the code server. From the code server, perform the following steps:

Step 1. Click on **Start** and select **Programs—>Windows Explorer**.

Step 2. Select the directory that you want to share. For example, c:\db2prods.

Step 3. Select **File—>Properties** from the menu bar. The properties window for the directory will open.

Step 4. Select the **Sharing** tab.

Step 5. Select the **Shared As** radio button.

Step 6. In the **Share Name** field, enter a share name. For example, db2nt.

Step 7. To specify *Read access* for everyone:

- a. Click the **Permissions** push button. The Access Through Share Permissions window opens.
- b. Ensure that the **Everyone** option is selected in the **Name** box.
- c. Click the **Type of Access** drop down box and select the **Read** option.
- d. Click **OK**. You are returned to the properties window of the directory for which you want to set up shared access.
- e. Click **OK**.

In our example, c:\db2prods uses a share name of db2nt. We will use *codesrv* to represent the name of the computer on which the DB2 install files were installed. These values will be used in the examples that follow.

Create a Response File



If you have already set up and configured a DB2 product and you want to distribute this exact configuration across your network, we recommend that you use the response file generator to create the response file for your installation. For more information on creating a response file see, “Response File Generator” on page 384.

If you have already generated a response file using the response file generator, go to “Run Setup with the Response File from the Client Workstation” on page 390.

The DB2 CD-ROM includes a ready-to-use sample response file with default entries. The sample response files are located in `x:\db2\common` directory, where `x`: represents the CD-ROM drive.

Response files are available for each DB2 product, see “Available Sample Response Files” on page 376 for more information.

To edit the appropriate sample response file, perform the following steps:

Step 1. Customize the response file.

To activate an item in the response file, remove the asterisk (*) to the left of the keyword. Then, replace the current setting to the right of the value with the new setting. The possible settings are listed to the right of the equal sign.

Keywords that are unique to installation are only specified in a response file during an distributed installation. For a list of installation keywords, see “Important Response File Keywords” on page 377.

Step 2. Save the file. If you have made any changes, save the file under a new file name to preserve the original sample response file. If you are installing directly from the CD-ROM, you store the renamed response file on another drive.

For example, the following response file would install a DB2 Administration Client on the `c:\sql11ib` directory, with the REBOOT and the catalog NO AUTHORIZATION options enabled:

```
⋮
FILE                = c:\sql11ib
TYPE                = 2
PROD                = ADMIN_CLIENT
REBOOT              = YES
⋮
DB2.CATALOG_NOAUTH = YES
⋮
```

If you specify the `DB2.CATALOG_NOAUTH=YES` keyword, users will not be required to have System Administrative (SYSADM) or System Controller (SYSCTRL) authority to catalog databases. This is the default setting with DB2 Client and DB2 Connect Personal Edition response files.

For more information on this parameter or other configuration parameters, refer to the *Administration Guide*.



Install DB2 products only on a drive which is local to the target workstation. Installing on a non-local drive can cause performance and availability problems.

Run Setup with the Response File from the Client Workstation



If you are planning to deploy your DB2 product across a network using Microsoft's System Management Server (SMS), go to "DB2 Product Installation Using SMS" on page 392.

To perform an installation from the workstation where the DB2 products will be installed, perform the following steps:

Step 1. Log on to the system with a user account that you want to use to perform the installation. For more information, see "Before You Begin" on page 387.

Step 2. Connect to the shared directory of the network drive or CD-ROM drive by entering the following command from the command prompt:

```
net use x: \\computer_name\directory_sharename /USER:domain\username
```

where:

- *x*: represents the shared directory on the local drive.
- *computer_name* represents the computer name of the remote machine where the DB2 install files reside.
- *directory_sharename* represents the share name of the directory on network drive or CD-ROM drive where the DB2 install files reside.
- *domain* represents the domain where the account is defined.
- *username* represents a user that has access to this machine.

For example, to use the remote db2prods directory, which was shared as db2nt and is located on the remote server codesrv, as the local *x*: drive, enter the following command:

```
net use x: \\codesrv\db2nt
```

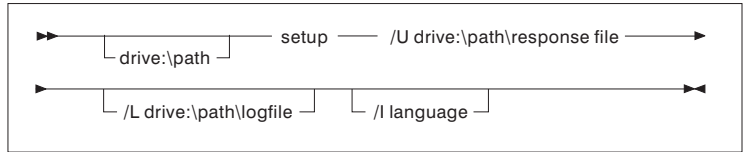


Depending on how security is set up across your network, you may have to specify the */USER* parameter.

Step 3. Run the setup program by performing the following steps:

Step a. Click on **Start** and select the **Run** option. The Run window opens.

Step b. In the **Open** field, enter the path to the setup program. The syntax of the setup command is as follows:



where:

/U Specifies the fully qualified response file name. If you changed and renamed the sample response file that is provided, make sure that this parameter matches the new name. This parameter is required.

/L Specifies the fully qualified log file name, where setup information and any errors occurring during setup are logged. This parameter is optional.

If you do not specify the log file's name, DB2 names it `db2.log`. DB2 then stores it in the `db2log` directory on the drive on which your operating system is installed.

/I Specifies the two-character country code that represents your language. If you do not specify the language, setup will determine the system language, and launch the appropriate DB2 install for that language. This parameter is optional.

For more information on country codes, see Table 39 on page 506.

For example, to install a DB2 Administration Client using a custom response file that you created called `admin.rsp` (located in the same directory as the DB2 install files), enter the following command:

```
x:\setup /U admin.rsp
```

If you are using a response file that was created using the response file generator, ensure that all the instance profiles are located in the same drive and directory as the response file that you specify.

Step c. Click on **OK** to start the setup program. The installation proceeds without further action on your part.

Step 4. Check the messages in the log file when the installation finishes.



DB2 Product Installation Using SMS

With Microsoft Systems Management Server (SMS), you can install DB2 across a network, and set up the installation from a central location. An SMS install will minimize the amount of work the users will have to perform. This installation method is ideal if you want to roll out an installation on a large number of clients all based on the same setup.

DB2 product installation using SMS involves three steps:

Step 1. “Import the DB2 Install File into SMS on the SMS Server”

Step 2. “Create the SMS Package on the SMS Server” on page 393

Step 3. “Distribute the DB2 Installation Package from the SMS Server” on page 394

When you are using SMS, you have control over which response file you will use. You can have several different installation options, resulting in several different response files. When you configure the SMS install package, you can specify which response file to use.

SMS Requirements

You must have at least SMS Version 1.2 installed and configured on your network for both your SMS server and SMS workstation. Refer to *Microsoft’s Systems Management Server Administrator’s Guide* for your platform for more details on how to:

- Set up SMS (including setting up primary and secondary sites).
- Add clients to the SMS system.
- Set up inventory collection for clients.

Import the DB2 Install File into SMS on the SMS Server

To set up a package through SMS, you will use the sample SMS package description (**db2.pdf**) file and your customized response file and instance profile.



If you are using a response file that was created using the response file generator, you must ensure that all the instance profiles are located in the same drive and directory as the response file that you specify.

To import the DB2 install files into SMS, perform the following steps:

- Step 1. Insert the appropriate CD-ROM into the drive.
- Step 2. Click **Start** and select **Programs—>Systems Management Server—>SMS Administrator**.
- Step 3. The **Microsoft SMS Administrator Logon** window opens, enter your logon ID and password, and click **OK**. The **Open SMS Window** window opens.
- Step 4. Select the **Packages** window type and click **OK**. The **Packages** window opens.
- Step 5. Select **File—>New** from the menu bar. The **Package Properties** window opens.
- Step 6. Click the **Import** push button. The **File Browser** opens. Find the db2.pdf file located in x:\db2\winnt95\common\, where x: represents the CD-ROM drive.
- Step 7. Click **OK**.

Create the SMS Package on the SMS Server

An *SMS package* is a bundle of information that you send from the SMS server to an SMS client. The package consists of a set of commands that can be run on the client workstation. These commands could be for system maintenance, changing client configuration parameters, or installing software.

To create an SMS package, perform the following steps:

- Step 1. From the **Package Properties** window, click on the **Workstations** push button. The **Setup Package For Workstations** window opens, with the imported response file and instance profile ready to use.
- Step 2. In the **Source Directory** field, enter the name of the parent directory where you put the copied DB2 files. For example, x:\db2prods, where x: represents your CD-ROM drive.
- Step 3. Select the name of the product to install from the **Workstation Command Lines** window.
- Step 4. If you changed and renamed the sample response file, click on the **Properties** push button. The **Command Line Properties** window opens. Change the value of the **Command Line** parameter to match the new response file name and path. If you are using a response file that was created using the response file generator, ensure that all the instance profiles are located in the same drive and directory as the response file that you specify.
- Step 5. Click **OK**.
- Step 6. Click the **Close** push button.
- Step 7. Click **OK** to close the opened windows. The **Packages** window shows the name of the new SMS package.

Distribute the DB2 Installation Package from the SMS Server

Now that you have created the package, you have three options:

- You can distribute your SMS package and then log on locally on the client workstation to run the package. This option requires that the user account used to perform the installation belongs to the *local Administrators* group where the account is defined.
- You can distribute your SMS package and then log on remotely on the client workstation to run the package. This option requires that the user account used to perform the installation belongs to the *Domain Admins* group.
- You can set up your SMS package with an auto-install feature.

Options 1 and 2 are available to you, but for a large number of installations we recommend option 3, which will be our focus for this step.

Once sent to the client workstation, the SMS package will tell the client workstation what code to execute, and the location, on the SMS server, of that code.

To send the code to a client workstation, perform the following steps:

- Step 1. Open the **Sites** window.
- Step 2. Open the **Packages** window.
- Step 3. In the **Packages** window, select the appropriate package and drag it onto the target client in the **Sites** window. The **Job Details** window opens. This window lists the package that will be sent to the client machine (Machine Path) and the command that will be executed at the workstation.
- Step 4. Select the **Run Workstation Command** check box and select the installation package that you want to use.
- Step 5. In the **Run Phase** box of the **Job Details** window, select the **Mandatory After** check box. A default mandatory date is set one week from the current date. Adjust the date as required.
- Step 6. Deselect the **Not Mandatory over Slow Link** check box.



This feature is critical if you are installing across a large number of workstations. It is recommended that you stagger the installation to avoid overloading your server. For example, if you are considering an overnight install, then spread out the install time for a manageable amount of client workstation.

For more information about completing the **Job Details** window, refer to *Microsoft's Systems Management Server Administrator's Guide* for your platform.

- Step 7. When the job specifications are complete, click **OK**. You are returned to the **Job Properties** window.
- Step 8. Add a comment that explains what the job will do. For example, `Install DB2 Run-Time Client`.
- Step 9. Click the **Schedule** push button and the **Job Schedule** window opens. This window will arrange a priority for this job. By default, the job is low priority and all other jobs will be executed first. It is recommended that you select medium or high priority. You can also select a time to start the job.
- Step 10. Click **OK** to close the **Job Schedule** window.
- Step 11. Click **OK**.

The job is created and the package is sent to the SMS client workstation.

To run the installation on the SMS client, perform the following steps:

- Step 1. On the target SMS client workstation, log on to the workstation with a user account that belongs to the *local Administrators* group where the account is defined. This level of authority is required because a system program install is being performed instead of a user program install.
- Step 2. Click **Start** and select **Programs—>SMS Client—>Package Command Manager**. The **Package Command Manager** window opens.
- Step 3. When the SMS client workstation receives the packages from the SMS server, it is listed in the **Package Name** section of the window. Select the package and click on the **Execute** push button. The installation runs automatically.
- Step 4. Following installation, you must reboot the SMS client workstation before using DB2.

Note: If you specified `REBOOT = YES` in your response file, the SMS client will reboot automatically.
- Step 5. Click **Start** and select **Programs—>SMS Client—>Package Command Manager**. The **Package Command Manager** window opens.
- Step 6. Click the **Executed Commands** folder and verify the execution of the package. Similarly, you can verify completion on the SMS server by checking the status of the job and ensuring that it has been changed to complete from pending or active.

On the SMS client, open the Package Command Manager again. When the package, which you created and sent to the client, appears under the Executed Commands folder, the installation has completed.

Configuring Client Settings

Configuring Remote Access to a Server Database

Once you have installed your DB2 product, you can configure your product to access remote databases individually on each client workstation using the Client Configuration Assistant (CCA) or the Command Line Processor (CLP). DB2 uses the **CATALOG** command to catalogue remote database access information:

- The **CATALOG NODE** command specifies the protocol information on how to connect to the host or server.
- The **CATALOG DATABASE** command catalogs the remote database name and assigns it a local alias.
- The **CATALOG DCS** command specifies that the remote database is a DRDA database. (This command is only required for DB2 Connect Personal Edition and DB2 clients.)
- The **CATALOG ODBC DATA SOURCE** command registers the DB2 database with the ODBC driver manager as a data source.

For more information on cataloging remote databases, refer to the *Administration Guide*.

If you plan to roll out multiple copies of DB2 clients with identical configurations, then you can create a batch file that will run your customized script.

For example, consider the following sample batch file, `myscript.bat`, used to run the script file:

```
@echo off
cls
db2cmd catmvs.bat
```

The `DB2CMD` command initializes the DB2 environment and the `catmvs.bat` file calls the batch job of the same name.

Here is a sample catalog script file, `catmvs.bat`, that could be used to add databases to a DB2 Connect Personal Edition workstation:

```
db2 catalog tcpip node tcptst1 remote mvshost server 446
db2 catalog database mvbdb at node tcptst1 authentication dcs
db2 catalog dcs database mvbdb as mvb_locator
db2 catalog system odbc data source mvbdb
db2 terminate
exit
```

You can either send these files to your client workstations manually or use SMS and have the script execute automatically after the installation and reboot have completed. To create another SMS package with the catalog script, perform the following steps:

- Step 1. Click **Start** and select **Programs—>Systems Management Server—>SMS Administrator** . The **Open SMS Window** window opens.
- Step 2. Select the **Packages** window type and click **OK**. The **Packages** window opens.
- Step 3. Select **File—>New** from the menu bar. The **Package Properties** window opens.
- Step 4. Enter a name for your new package. For example, batchpack.
- Step 5. Enter a comment about the package. For example, Package for batch file.
- Step 6. Click on the **Workstations** push button. The **Setup Package for Workstations** window opens.
- Step 7. Enter the source directory. Ensure that the source directory is a location that both the server and the client have access to, and that contains the batch file that is to be run from the client workstation.
- Step 8. Under the **Workstation Command Lines** section, click on **New**. The **Command Line Properties** window opens.
- Step 9. Enter a command name.
- Step 10. Enter the command line.
- Step 11. Click the check box for the platforms that should be supported, under the **Supported Platforms** section.
- Step 12. Click **OK**.
- Step 13. Click **Close**.
- Step 14. Click **OK**.

Distribute this package in the same way as you did in “Distribute the DB2 Installation Package from the SMS Server” on page 394.

Configuring db2cli.ini

The `db2cli.ini` file is an ASCII file which initializes the DB2 CLI configuration. This file is shipped to help you get started and can be found in the `x:\sqllib` directory, where `x`: represents the drive where you installed the DB2 files.

If you need to use any specific CLI optimization values or CLI parameters, you can use your customized `db2cli.ini` file for your DB2 client workstations. To do so, distribute your `db2cli.ini` file over to each DB2 client workstations and put it into their `\sqllib` directory over.

Exporting and Importing a Profile

If you wanted to use an instance profile, and did not use one when you installed your DB2 product using the response file that was created by the response file generator, you can enter the **db2cfexp** command to create an instance profile and the **db2cfimp** command to import an instance profile. For more information, refer to the *Command Reference*.



You can also use the CCA to export and import an instance profile.

Chapter 21. Distributed DB2 Installation on UNIX Operating Systems

Before You Begin

Before you begin any installation please read through this entire section. There are configuration and setup details that should be considered before starting any installation.

Limitations on Response File Install

You should be aware of the following limitations when using the response files method to install DB2 on UNIX platforms:

- If you set any instance or global profile registry keywords to BLANK (the word "BLANK"), the effect is to delete that keyword from the list of currently set keywords. If the registry variable corresponding to a keyword is not already set, and you run a response file install with this keyword set to BLANK, then you will receive an error.
- If you are using the response file to install on Linux, please ensure that you have sufficient space prior to installing. Otherwise you may need to do some manual cleanup if the installation fails.
- You can use the response file to install additional components/products after an initial install. However, you should not comment out the PROD and COMP keywords, or you may have some missing components even with a successful response file install.

Step 1. Mount the CD-ROM

For information on how to mount the CD-ROM, refer to the appropriate *Quick Beginnings* book.

Note: If you are going to use the response file feature, then you must set up all the users/groups before running the response file installation.



We recommend installing from a file system network hard drive rather than a CD-ROM, especially if you use the CD-ROM drive for other tasks. Installing from a mounted CD-ROM drive will significantly increase the amount of time it will take to perform the install. If you are planning on installing multiple clients, you should set up a mounted file system on a code server to improve performance.

Step 2. Create a Response File

The DB2 CD-ROM includes a ready-to-use sample response file with default entries. The sample response files are located in

`<cd-rom>/db2/install/samples`

where <cd-rom> represents the location of the installable version of DB2.

Response files are available for each DB2 product. For more information see “Available Sample Response Files” on page 376.

To create a customized response file from the sample, perform the following steps:

- Step 1. Copy the sample response file to a local file system and edit it.
- Step 2. To activate an item in the response file, remove the asterisk (*) to the left of the keyword. Then, replace the current setting to the right of the value with the new setting. The possible settings are listed to the right of the equal sign.

Keywords that are unique to installation are only specified in a response file during an distributed installation. For a list of installation keywords, see “Important Response File Keywords” on page 377.
- Step 3. Save the file on an exported file system available to everyone on the network.

If you are installing directly from the CD-ROM, you must store the renamed response file on another drive.

Note: You can specify the name of the instance owner in the response file. If this user does not already exist, DB2 will create this user on your system. The Admin instance can be created in a similar manner. If NIS/NIS+ is involved, the users/groups will have to be created first.

Step 3. Start an Unattended Installation with a Response File

To perform an unattended installation, perform the following steps:

Step 1. Log in as a user with root authority.

Step 2. Enter the **db2setup** command as follows:

```
<cd-rom> /db2setup -r <reponsefile_directory>/<reponse_file>
```

where <cd-rom> represents the location of the DB2 installable image; <responsefile_directory> represents the directory where the customized response file is located, and <response_file> represents the name of the response file.

Step 3. Check the messages in the log file when the installation finishes. The location of the log file is: /tmp/db2setup.log

Chapter 22. Distributed DB2 Installation on OS/2 Operating Systems

You can use this feature to install DB2 products on OS/2 operating systems:

- Locally, from a CD-ROM (see “Installing DB2 Products from a Hard Disk or CD-ROM”).
- Remotely, across a network connection from a hard disk or CD-ROM (see “Installing DB2 Products from a Hard Disk or CD-ROM”).



If you need to install the same DB2 product on several machines, you can use the unattended installation's "response file" feature: create a customized configuration; generate the customized response file; then reuse it as a template for each installation.

Installing DB2 Products from a Hard Disk or CD-ROM

To install a DB2 product on local or remote workstations from a hard disk or CD-ROM, perform the following steps:

Step 1. “Make DB2 Files Available for Installation”

Step 2. “Create a Response File for Distributed Installation” on page 404

Step 3. “Run the CMD File from the Remote Workstation” on page 405

Before You Begin

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

Make DB2 Files Available for Installation

To make the DB2 install files accessible to the machine where DB2 will be installed:

- If performing remote installs directly from the CD-ROM:
 1. Insert the appropriate CD-ROM into the drive.
 2. Ensure that the CD-ROM will remain in the drive for as long as it is needed for the install. If the drive is often used for other tasks as well, consider installing from the hard disk instead.
- If installing from a hard disk, you must copy the required files from the CD-ROM directories to the hard disk. At the command prompt, use the **xcopy** command with the **/s** option.
 - The syntax for the two commands required is:

```
xcopy x:\db2\os2\language e:\clients\os2\language /s
xcopy x:\db2\os2\common e:\clients\os2\common /s
```

where:

- *x*: represents the CD-ROM drive
- *language* represents the two-character code that represents your language (for example, en for English). Table 39 on page 506 lists the code for each available language.
- *e*: represents the destination drive

If performing a remote installation, you must enable your network clients to access the DB2 install files.

Grant *shared access* authority to the directory that you just created, or to the CD-ROM drive if installing directly from CD-ROM.

Create a Response File for Distributed Installation

During an distributed installation, you supply the setup and configuration data in a response file that you create before running the installation. DB2's install package includes a ready-to-use sample response file, with default entries already in place.



If you intend to use the sample response file provided, without making any changes to its values, you can skip this step and go to "Run the CMD File from the Remote Workstation" on page 405.

Edit the sample response file:

1. Find the correct sample response files for the product you want to install. Response files available for each DB2 product, see "Available Sample Response Files" on page 376. Copy the files to a local directory.
2. To activate an item in the response file, remove the asterisk (*) to the left of the keyword/environment variable, erase the current setting to the right of the value and type in a new setting. The possible settings are listed to the right of the current setting.

An example section of a response file appears below:

```
⋮
FILE                = c:\sql11b

COMP                = DB2 Run-Time Client
⋮
*DB2ACCOUNT        = BLANK or char(199)

*DB2BQTIME         = BLANK or 1 - MAX
⋮
```

Keywords that are unique to installation are only specified in a response file during an distributed installation. For a list of installation keywords, see “Important Response File Keywords” on page 377.

3. Exit the file. If you have made any changes, save the file under a new file name to preserve the original sample response file.

If you are installing directly from the CD-ROM, you must store the renamed response file on a local drive.

Run the CMD File from the Remote Workstation

A command (CMD) file contains the commands which will start the installation program. You must edit this file before running the installation.

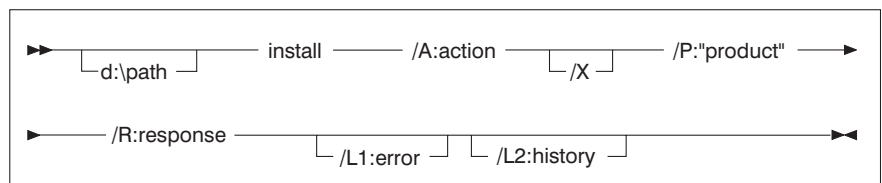
1. Find and open the correct CMD file.

If installing directly from the CD-ROM, substitute x:\os2 for e:\clients\os2\ in the directory names below.

- The CMD files are:
 - db2admc1.cmd for a DB2 Administration client.
 - db2conee.cmd for DB2 Connect Enterprise Edition.
 - db2conpe.cmd for DB2 Connect Personal Edition.
 - db2rtc1.cmd for a DB2 Run-Time client.
 - db2sdk.cmd for DB2 Application Development Client.
 - db2udbee.cmd for DB2 Universal Database Enterprise - Extended Edition.
 - db2udbwe.cmd for DB2 Universal Database Workgroup Edition.
 - db2udbpe.cmd for DB2 Universal Database Personal Edition.
 - db2udbwm.cmd for DB2 Data Warehouse Center.

2. Edit the command in the file, to fill in the information needed for the installation.

- You must specify the complete install command. A complete command requires the following syntax:



The following is an example of a complete command in the sample command files:

```
e:\clients\os2\language\install\install /X
/P:"IBM DB2 Run-Time Client"
/R:e:\clients\os2\language\db2rtc1.rsp /L1:d:\error.log
/L2:d:\history.log
```

where:

d:\path

Specifies the location of the install files. If installing from the hard drive, specify the directory created in Step 1.

- /A** (Optional) Specifies the action to be performed. Only required when deleting a DB2 product (/A:D).
- /X** Specifies that the installation will run in unattended mode.
- /P** (Optional) Specifies the name of the product you want to install. The name must be exactly the same as the translated product name that the installer uses. Only required if there is more than one product on this CD-ROM.
- /R** Specifies the fully qualified response file name. If you changed and renamed the sample response file provided, make sure this parameter matches the new name.
- /L1** (Optional) Specifies the fully qualified log file name, where setup information and any errors occurring during setup are logged. Defaults to x:/DB2LOG/L1.LOG, where x: represents your boot drive.
- /L2** (Optional) Specifies the fully qualified history log name, where all the files processed by the installation program are listed. Defaults to x:/DB2LOG/L2.LOG, where x: represents your boot drive.

3. Save and exit the CMD file.

If you are installing directly from the CD-ROM, you must rename the CMD file, store it on a local drive, and use that file name in the next step.

- 4. Start the installation by entering the CMD file's name on the command line. The DB2 client is now ready to install on the target workstation.
- 5. When installation is complete, check the error and history logs for any errors or problems.



To configure your client to access remote servers, go to "Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 29.

Part 7. Thin Client and Thin Connect Architecture

Chapter 23. Installing and Configuring Thin-Workstations

You can install a DB2 Client or DB2 Connect Personal Edition on a workstation that is running a Windows 32-bit operating systems and have these workstations act as code servers to DB2 Thin-Client or DB2 Thin-Connect workstations in your enterprise. This section describes the steps and considerations involved in setting up this environment.

In this section, the term Thin-workstation can refer to either a DB2 Thin-Client workstation or a DB2 Thin-Connect workstation. A Thin-workstation can only load code from a code server that is running the same product.

These Thin-workstations load the DB2 Client or DB2 Connect Personal Edition code across a LAN connection from these code servers. A Thin-workstation functions like any other DB2 Client or DB2 Connect Personal Edition workstation; this type of architecture is transparent to the user. The main difference is that the code is installed on a code server, and not individually on each workstation. Each Thin-workstation needs only a minimal amount of code and configuration to establish links to a code server. This in contrast to a locally installed DB2 Client or DB2 Connect Personal Edition workstation, where all the code is stored and run locally. This configuration should not be confused with a Citrix environment. In a Citrix environment, both the code and the processing is handled by the Citrix server. In a Thin environment no processing is done at the code server in this environment.

A typical DB2 Thin-Client environment is shown in Figure 3 on page 410. A DB2 Administration Client is installed on a machine with the Thin-Client Code Server component. After some configuration, this machine will be known as a DB2 Thin-Client Code Server. A DB2 Administration Client is the only type of client that can act as a code server for Thin-Client workstation. The DB2 Thin-Client workstations access the code server to dynamically load any code required. Once the code is loaded, all processing is done locally on the DB2 Thin-Client workstations. Using local database configuration information, a connection is made to a target DB2 server and the data is retrieved.

The DB2 code is actually run on the Thin-Client workstations, the code is only loaded from the DB2 Thin-Client Code Server. There is no DB2 code installed on the Thin-Client workstations.

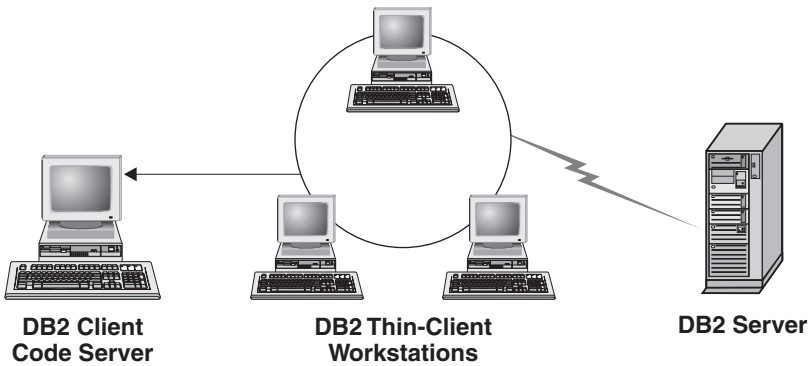


Figure 3. A typical DB2 Thin-Client setup

A typical DB2 Thin-Connect environment is shown in Figure 4. DB2 Connect Personal Edition is installed on a machine with the Thin Connect Code Server component. After some configuration, this machine will be known as a DB2 Thin-Connect Code Server. A DB2 Connect Personal Edition workstation is the only type of workstation that can act as a code server for DB2 Thin-Connect workstations. The DB2 Thin-Connect workstations function just like the DB2 Thin-Client workstations. They dynamically load any code required from the DB2 Thin-Connect Code Server. Once the code is loaded, all processing is done locally on the DB2 Thin-Connect workstations. Using local database configuration information, a connection is made to a target host or AS/400 DB2 server and the data is retrieved.

For example, the DB2 Connect code is run on the Thin-Connect workstations. This environment could also be extended to access a database on a DB2 server that is not located on a host or AS/400 system using the built-in client component with DB2 Connect Personal Edition.

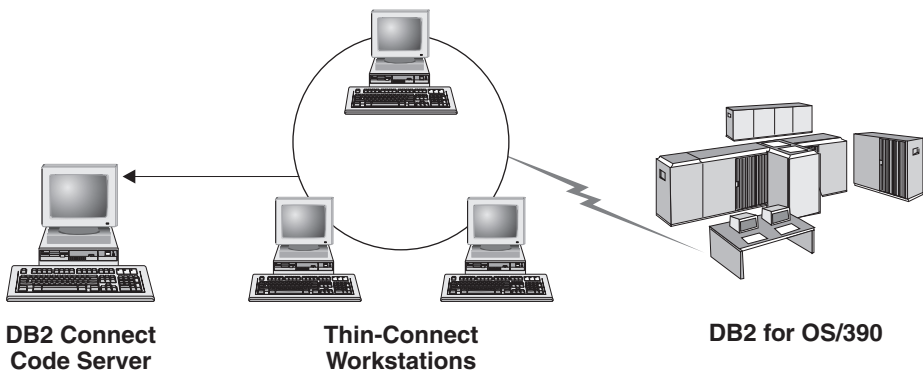


Figure 4. A typical DB2 Thin-Connect setup

The Benefits and Disadvantages in a Thin Environment

As with most scenarios, there are advantages and disadvantages associated with a Thin-workstation in your environment. You have to go over these issues and decide whether or not this type of configuration suits your business needs. For the most part, you will find this method of supporting a DB2 Client or DB2 Connect Personal Edition well suited for most business models.

The benefits of a Thin-environment are:

1. Reduced disk space required for each Thin-workstation; you can save from 16–112 MB per workstation.
2. Easier software changes and updates; you only have one copy of the DB2 product to change and update.
3. Centralized maintenance of code; you need to install, update or migrate the code on only one machine.
4. FixPack rollout and rollbacks are easier to administer and accomplish.
5. Installation can follow a PUSH or PULL methodology; you can integrate this type of deployment with a system management tool and PUSH the installation out to its target workstations.

The drawback of a Thin-workstation environment is an initial potential loss of system performance, since at program invocation time, DB2 programs are loaded from a code server instead of the client machine. However, any loss will depend on network load, network speed, and code server speed and load.

Once this code has been loaded by the Thin-workstations, the fact that these are Thin-workstations becomes transparent to the user. Thin-workstations have to dynamically load libraries as needed, so if you start another DB2 application, you may experience a minor performance degradation period while this new application is being loaded.

For example, when you connect to a database, you will notice a short performance loss when compared to a client where the code is locally installed. After this code was loaded, your client would have the same performance as the locally installed client. If after you have connected to the database you decided to start the Client Configuration Assistant (CCA), you would then experience a performance loss as the code associated with this connectivity tool was loaded from the code server. In reality, you likely wouldn't be loading the CCA and other components very often, you will mostly be using the run-time environment.

Another potential disadvantage is the location of the catalog files. The catalog files contain all the information needed for a workstation to connect to a

database. Unless you are using Lightweight Directory Access Protocol (LDAP) in your environment, you have to maintain catalog information on each and every workstation, just like a regular installation.

If you are not using LDAP, there are some attractive solutions to this problem. You can get around cataloging databases on each and every Thin-workstation by using the profile export and import options provided by the CCA. For example, a e-mail message could easily be sent to each workstation that would update each machine with the correct catalog information.

Performing the Installation of a DB2 Thin-Client or DB2 Thin-Connect Environment

Performing a Thin-installation is a straightforward process. To set up this type of environment, you need to perform the following steps:

- “Step 1. Install a DB2 Administration Client or DB2 Connect Personal Edition With The Code Server Component”
- “Step 2. Setup Cross-Platform Support on the Code Server (Optional)” on page 413
- “Step 3. Share the Code Server Directory where the DB2 Client or DB2 Connect Personal Edition Code is Installed” on page 414
- “Step 4. Create a Response File for the Target Thin-Workstation.” on page 415
- “Step 5. Make the Code Server Accessible to the Target Thin-Workstation.” on page 417
- “Step 6. Create the Target Thin-Workstation” on page 418

Step 1. Install a DB2 Administration Client or DB2 Connect Personal Edition With The Code Server Component

You need to install a DB2 Administration Client or DB2 Connect Personal Edition on the workstation that will act as the code server for the Thin-Client or Thin-Connect target workstations. A DB2 Thin-Client can only load code from a DB2 Thin-Client Code Server and a DB2 Thin-Connect workstation can only load code from a DB2 Thin-Connect Code Server.

Note: When you are installing your DB2 Client, make sure you perform a Custom installation. From the Select Components window you can select the components you want to install. You must select the **Thin Client Code Server** to install the files needed to setup the thin client.

To install a DB2 Administration Client, see “Chapter 2. Installing DB2 Clients” on page 11. To install DB2 Connect Personal Edition, refer to your *Quick Beginnings* book.

Step 2. Setup Cross-Platform Support on the Code Server (Optional)

This step describes how to setup a Windows NT and Windows 2000 code server (referred throughout this section as Windows code server) to service Windows 9x Thin-Client workstations.



If you are not planning on supporting a mix of Windows NT, Windows 2000, and Windows 9x Thin-workstations in your environment, you can skip this step.

A code server can only provide code for Thin-workstations that belong to the same family of operating systems. This means that a Windows NT or Windows 2000 code server could not support a Windows 9x-based Thin-workstation and vice-versa.

If you have deployed a heterogeneous configuration of Windows 32-bit operating systems in your enterprise, you can set up your code server to support both Windows NT or Windows 2000-based and Windows 9x-based Thin-workstations by performing the following steps:

Note: The example that follows will assume that you are configuring a Windows NT-based code server to service Windows 9x Thin-workstations.

To have a Windows code server service a Thin-workstation on a Windows 9x machine, perform the following steps:

1. Create a directory on the Windows NT or Windows 2000 code server that will be used to service Windows 9x Thin-workstations by entering the **md d:\sqllib9x** command, where *d:* represents a local hard drive.
2. Copy the DB2 product directory on the code server (for example, *c:\sqllib*) into the directory that you just created by entering the following command:

```
xcopy c:\sqllib\*. * d:\sqllib9x /s /e
```

where:

- *c:* represents the drive on the code server where the DB2 product was installed.
 - *d:* represents the drive on the code server where the *sqllib9x* directory was created in the previous step.
3. Change focus to the directory that you created for the cross platform code server. For example, you would enter the **cd d:\sqllib9x** command. This directory would be used to serve Thin-workstations running on Windows 9x.
 4. Enter the **d:\sqllib9x\bin\db2thn9x.bat target_platform** command to enable this machine to service a cross platform Thin-workstation

where:

- *d:* is the local drive that you created to act as a code server for cross platform Thin-workstations.
- *target_platform* is the platform that this directory will support. This value can take one of two settings, *nt* or *9x*. If the code server was running Windows NT or Windows 2000 and you wanted it to service Thin-workstations running Windows 9x, you would use the *9x* parameter. If the code server was running Windows 9x and you wanted it to service Thin-workstations running Windows NT or Windows 2000, you would use the *nt* parameter.

You now have two code bases on your code server. If you are going to create a Thin-workstation on a Windows NT or Windows 2000 machine, use the Windows NT or Windows 2000 code base (for example, *c:\sql11ib*) in the remaining steps. If you are going to create a Thin-workstation on a Windows 9x machine and have the code served by a Windows NT or Windows 2000 code server, use the Windows 9x code base (for example, *d:\sql11ib9x*) in the remaining steps.

Note: If you are logged onto a Windows 9x Thin Client workstation that is running code off of a Windows NT or Windows 2000 code server. You must ensure that the user account that you are logged on to the Windows 9x workstation is locally defined on the Windows NT or Windows 2000 code server.

Step 3. Share the Code Server Directory where the DB2 Client or DB2 Connect Personal Edition Code is Installed

In order for the Thin-workstations to load the required code from the code server, each of the target Thin-workstations must be able to read the directory where the DB2 Client or DB2 Connect Personal Edition code is installed. To make the code directory available to all Thin-workstations in READ mode, perform the following steps:

1. Click on **Start** and select **Programs** → **Windows Explorer**.
2. Select the directory where you installed the DB2 product. Use the *c:\sql11ib* directory for Thin-workstations running on Windows NT or Windows 2000. If you were going to set up Thin-workstations that were running on Windows 9x, you would also have to share the *d:\sql11ib9x* directory.
3. Select **File** → **Properties** from the menu bar.
4. Select the **Sharing** tab.
5. Select the **Shared As** radio button.
6. In the **Share Name** field, enter a share name. For example, enter **NTCODESERVER**.

7. All target Thin-workstations need to have READ access to this directory for all users. If you are setting up a Windows NT or Windows 2000-based code server, specify READ access for everyone as follows:
 - a. Click the **Permissions** button. The Access Through Share Permissions window opens.
 - b. In the **Name** box, select **Everyone**.
 - c. Click the **Type of Access** drop down box and select **Read**.
 - d. Click **OK** until all windows are closed.

Note: If you are setting up a Windows 9x-based code server, you do not need to specify this type of access when you set up a share. By default, everyone is granted read access.

Step 4. Create a Response File for the Target Thin-Workstation.

When you installed the code server, you performed an interactive installation. In this installation, you have to manually respond to prompts from the setup program in order to install your product. Your responses provided the information needed to install the DB2 product and configure its environment. During a distributed installation, this information is provided in the form of keywords and values in a response file. For this reason, a distributed installation is often referred to as a unattended installation or even a silent installation. For a detailed description of a distributed installation and how to perform this installation for a target Thin-workstation, see “Chapter 19. An Introduction to Distributed Installation” on page 375.

For a DB2 Thin-Client or DB2 Thin-Connect installation, there is a ready-to-use response file called `db2thin.rsp`, which you can use to install either Thin-workstation. The default settings for the most common installation type are provided in this file. You can find this response file in the `c:\sql11b\thinsetup` directory, where `c:` represents the drive where you installed your DB2 product.

A response file contains:

- Keywords unique to installation
- Registry value/environment variable settings
- Database manager configuration parameter settings

In a response file, the asterisk (*) acts like a comment. Any line that is prefixed with an asterisk will be ignored during the installation. If you do not specify a keyword, or it is commented out, a default value will be used. To enable a parameter, remove the asterisk.

For example, let’s assume that you want to install support for ODBC. The default entry for this keyword in the response file is:

```
*COMP =ODBC_SUPPORT
```

To install this component, you would remove the asterisk from this line so it would look like:

```
COMP =ODBC_SUPPORT
```

For some keywords, you have to set values. To enable these keywords, remove the asterisk as usual, but make sure that you also replace the contents to the right of the equals sign with the value you want for that parameter.

For example,

```
*DB2.DIAGLEVEL = 0 - 4
```

would be:

```
DB2.DIAGLEVEL = 4
```

if you wanted to set this parameter to 4.

The following is a section of the db2thin.rsp sample response file:

```
* Required Global DB2 Registry Variable
* -----
DB2INSTPROF                = C:\CFG

* General Options
* -----
*TYPE                      = 0,1,2 (0=compact, 1=typical, 2=custom)
*COMP                      = ODBC_SUPPORT
*COMP                      = CONTROL_CENTER
*COMP                      = EVENT_ANALYZER
*COMP                      = WEB_ADMINISTRATION
*COMP                      = QUERYMONITOR
*COMP                      = TRACKER
*COMP                      = QUERYADMIN
*COMP                      = CLIENT_CONFIGURATION_ASSISTANT
*COMP                      = COMMAND_CENTER
*COMP                      = DOCUMENTATION
*CREATE_ICONS              = YES or NO (default=YES)
*REBOOT                    = YES or NO
```

For more information, see “Available Sample Response Files” on page 376 for detailed descriptions about these parameters.

For example, set the following:

```
DB2INSTPROF                = C:\CFG
TYPE                      = 1
REBOOT                    = YES
```

Once you have finished editing this file, save it as a different name so you can maintain the integrity of the sample. Call this file `test.rsp` and save it in the same directory that you shared in the previous step (`c:\sql11b`).

Step 5. Make the Code Server Accessible to the Target Thin-Workstation.

The code server must be accessible before you can setup the target workstation to act as a Thin-workstation. You may choose to map a network drive to access the `thnsetup` directory which is under the shared directory that you created on the code server.

1. Click **Start** and select **Programs** → **Windows Explorer**.
2. From the **Tools** menu, select **Map Network Drive**.
3. In the **Drive** drop down list, select the drive you want to map the location of the code server to.
4. In the Path, type the location of the share in the format

`\\computer_name\share_name`

where:

computer_name

represents the computer name of the code server.

share_name

represents the share name of the shared directory on the code server. In our example, this was `NTCODESERVER`.

5. Check the **Reconnect at Logon** box to make the share persistent.

If you are running Windows NT or Windows 2000, you may also specify user information in the Connect as Input field. The format is

`domain\username`

where

domain

represents the domain where the user account is defined. This is only required the account is a domain account and you are not logged on to the system with a user account that has READ access on the remote code server.

username

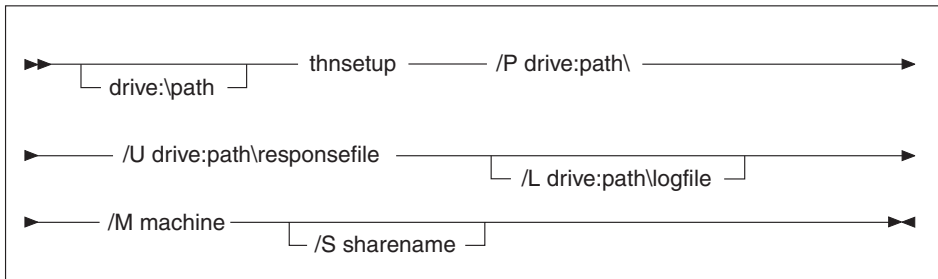
represents the user has access to this machine. This is only required if you are not logged on to the system with a user account that has READ access on the remote code server or you specified the domain parameter.

Note: On a Windows 9x workstation, you must be logged on to the workstation as a valid user with shared directory access to the code server.

If you are planning on having a DB2 for Windows NT or Windows 2000 code server service a Windows 9x Thin-workstation (or vice-versa), make sure that you enter the share name for the directory that you created for this purpose.

Step 6. Create the Target Thin-Workstation

To finish this setup, you need to run the **thnsetup** command. This command will set up the DB2 Thin-Client or DB2-Thin Connect workstation and the required links to the code server. The **thnsetup** command can be entered with the following parameters:



where:

/P Specifies the path where the DB2 code is installed on the code server. Specify this parameter using the path that you attached to in the previous step. Remember, if this is a Windows NT or Windows 2000 code server and your target workstations are running Windows 9x, you have to specify the directory you set up for Windows 9x. This parameter is required.

Note: If you have not already mapped a persistent network drive to the code server as described in “Step 5. Make the Code Server Accessible to the Target Thin-Workstation.” on page 417, then this parameter should be the drive letter which will be used to represent the network drive.

/U Specifies the fully qualified response file name. This parameter is required. For our example, use the response file that you saved as `test.rsp` on the remote code server.

/L Specifies the fully qualified log file name, where setup information and any errors occurring during setup are logged. If you do not specify the log file’s name, the default `db2.log` file name is used. This file will be created in a directory called `db2log`, on the drive where your operating system is installed. This parameter is optional.

/M Specifies the computer name of the code server. This parameter is only necessary if you did not map a persistent network drive as in “Step 5. Make the Code Server Accessible to the Target Thin-Workstation.” on page 417.

/S Specifies the share name of the code server where the DB2 product was installed. This parameter is only necessary if you did not map a persistent network drive as in “Step 5. Make the Code Server Accessible to the Target Thin-Workstation.” on page 417.

For example, to create a Thin-workstation where the shared *NTCODESERVER* directory on a code server called *myserver* is mapped locally to the *x:* drive, and the response file is called *test.rsp* and is located in the same directory as the code server, enter the following command:

```
x:\thnsetup\thnsetup /P x:\ /U x:\thnsetup\test.rsp
```

When the **thnsetup** command completes, check the messages in the log file. Since we set the machine to reboot once the setup of the Thin-workstation completes, if your machine did not reboot, you know you have encountered an error.

Your Next Step

Now that you have setup and configured your Thin-workstations, you need to set up access to the databases that you want your users to access on each workstation. We suggest you use the Client Configuration Assistant (CCA) to set up access to these databases on the code server. For more information on how to use the CCA, see “Starting the Client Configuration Assistant” on page 447.

After you have cataloged all of the databases on the code server that you want your Thin-workstations to access, you need to update all of the local catalog directories on each Thin-workstation. We recommend that you use the profile option of the CCA. The CCA allows you to export and import Client Profiles that contain database connection information, as well as configuration settings. On the code server, export a Client Profile. This profile will contain all of the information needed to set up the exact same database connections and configurations on each of the Thin-workstations that you just installed. Your end users can either use the CCA import option if you want to expose this tool. If you are trying to hide your DB2 product from your users, you can use the **db2cfimp** command. This command could be pulled down via an e-mail or pushed out with SMS. Using this command allows you to hide any DB2 tasks from your end users. For more information on the **db2cfimp** command, refer to the *Command Reference*.

Part 8. Configuring a Federated Database System

Chapter 24. Creating and Configuring a Federated Database System

A DB2 federated database system enables users and applications to reference multiple database management systems or databases within a single SQL statement. For example, with DB2's federated database support, you can join data that is located in a DB2 Universal Database table, a DB2 for OS/390 table, and an Oracle 7 table. Statements of this type are called distributed requests:

Location transparency

Separating the location of a data source from its network address.

Network address

Achieved by assigning nicknames to all tables and views that will be involved in distributed request queries.

Nickname

Consists of a user-defined name that is mapped to values that identify the physical address of a data source.

Both the nickname and these values are stored in the federated server's local database system catalogs.

Federated support for DB2 family data sources is included with Personal Edition, Workgroup Edition, Enterprise Edition, and Enterprise - Extended Editions of DB2.

Federated support for Oracle data sources requires DB2 Relational Connect.

For more information about federated database system concepts, refer to the *Administration Guide*.

Figure 5 on page 424 illustrates a typical federated system environment.

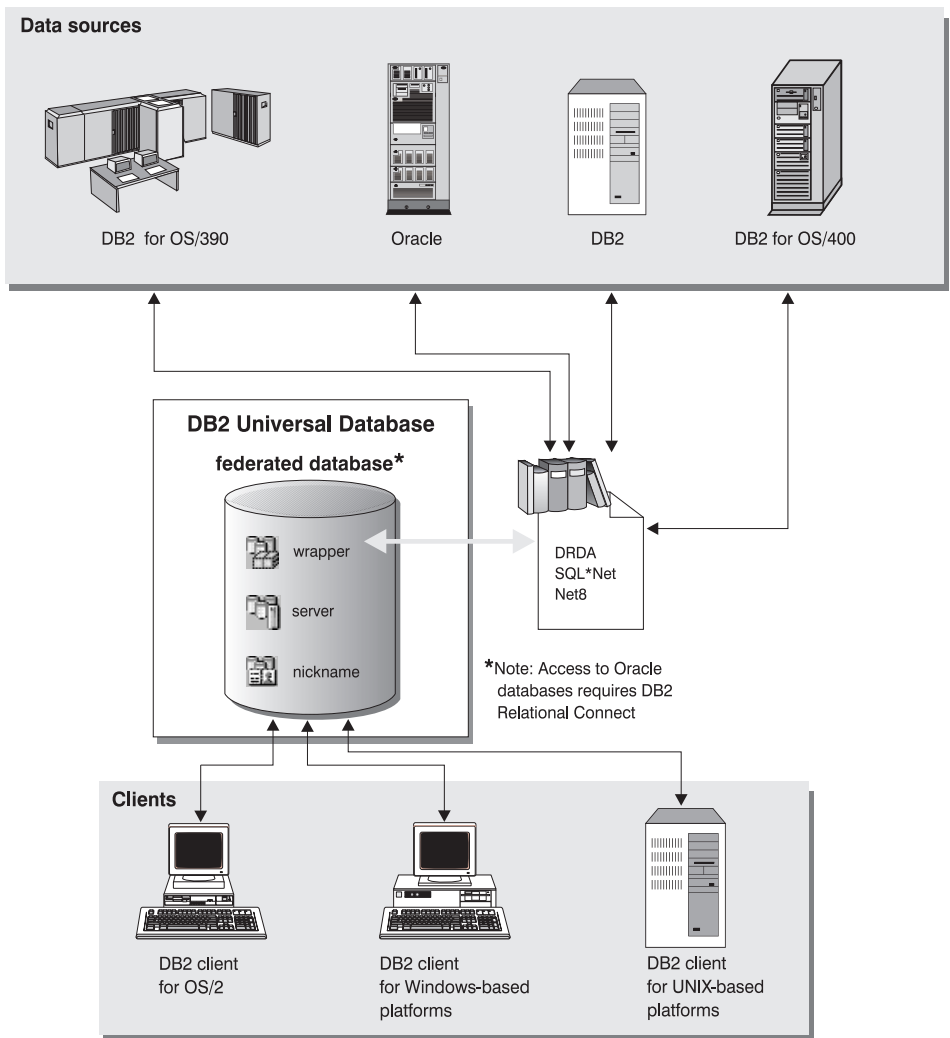


Figure 5. DB2 Federated System

Supported Data Sources

DB2 family and Oracle data sources can be included in distributed requests using nicknames. Table 35 on page 425 lists the different supported versions and any required maintenance, and corresponding access methods.

Table 35. Data Sources and Their Access Methods

Data Source	Access Method
<ul style="list-style-type: none">• DB2 Universal Database• DB2 for OS/390 Version 5 with PTF PQ07537• DB2 for MVS V2R3 with APAR PN43135, UN75958, UN54600, and UN56735• DB2 for MVS V3R1 with APAR PN70612, UN42626, UN54601, and UN73393• DB2 for MVS V4R1 with APAR PN70612• SQL/DS• DB2 for Common Servers V2• DB2 Parallel Edition• DataJoiner¹	DRDA
Oracle V7.0.16 or later	Oracle SQL*Net or Net8
Any OLE DB provider	OLE DB 2.0 or later

Note:

1. On Windows systems, DataJoiner cannot be installed on the same machine as DB2 Universal Database. On UNIX systems, DataJoiner can be on the same machine, but you must run it on its own instance. Do not use an existing V7.1 DB2 instance.

Chapter 25. Setting Up a Federated System to Access DB2 Family Data Sources

This chapter explains how to configure your federated server to access data stored in DB2 family databases and contains the following sections:

- “Enabling Federated Database Functionality”
- “Adding DB2 Family Data Sources to a Federated System” on page 428
- “Verifying Connections to DB2 Family Data Sources” on page 431

The instructions in this section apply to Windows NT, Windows 2000, and supported UNIX platforms; platform-specific differences are noted where they occur.

Enabling Federated Database Functionality

To take advantage of federated database functionality:

On UNIX systems

During the installation of DB2 Universal Database you must select the Distributed Join for DB2 Data Sources option and optionally create an instance to use with this option. If you choose to create an instance, the FEDERATED parameter will be set to YES by default. If you choose to create an instance at a later date, you must manually set the FEDERATED parameter to YES for that instance.

This is only necessary if you are creating an instance using **db2icrt**. If you go back to using **db2setup** to either create or set up an existing instance, then the FEDERATED parameter will be set to YES again.

On Windows systems

Federated database functionality is enabled by default as part of the DB2 installation.

Adding DB2 Family Data Sources to a Federated System

Step 1. Configure network communications.



Configuring your federated server to access DB2 family data sources is similar to configuring a client to communicate with a DB2 server. Configuration instructions are provided in:

- “Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant” on page 29
 - “Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor” on page 41
-

Step 2. Catalog an entry in the federated server’s node directory that points to the location of the DB2 data source. The federated server determines the access method to use based upon the type of node being cataloged and the type of DB2 family database being accessed.

- If you are using SNA as your communication protocol, issue the CATALOG APPC NODE command:

```
CATALOG APPC NODE DB2NODE REMOTE DB2CPIC SECURITY PROGRAM
```

where:

- DB2NODE is a name that you assign to the node that you are cataloging.
 - DB2CPIC is the symbolic destination name of the remote partner node.
 - PROGRAM specifies that both a user name and a password are to be included in the allocation request sent to the partner LU.
- If you are using TCP/IP as your communication protocol, issue the CATALOG TCP/IP NODE command:

```
CATALOG TCPIP NODE DB2NODE REMOTE SYSTEM42 SERVER DB2TCP42
```

where:

- DB2NODE is a name that you assign to the node that you are cataloging.
- SYSTEM42 is the host name of the system where the data source resides.
- DB2TCP42 is the primary port name that is defined by the data source for use by data source clients.

Refer to the *Command Reference* for more information about these commands.

Step 3. Use the CREATE WRAPPER statement to define the wrapper module that will be used to access DB2 data sources. Wrappers are the

mechanism that federated servers use to communicate with and retrieve data from data sources. The following example shows a CREATE WRAPPER statement:

```
CREATE WRAPPER DRDA
```

where DRDA is the default name of the wrapper module used to access DB2 family data sources.

You can substitute the default name with a name that you choose; however, if you do so, you also must include the LIBRARY parameter and the name of the wrapper library for your federated server platform. Refer to the *SQL Reference* for more information about wrapper libraries.

- Step 4. Optional: Set the DB2_DJ_COMM environment variable to include the wrapper library that corresponds to the wrapper module that you created in the previous step; for example:

```
db2set DB2_DJ_COMM = libdrda.a
```

The DB2_DJ_COMM environment variable controls whether a wrapper module is loaded when the federated server initializes, which can result in improved performance when the DB2 family data source is first accessed. Refer to the *SQL Reference* for more information about wrapper library names.

- Step 5. Use the CREATE SERVER statement to define each DB2 server to which communications are configured; for example:

```
CREATE SERVER DB2SERVER TYPE DB2/OS390 VERSION 6.1 WRAPPER DRDA  
OPTIONS (NODE 'db2node', DBNAME 'quarter4')
```

where:

- DB2SERVER is a name that you assign to the DB2 data source. This name must be unique.
- DB2/390 is the type of DB2 data source to which you are configuring access.
- 6.1 is the version of DB2 for OS/390 that you are accessing.
- DRDA is the wrapper name that you defined in the CREATE WRAPPER statement.
- db2node is the name of the node where DB2SERVER resides. Obtain the node value by issuing the **db2 list node directory** command at the DB2 data source. This value is case sensitive.
- quarter4 is the name of a database at DB2SERVER. This value is case sensitive.

Although the node and database values are specified as options, they are required for DB2 data sources. Refer to the *SQL Reference* for a comprehensive list of options.

- Step 6. If a user ID or password at the federated server is different from a user ID or password at a DB2 family data source, use the CREATE USER MAPPING statement to map the local user ID to the user ID and password defined at the DB2 family data source; for example:

```
CREATE USER MAPPING FOR DB2USER SERVER DB2SERVER
OPTIONS ( REMOTE_AUTHID 'db2admin', REMOTE_PASSWORD 'day11te')
```

where:

- DB2USER is the local user ID that you are mapping to a user ID defined at a DB2 family data source.
- DB2SERVER is the name of the DB2 family data source that you defined in the CREATE SERVER statement.
- db2admin is the user ID at the DB2 family data source to which you are mapping DB2USER. This value is case sensitive.
- day11te is the password associated with db2admin. This value is case sensitive.

- Step 7. Use the CREATE NICKNAME statement to assign a nickname to a view or table located at your DB2 family data source. You will use this nickname when you query the DB2 family data source. The following example shows a CREATE NICKNAME statement:

```
CREATE NICKNAME DB2SALES FOR DB2SERVER.SALESDATA.MIDWEST
```

where:

- DB2SALES is a unique nickname for the DB2 table or view.
- DB2SERVER.SALESDATA.MIDWEST is a three-part identifier that follows this format:

data_source_name.remote_schema_name.remote_table_name

Refer to the *SQL Reference* for more information about the CREATE NICKNAME statement.

Refer to the *Administration Guide* for more information about nicknames in general.

- Step 8. Repeat the previous step for all database objects that you want to create nicknames for.

Verifying Connections to DB2 Family Data Sources

This section explains how to verify that you have correctly configured your federated system to access DB2 family data sources. You must have completed all the steps in “Adding DB2 Family Data Sources to a Federated System” on page 428 before continuing with this section.

- Step 1. Log on to the system with a user ID that has SYSADM or SYSCTRL authority.
- Step 2. If you have not already done so, create the sample database on your DB2 federated server by issuing the following command:

```
DB2SAMPL
```

- Step 3. Connect to the sample database on your DB2 federated server:

```
CONNECT TO SAMPLE
```

- Step 4. Issue the following SQL statement:

```
SELECT * FROM SYSCAT.SYSTABLES
```

You should receive the entire contents of the SYSCAT.TABLES system catalog table.

- Step 5. If you have not already done so, create the sample database at your DB2 family data source.
- Step 6. Add a system catalog table from the DB2 family data source’s sample database to your federated environment:
- When issuing the CREATE SERVER statement for the data source, specify SAMPLE for the DBNAME.
 - When issuing the CREATE USER MAPPING statement, make sure the user ID at the data source holds at least SELECT permission for the sample database.
 - Create a nickname for the sample database’s SYSCAT.COLUMNS system catalog table.
- Step 7. Using the nickname for the SYSCAT.COLUMNS table, issue the following SQL SELECT statement to retrieve data from the DB2 family data source; for example:

```
SELECT * FROM nickname
```

where *nickname* is the nickname for DB2 family sample database’s SYSCAT.COLUMNS table.

You should receive the entire contents of SYSCAT.COLUMNS.

After you successfully select data from both the federated database and a DB2 family data source, you can try joining data from the two data sources to complete the verification procedure.



If you will be including Oracle data sources in distributed request queries, proceed to “Chapter 26. Setting Up a Federated System to Access Oracle Data Sources” on page 433.

Chapter 26. Setting Up a Federated System to Access Oracle Data Sources

Accessing Oracle data sources requires DB2 Relational Connect installed on your DB2 federated server.

Two different Oracle wrapper modules are included with DB2 Relational Connect: one for use with the SQL*Net V1 or V2 client software, and one for use with the Net8 client software. The client software that you use dictates the wrapper module that you will use. If you use SQL*Net, you must use the sqlnet wrapper, and if you use Net8, you must use the net8 wrapper.

Regardless of the client software that you use, you can access both Oracle Version 7 and Oracle Version 8 data sources.

This chapter explains how to configure your federated server to access data stored on Oracle data sources through the use of nicknames and contains the following sections:

- “Installing DB2 Relational Connect”
- “Adding Oracle Data Sources to a Federated System” on page 435
- “Oracle Code Page Options” on page 441
- “Verifying Connections to Oracle Data Sources” on page 441

The instructions in this chapter apply to Windows NT, Windows 2000, and UNIX platforms; platform-specific differences are noted where they occur.

Installing DB2 Relational Connect

This section provides instructions for installing DB2 Relational Connect on Windows systems and on AIX systems.

Installing DB2 Relational Connect on Windows Systems

Before you install DB2 Relational Connect on your Windows system, make sure you have installed either DB2 Universal Database Enterprise Edition or DB2 Universal Database Enterprise - Extended Edition.

The following instructions explain how to install DB2 Relational Connect on your Windows NT or Windows 2000 system.

1. Log on to the system with the user account that you created to perform the installation.

2. Shut down any other programs so that the setup program can update files as required.
3. Insert the DB2 Relational Connect CD into the drive. The auto-run feature automatically starts the setup program. The setup program will determine the system language and launch the setup program for that language. If you want to run the setup program in a different language, or the setup program failed to auto-start, refer to the following tip.



To manually invoke the setup program, perform the following steps:

- a. Click Start and select the Run option.
- b. In the Open field, enter the following command:

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

where:

- x: represents your CD-ROM drive.
- language represents the country code for your language (for example, EN for English).

- c. Click OK.

The installation launchpad opens.

From the launchpad you can view the installation prerequisites and the release notes, you can take a Quick Tour to explore the features, capabilities, and benefits of DB2 Universal Database Version 7, or you can proceed directly to the installation.

4. Select **Install** to begin the installation procedure.
5. After you have initiated the installation from the launchpad, proceed by following the setup program's prompts.

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

Installing DB2 Relational Connect on AIX Systems

Before you install DB2 Relational Connect, make sure you have installed either DB2 Universal Database Enterprise Edition or DB2 Universal Database Enterprise - Extended Edition. If you intend to include DB2 family databases in your distributed requests, you must have selected the Distributed Join for DB2 data sources option during the installation of DB2 Universal Database Enterprise Edition or DB2 Universal Database Enterprise - Extended Edition and verified that the FEDERATED parameter is set to YES.

The following instructions explain how to install DB2 Relational Connect on your AIX system.

The **db2setup** utility is the recommended method for installing DB2 Relational Connect on AIX systems. This utility can perform all of the tasks required to install DB2 Relational Connect.

To install DB2 Relational Connect on your AIX using the **db2setup** utility:

1. Log in as a user with root authority.
2. Insert and mount your DB2 product CD-ROM. For information on how to mount a CD-ROM, refer to *Quick Beginnings for UNIX*.
3. Change to the directory where the CD-ROM is mounted by entering the **cd /cdrom** command, where **cdrom** is the mount point of your product CD-ROM.
4. Enter the **./db2setup** command. After a few moments the Install DB2 Relational Connect V7 window opens.

Selected options are denoted by an asterisk.

When installation is complete DB2 Relational Connect will be installed in the `/usr/lpp/db2_07_01` directory with your other DB2 products.

Adding Oracle Data Sources to a Federated System

To access to Oracle databases using nicknames:

- Step 1.** Install and configure the Oracle client software on the DB2 federated server using the documentation provided by Oracle.

You can use either SQL*Net or Net8 to access both Oracle Version 7 and Oracle Version 8 data sources.

Recommendation for federated servers running on UNIX platforms:

Request a relink of SQL*Net or Net8 during the installation of the Oracle client software.

- Step 2.** Set data source environment variables by modifying the `DB2DJ.ini` file and issuing the **db2set** command. The **db2set** command updates the DB2 profile registry with your settings.

The `DB2DJ.ini` file contains configuration information about the Oracle client software installed on your federated server. In a partitioned database system, you can use a single `DB2DJ.ini` file for all nodes in a particular instance, or you can use a unique `DB2DJ.ini` file for one or more nodes in a particular instance. In a non-partitioned database system, there can be only one `DB2DJ.ini` file per instance.



Proceed to 2b on page 437 if the default settings in `DB2DJ.ini` are acceptable for your configuration.

- a. Edit the DB2DJ.ini file located in sqllib/cfg, and set the following environment variables:

ORACLE_HOME

Set the ORACLE_HOME environment variable to the Oracle home directory; for example:

```
ORACLE_HOME=oracle_home_directory
```

SQL*Net and Net8 require that this variable be set prior to starting your federated instance. If this variable is changed, the federated instance must be stopped and restarted before the new ORACLE_HOME value takes effect.

If an individual user of the federated instance has the ORACLE_HOME environment variable set, it is not used by the federated instance. The federated instance uses only the value of ORACLE_HOME that you set in the DB2 profile registry.

ORACLE_BASE

For federated servers running on versions of UNIX, if you set the ORACLE_BASE variable during the installation of the Oracle client software, you should also set the ORACLE_BASE environment variable on the federated server:

```
ORACLE_BASE=oracle_home_directory
```

ORA_NLS

For federated servers running on versions of UNIX that will be accessing an Oracle 7.2 or later data source, set the ORA_NLS environment variable:

```
ORA_NLS=oracle_home_directory/ocommon/nls/admin/data
```



Refer to “Oracle Code Page Options” on page 441 for more information about National Language Support for Oracle data sources.

TNS_ADMIN

If the SQL*Net or Net8 tnsnames.ora file resides outside the default search path, you must set the TNS_ADMIN environment variable to specify the location of the tnsnames.ora file; for example:

```
TNS_ADMIN=x:\path\tnsnames.ora
```

For Windows servers:

The default location of this file varies depending on the client software that is being used:

- If you are using SQL*Net, tnsnames.ora is in the %ORACLE_HOME%\NETWORK\ADMIN directory.
- If you are using Net8, tnsnames.ora is in the %ORACLE_HOME%\NET8\ADMIN directory.

For UNIX servers:

The default location of this file is
\$ORACLE_HOME/admin/util/network

- b. Issue the **db2set** command to update the DB2 profile registry with your changes.

If you are using this DB2DJ.ini file in a non-partitioned database system, or if you want the values in this DB2DJ.ini file to apply to the current node only, issue:

```
db2set DB2_DJ_INI = sql1lib/cfg/db2dj.ini
```

If you are using this DB2DJ.ini file in a partitioned database system and you want the values in this DB2DJ.ini file to apply to all nodes within this instance, issue:

```
db2set -g DB2_DJ_INI = sql1lib/cfg/db2dj.ini
```

If you are using this DB2DJ.ini file in a partitioned database system, and you want the values in this DB2DJ.ini file to apply to a specific node, issue:

```
db2set -i INSTANCEX 3 DB2_DJ_INI = sql1lib/cfg/node3.ini
```

where:

- INSTANCEX is the name of the instance.
- 3 is the node number as listed in the db2nodes.cfg file.
- node3.ini is the modified and renamed version of the DB2DJ.ini file.

- Step 3. Ensure that the SQL*Net or Net8 tnsnames.ora file is updated for each Oracle server to which communications are configured.

Within the tnsnames.ora file, the SID is the name of the Oracle instance, and the HOST is the host name where the Oracle server is located.

- Step 4. Recycle the DB2 instance:

For Windows servers:

```
NET STOP instance_name
NET START instance_name
```

For UNIX servers:

```
db2stop  
db2start
```

- Step 5. Use the CREATE WRAPPER statement to define the wrapper library that will be used to access Oracle data sources. Wrappers are the mechanism that federated servers use to communicate with and retrieve data from data sources. The following example shows a CREATE WRAPPER statement:

```
CREATE WRAPPER SQLNET
```

where SQLNET is the default name of the wrapper module used with Oracle's SQL*Net client software. If using Oracle's Net8 client software, use NET8.

You can substitute the default name with a name that you choose; however, if you do so, you also must include the LIBRARY parameter and the name of the wrapper library for your DB2 server platform. Refer to the *SQL Reference* for more information about wrapper library names.

- Step 6. Optional: Set the DB2_DJ_COMM environment variable to include the wrapper library that corresponds to the wrapper module that you created in the previous step; for example:

```
db2set DB2_DJ_COMM = libsqlnet.a
```

The DB2_DJ_COMM environment variable controls whether a wrapper module is loaded when the federated server initializes, which can result in improved performance when the Oracle data source is first accessed. Refer to the *SQL Reference* for more information about wrapper library names.

- Step 7. Use the CREATE SERVER statement to define each Oracle server to which communications are configured; for example:

```
CREATE SERVER ORASERVER TYPE ORACLE VERSION 7.2 WRAPPER SQLNET  
OPTIONS (NODE "oranode")
```

where:

- ORASERVER is a name that you assign to the Oracle server. This name must be unique.
- ORACLE is the type of data source to which you are configuring access.
- 7.2 is the version of Oracle that you are accessing.
- SQLNET is the wrapper name that you defined in the CREATE WRAPPER statement.
- oranode is the name of the node where ORASERVER resides. Obtain the node value from the tnsnames.ora file. This value is case

sensitive. Figure 6 illustrates the relationship of the node option and the tnsnames.ora file.

Although the node value is specified as an option, it is required for Oracle data sources. Refer to the *SQL Reference* for a comprehensive list of options.

Figure 6 shows the information in the tnsnames.ora file, the SYSCAT.SERVEROPTIONS view, and the SYSCAT.SERVERS view.

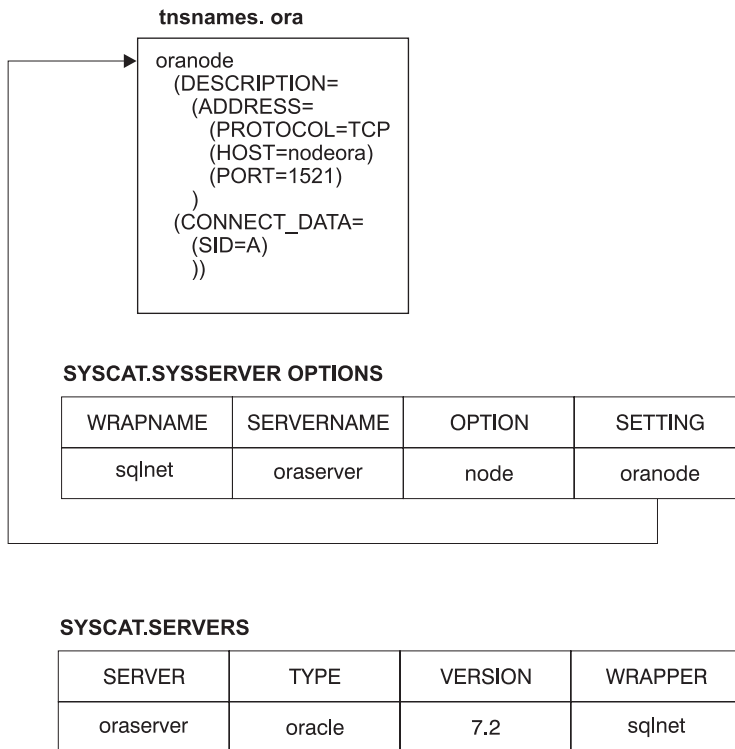


Figure 6. Relationship between DB2 System Files and the Oracle tnsnames.ora File

Step 8. If a user ID or password at the federated server is different from a user ID or password at an Oracle data source, use the CREATE USER MAPPING statement to map the local user ID to the user ID and password defined at the Oracle data source; for example:

```

CREATE USER MAPPING FOR DB2USER SERVER ORASERVER
OPTIONS ( REMOTE_AUTHID 'orauser', REMOTE_PASSWORD "day11te")
  
```

where:

- DB2USER is the local user ID that you are mapping to a user ID defined at an Oracle data source.

- ORASERVER is the name of the Oracle data source that you defined in the CREATE SERVER statement.
- orauser is the user ID at the Oracle data source to which you are mapping DB2USER. This value is case sensitive.

Restriction:

The Oracle user ID (at the Oracle data source, not at the DB2 federated server) must have been created using the Oracle **create user** command with the 'identified by' clause, instead of the 'identified externally' clause.

- day11te is the password associated with "orauser". This value is case sensitive.

Step 9. Use the CREATE NICKNAME statement to assign a nickname to a view or table located at your Oracle data source. You will use this nickname when you query the Oracle data source. The following example shows a CREATE NICKNAME statement:

```
CREATE NICKNAME ORASALES FOR ORASERVER.SALESDATA.MIDWEST
```

where:

- ORASALES is a unique nickname for the Oracle table or view.
- ORASERVER.SALESDATA.MIDWEST is a three-part identifier that follows this format:

data_source_name.remote_schema_name.remote_table_name

Refer to the *SQL Reference* for more information about the CREATE NICKNAME statement.

Refer to the *Administration Guide* for more information about nicknames in general.

Step 10. Repeat the previous step for all database objects that you want create nicknames for.

Step 11. For each HOST in the DESCRIPTION section of the tnsnames.ora file, update the /etc/hosts file for UNIX servers and the x:\winnt\system32\drivers\etc\hosts file for Windows servers if necessary.

Whether you must update this file depends on how TCP/IP is configured on your network. Part of the network must translate the remote host name specified in the DESCRIPTION section in the tnsnames.ora file ("oranode" in the example) to an address. If your network has a name server that recognizes the host name, you do not need to update the TCP/IP hosts file. Otherwise, you need an entry for the remote host. See your network administrator to determine how your network is configured.



If you need more information about Oracle code page mappings, continue to “Oracle Code Page Options”.

To verify that you successfully configured your federated server to access data sources, go to “Verifying Connections to Oracle Data Sources”.

Oracle Code Page Options

Table 36 provides equivalent Oracle options for common NLS code pages. Either your Oracle data sources either must be configured to correspond to these equivalents, or the client code must be able to detect the mismatch and flag it as an error or map the data by using its own semantics. Refer to your data source documentation for more information.

Table 36. Oracle Code Page Options

Code page	Equivalent Oracle option
850	NLS_LANG=American_America.US7ASCII
932	NLS_LANG=Japanese_Japan.JA16SJIS
1046	NLS_LANG=Arabic_UnitedArabEmirates.US7ASCII
819	NLS_LANG=German_Germany.WE8ISO8859P1
912	NLS_LANG=German_Germany.EE8ISO8859P2
1089	NLS_LANG=Arabic_UnitedArabEmirates.AR8ISO8859P6
813	NLS_LANG=Greek_Greece.EL8ISO8859P7
916	NLS_LANG=American_America.IW8ISO8859P8
920	NLS_LANG=Turkish_Turkey.TR8ISO8859P9
950	NLS_LANG=Chinese_Taiwan.ZHT16BIG5
970	NLS_LANG=Korean_Korea.KO16KSC5601
1383	NLS_LANG=Chinese_China.ZHS16CGB231280

Verifying Connections to Oracle Data Sources

This section explains how to verify that you have correctly configured your federated system to access Oracle data sources. You must have completed all the steps in “Adding Oracle Data Sources to a Federated System” on page 435 before continuing with this section.

Step 1. Log on to the system with a user ID that has SYSADM or SYSCTRL authority.

Step 2. Create the sample database on your DB2 federated server by issuing the following command:

```
DB2SAMPL
```

Step 3. Connect to the sample database on your DB2 federated server:

```
CONNECT TO SAMPLE
```

Step 4. Issue the following SQL statement:

```
SELECT * FROM SYSCAT.SYSTABLES
```

You should receive the entire contents of the SYSCAT.TABLES system catalog table.

Step 5. Add the Oracle system catalog table ALL_TABLES to your federated environment:

- When issuing the CREATE USER MAPPING statement, make sure the user ID at the data source holds at least SELECT permission for the sample table.
- Create a nickname for the ALL_TABLES system catalog table.

Step 6. Using the nickname for the Oracle system catalog table, issue a SELECT statement for a table at your Oracle data source; for example:

```
SELECT * FROM nickname
```

where *nickname* is the nickname for Oracle sample database's system catalog table

The result set you receive should contain all the columns and rows in ALL_TABLES.

After you successfully select data from both the federated database and the Oracle data source, you can try joining data from the two data sources to complete the verification procedure.

Chapter 27. Setting Up a Federated System to Access OLE DB Data Sources

This chapter explains how to configure your federated server to access OLE DB providers using OLE DB table functions.

Enabling Federated Database Functionality

To take advantage of federated database functionality, during installation of DB2 Universal Database you must have selected the Distributed Join for DB2 Databases option. This option installs libraries in SQLLIB/lib that enable your DB2 federated server to access OLE DB providers using OLE DB table functions.

Adding OLE DB data sources to a Federated System

To access OLE DB data sources using OLE DB table functions:

1. Install and configure OLE DB 2.0 or higher and OLE DB provider for data source. Follow the software requirements from your OLE DB provider.
2. Use the CREATE WRAPPER statement to define the wrapper library that will be used to access OLE DB providers.

Wrappers are the mechanism that federated servers use to communicate with and retrieve data from data sources. The following example shows a CREATE WRAPPER statement:

```
CREATE WRAPPER OLEDB
```

where OLEDB is the default name of the wrapper module used with OLE DB providers. You can substitute the default name with a name that you choose; however, if you do so, you also must include the LIBRARY parameter and the name of the wrapper library for your federated server platform. See the *SQL Reference, Volume 2* for more information about wrapper library names.

3. Use the CREATE SERVER statement to define a server name for an OLE DB data source. For example:

```
CREATE SERVER Nwind
WRAPPER OLEDB
OPTIONS (
CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;
Data Source=c:\msdasdk\bin\oledb\nwind.mdb',
COLLATING_SEQUENCE 'Y');
```

where:

- Nwind is a name that you assign to an OLE DB data source.
- OLEDB is the wrapper name.
- OPTIONS lists other parameters. In this example:
 - CONNECTSTRING provides initialization properties needed to connect to a data source. The string contains a series of keyword and value pairs separated by semicolons. The equal sign (=) separates each keyword and its value. Keywords are the descriptions of the OLE DB initialization properties (property set DBPROPSET_DBINIT) or provider-specific keywords.

For the complete syntax and semantics of the CONNECTSTRING option, see the *Microsoft OLE DB 2.0 Programmer's Reference and Data Access SDK*, Microsoft Press, 1998.

- COLLATING_SEQUENCE specifies whether the data source uses the same collating sequence as DB2 Universal Database. Valid values are Y (the same collating sequence is used) and N (a different collating sequence is used). If a COLLATING_SEQUENCE is not specified, the data source is assumed to have a different collating sequence than DB2 Universal Database.
4. If a user ID or password at the federated server is different from a user ID or password at an OLE DB data source, use the CREATE USER MAPPING statement to map the local user ID to the user ID and password defined at the OLE DB data source; for example:

```
CREATE USER MAPPING FOR john
SERVER Nwind
OPTIONS (REMOTE_AUTHID 'dave', REMOTE_PASSWORD 'mypwd');
```

where:

- john is the local user ID that you are mapping to a user ID defined at an OLE DB data source.
 - Nwind is the name of the OLE DB data source that you defined in the CREATE SERVER statement.
 - dave is the user ID at the OLE DB data source to which you are mapping john. This value is case sensitive.
 - mypwd is the password associated with dave. This value is case-sensitive.
5. You can use the server name Nwind to identify the OLE DB provider by using the CREATE FUNCTION statement:

```
CREATE FUNCTION orders ()
RETURNS TABLE (orderid INTEGER, ...)
LANGUAGE OLEDB
EXTERNAL NAME 'Nwind!orders';
```

Part 9. Appendixes

Appendix A. Basic Task Knowledge

This section describes the basic tasks that you will need to know to use this product effectively.



Go to the task that you want to perform:

- “Starting the Client Configuration Assistant”.
 - “Starting the DB2 Control Center”.
 - “Entering Commands Using the Command Center” on page 448.
 - “Entering Commands Using the Command Line Processor” on page 449.
 - “Working with the System Administrative Group” on page 451.
 - “Granting Advanced User Rights on Windows” on page 452.
 - “Working with the Business Intelligence Functions” on page 453.
 - “Mounting CD-ROMs on UNIX Operating Systems” on page 453.
 - “Setting the Number of Licensed Processors” on page 456.
 - “Upgrading DB2 from Try and Buy Mode” on page 457.
-

Starting the Client Configuration Assistant

Start the Client Configuration Assistant (CCA) as follows:

OS/2 Click on **OS/2 Warp**, and select **IBM DB2 —> Client Configuration Assistant**

Windows 32-bit operating systems

Click on **Start** and select **Programs—>IBM DB2—>Client Configuration Assistant**

You can also start the CCA by entering the **db2cca** command at a command prompt.

Starting the DB2 Control Center

You can run the DB2 Control Center as a Java *application* or as a Java *applet*.

To run the Control Center as an application

Enter the **db2cc** command. Your system must have the correct Java Runtime Environment installed in order to run the Control Center as an application.

On Windows 32-bit and OS/2 systems, you can also start the Control Center as an application by invoking the **Control Center** icon in your **IBM DB2** program group.

To run the Control Center as an applet

You must have a Java-enabled browser and you must perform some additional configuration steps to run the Control Center as an applet. For detailed instructions on running the Control Center as an applet or application, see “Chapter 8. Control Center Installation and Configuration” on page 115.

Entering Commands Using the Command Center

This section describes how to enter commands using the Command Center. There are two versions of the Command Center. This section documents the Command Center that is accessible from the DB2 Control Center.

Note: If you do not have the Control Center installed, a Command Center with limited functionality is available through the IBM DB2 program group, or by entering the **db2ctr** command.

From the Command Center you can:

- Run SQL statements, DB2 commands, and operating system commands.
- See the execution result of SQL statements and DB2 commands in a results window. You can scroll through the results and save the output to a file.
- Save a sequence of SQL statements and DB2 commands to a script file. You can then schedule the script to run as a job. When a saved script is modified, all jobs dependent on the saved script inherit the new modified behavior.
- Recall and run a script file.
- See the execution plan and statistics associated with a SQL statement before execution.
- Get quick access to database administrative tools from the main tool bar.
- Display all the command scripts known to the system through the Script Center, with summary information listed for each.
- Use the SQLAssist tool to build complex queries.
- Display results in an table you can edit.

To start the Command Center, click on the **Command Center** icon in the Control Center.

The Command Center contains a large input area for entering commands. To run the commands you have entered, click on the **Execute** icon (the gears icon).



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

```
list database directory
```

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

```
!dir
```

If you want to enter multiple commands, you must end each command with the termination character, then press **Enter** to start the next command on a new line. The default termination character is a semicolon (;).

For example, you could connect to a database called SAMPLE and list all the system tables by entering the following command:

```
connect to sample;  
list tables for system
```

After you have clicked on the **Execute** icon, the results are displayed.

To recall commands that you have entered during your session, select the **Command history** drop down box, and select a command.

To save commands, select **Interactive** —> **Save Command As** from the menu bar. For more information, click on the **Help** push button or press the **F1** key.



You can use the **Append to Script** button and the Script page of the Command Center to store commonly used SQL statements or DB2 commands as scripts. For more information, click on the **Help** push button or press the **F1** key.

Entering Commands Using the Command Line Processor

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

DB2 Command Window

The DB2 command line processor behaves like a command window from your operating system. You can enter operating system commands, DB2 commands, or SQL statements and view their output.

Interactive Input Mode

The db2 prefix that you use for DB2 commands (in the DB2 Command Window) is pre-entered for you. You can enter operating systems commands, DB2 commands, or SQL statements and view their output.

File Input Mode

Processes commands that are stored in a file. For information on the file input mode, refer to the *Command Reference*.

DB2 Command Window

To invoke a DB2 Command Window, do the following:

OS/2 Open any OS/2 command window.

Windows 32-bit operating systems

Click on **Start** and select **Programs** → **IBM DB2** → **Command Window**

You can also invoke the DB2 command window by entering the **db2cmd** command at your operating system's prompt.

UNIX Open any operating system command window.

If you are entering commands via the Command Window, you must include the db2 prefix. For example:

```
db2 list database directory
```



If the DB2 command contains characters that have special meaning on the operating system you are using, you will need to enter the command in quotation marks to ensure that it is run properly.

For example, the following command would retrieve all the information from the *employee* table, even if the * character has a special meaning on the operating system:

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

To enter a long command that does not fit on a single line, you must use a space followed by the line continuation character "\ " at the end of one line, then press the **Enter** key to continue the command on to the next. For example:

```
db2 select empno, function, firstname, lastname, birthdate, from \  
db2 (cont.) => employee where function='service' and \  
db2 (cont.) => firstname='Lily' order by empno desc
```

Interactive Input Mode

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

OS/2 Click on **OS/2 Warp** and select **IBM DB2** → **Command Line Processor** or enter the **db2** command.

Windows 32-bit operating systems

Click on **Start** and select **Programs** → **IBM DB2** → **Command Line Processor**.

You can also invoke the command line processor in interactive input mode by entering the **db2cmd** command followed by the **db2** command at your operating system's prompt.

UNIX Enter the **db2** command from the command line processor

In interactive input mode, the prompt looks like this:

```
db2 =>
```

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

```
db2 => list database directory
```

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

```
db2 => !dir
```

To enter a long command that does not fit on a single line, you must use a space followed by the line continuation character "\" at the end of one line, then press the **Enter** key to continue the command on to the next. For example:

```
db2 select empno, function, firstname, lastname, birthdate, from \
db2 (cont.) => employee where function='service' and \
db2 (cont.) => firstname='Lily' order by empno desc
```

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

For more information on advanced topics using the CLP, refer to the *Command Reference*.

Working with the System Administrative Group

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

OS/2 Any valid DB2 user ID which belongs to the Administrator or Local Administrator group.

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

Windows 9x Any Windows 9x user.

Windows NT and Windows 2000

Any valid DB2 user account which belongs to the local Administrators group on the machine where the account is defined.

For example, if a user logs on to a domain account and tries to access a DB2 database, DB2 will go to a Domain Controller to enumerate groups (including the Administrator's group). You can change this behavior in either of two ways:

1. Set the registry variable `DB2_GRP_LOOKUP=local` and add the domain accounts (or global groups) to the local Administrators group.
2. Update the database manager configuration parameter `SYSADM_GROUP` to specify a new group. If you want that group enumerated on the local machine, you must also set the `DB2_GRP_LOOKUP` registry variable.

For a domain user to have SYSADM authority, it must belong to the Administrators group on the Domain Controller. Since DB2 always performs authorization at the machine where the account is defined, adding a domain user to the local Administrators group on the server does not grant the domain user SYSADM authority to this group.

To avoid adding a domain user to the Administrators group at the Domain Controller, we suggest that you create a global group and add the domain users that you want to grant SYSADM authority, and then update the DB2 configuration parameter `SYSADM_GROUP` with the name of the global group. To do so, enter the following commands:

```
db2stop
db2 update dbm cfg using sysadm_group global_group
db2start
```

For information on how to change the default SYSADM settings and how to assign this authority to a different user or set of users, refer to the *Administration Guide*.

Granting Advanced User Rights on Windows

Windows NT

To grant advanced user rights on Windows NT you must be logged on as a Local Administrator. Perform the following steps to grant user rights:

1. Click **Start** and select **Programs** → **Administrative Tools (Common)** → **User Manager for Domains**.
2. In the User Manager window, select **Policies** → **User Rights** from the menu bar.
3. In the User Rights Policy window, select the **Show Advanced User Rights** check box then in the **Right** drop down box, select the user right you want to grant. Click **Add**.

4. In the Add Users and Groups window select the user or a group you want to grant the right to and click **OK**.
5. In the User Rights Policy window, select the user or group you have added from the **Grant To** list box and click **OK**.

Windows 2000

To grant advanced user rights on Windows 2000 you must be logged on as a Local Administrator. Perform the following steps to grant user rights:

1. Click **Start** and select **Settings** → **Control Panel** → **Administrative Tools**.
2. Select **Local Security Policy**.
3. In the left window pane, expand the **Local Policies** object, then select **User Rights Assignment**.
4. In the right window pane, select the user right that you want to assign.
5. From the menu, select **Action** → **Security...**
6. Click **Add**, then select a user or group to assign the right to, and click **Add**.
7. Click **OK**.

Working with the Business Intelligence Functions

The Business Intelligence Tutorial walks you through several basic and advanced tasks using the Data Warehouse Center and the OLAP Starter Kit. You can start the Tutorial from the **Help** menu in the Data Warehouse Center or from the **Help** menu in the OLAP Starter Kit desktop. You can also start the Tutorial from the Getting Started item in the Information Center.

Mounting CD-ROMs on UNIX Operating Systems

The following sections describe how to mount your DB2 product CD-ROM on UNIX-based operating systems.

Mounting the CD-ROM on AIX

To mount the CD-ROM on AIX using the System Management Interface Tool (SMIT), perform the following steps:

1. Log in as a user with root authority.
2. Insert the CD-ROM in the drive.
3. Create a CD-ROM mount point by entering the `mkdir -p /cdrom` command, where `cdrom` represents the CD-ROM mount point directory.
4. Allocate a CD-ROM file system using SMIT by entering the **smit storage** command.
5. After SMIT starts, select **File Systems** → **Add / Change / Show / Delete File Systems** → **CDROM File Systems** → **Add CDROM File System**.

6. In the Add a File System window:
 - Enter a device name for your CD-ROM file system in the **DEVICE Name** field. Device names for CD-ROM file systems must be unique. If there is a duplicate device name, you may need to delete a previously-defined CD-ROM file system or use another name for your directory. In our example, we will use `/dev/cd0` as the device name.
 - Enter the CD-ROM mount point directory in the **MOUNT POINT** window. In our example, the mount point directory is `/cdrom`.
 - In the **Mount AUTOMATICALLY at system restart** field, select yes to enable automatic mounting of the file system.
 - Click **OK** to close the window, then click **Cancel** three times to exit SMIT.
7. Next, mount the CD-ROM file system by entering the **smit mountfs** command.
8. In the Mount a File System window:
 - Enter the device name for this CD-ROM file system in the **FILE SYSTEM name** field. In our example, the device name is `/dev/cd0`.
 - Enter the CD-ROM mount point in the **Directory over which to mount** field. In our example, the mount point is `/cdrom`.
 - Enter `cdarfs` in the **Type of Filesystem** field. To view the other kinds of file systems you can mount, click **List**.
 - In the **Mount as READ-ONLY system** field, select to yes.
 - Accept the remaining default values and click **OK** to close the window.

Your CD-ROM file system is now mounted. To view the contents of the CD-ROM, place the disk in the drive and enter the `cd /cdrom` command where `cdrom` is the CD-ROM mount point directory.

Mounting the CD-ROM on HP-UX

Because DB2 Version 7.1 for HP-UX contains several files with long file names, the mount command may fail. The following steps will enable you to mount successfully your DB2 for HP-UX product CD-ROM:

1. Log in as a user with root authority.
2. In the `/etc` directory, add the following line to the `pfs_fstab` file:

```
/dev/dsk/c0t2d0 mount_point pfs-rrip ro,hard
```

where `mount_point` represents the mount point of the CD-ROM.

3. Start the `pfs` daemon by entering the following commands (if they are not already running):

```
/usr/sbin/pfs_mountd &
/usr/sbin/pfsd 4 &
```

4. Insert the CD-ROM in the drive and enter the following commands:

```
mkdir /cdrom
/usr/sbin/pfs_mount /cdrom
```

where */cdrom* represents the mount point of the CD-ROM.

5. Log out.

Mounting the CD-ROM on Linux

To mount the CD-ROM on Linux:

1. Log in as a user with root authority.
2. Insert the CD-ROM in the drive and enter the following command:

```
mount -t iso9660 -o ro /dev/cdrom /cdrom
```

where */cdrom* represents the mount point of the CD-ROM.

3. Log out.

Note that some window managers may automatically mount your CD-ROM for you. Consult your system documentation for more information.

Mounting the CD-ROM on PTX

To mount the CD-ROM on PTX:

1. Log in as a user with root authority.
2. Insert the CD-ROM in the drive and enter the following commands:

```
mkdir /cdrom
mount -r -F cdfs /dev/dsk/cd0 /cdrom
```

where */cdrom* is the mount point of the CD-ROM.

3. Log out.

Mounting the CD-ROM on Solaris

To mount the CD-ROM on Solaris:

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

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

where */cdrom/unnamed_cdrom* represents the CD-ROM mount directory and */dev/dsk/c0t6d0s2* represents the CD-ROM drive device.

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

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

```
/cdrom/unnamed_cdrom
```

4. Log out.

Setting the Number of Licensed Processors

Note: This section applies only to DB2 Enterprise Edition, DB2 Enterprise-Extended Edition, and DB2 Warehouse Manager.

If you are using an SMP machine and you have purchased additional processor entitlements, you can update this information using the **db2licm** command.

Note: An instance must be created on UNIX-based operating systems before performing these steps.

To update the number of License Processors, perform the following steps:

1. Log in as a user with SYSADM, SYSCTRL or SYSMANT authority.
2. The **db2licm** utility can be found in the following locations:
 - For Unix-based operating systems, if `INSTHOME/sql11ib/adm` is not in your `PATH`, change your directory.
 - For Windows 32-bit operating systems and OS/2, go to `x:\DB2DIR\bin`, where `x:\DB2DIR\` is your DB2 installation drive and path.
3. Obtain the product password by issuing the **db2licm -l** command. The DB2 products are listed as follows:

Enterprise-Extended Edition

DB2UDBEEE DB

Enterprise Edition

DB2UDBEE DB2

Warehouse Manager

DB2UDBWM DB2

Relational Connect

DB2RELC DB2

Spatial Extenders

DB2UDBGSE

4. Update the number of processors using:
`db2licm -n [product password] [number of processors]`

Upgrading DB2 from Try and Buy Mode

You can upgrade a DB2 product from the Try and Buy mode to a licensed version in two ways. You can use the command line utility called **db2licm** or use the License Center. The License Center is a built-in feature of the Control Center.

On UNIX Operating Systems

To add a license using the command line:

1. Log in as a user with root authority.
2. The **db2licm** utility can be found in the following locations:
 - For AIX, `/usr/lpp/db2_07_01/adm/`
 - For LINUX, `/usr/IBMDB2/V7.1/adm/`
 - For HP-UX, PTX, Solaris, PTX, `/opt/IBMDB2/V7.1/adm/`
3. Issue the following command:

```
db2licm filename.lic
```

where *filename.lic* represents the name of the licensing file. For more information about the **db2licm** command, refer to the *Command Reference*.

On OS/2 and Windows 32-bit Operating Systems

To add a license using the command line:

1. Go to the `<install directory>\bin` directory, where *<install directory>* represents the directory where you installed the product.
2. To add a license, issue the following command:

```
db2licm path/filename.lic
```

License files are contained in the `db2/license` directory of the installation CD.

For more information about the **db2licm** command, refer to the *Command Reference*.

Adding a license using the License Center

To add a license using the License Center:

1. Launch the Control Center.
2. Select **License Center** from the **Tools** menu list.
3. Refer to the online help for the License Center available from the Control Center.

Appendix B. Working with NetQuestion

This chapter describes installation prerequisites, workarounds for known issues, advanced configuration, and uninstall procedures for DB2's online search system called NetQuestion.

NetQuestion is installed automatically if you choose to install product documentation with any DB2 product on Windows 32-bit, OS/2, AIX, HP-UX, and Solaris operating systems. It is also installed as part of the base operating system on systems running AIX 4.3.2 and later.

Provided your operating system meets NetQuestion's installation prerequisites, which are the usually same as DB2, there is nothing special you have to do to install and work with NetQuestion. You can begin searching the DB2 information after you have finished installing your DB2 product.

NetQuestion Overview

When you install DB2 online documentation, the NetQuestion search system is also installed. The following components make up the search system:

- The DB2 Information page containing a **Search** field. The location of this page on each operating system is provided below.
- A basic web server. If you will be serving DB2 documentation a client/server environment, for security reasons we recommend that you use a third-party web server such as Apache, Netscape Enterprise Server or Lotus Domino Go!
- A search CGI. This executable passes queries between the web server and the NetQuestion search server.
- A NetQuestion search server. The search server is autostarted after you install the DB2 online documentation.
- One or more predefined document indices. There is one index for the online books and one for the Control Center online help for each supported language. The book indices are named DB2S71xx and the Control Center help indices are named DB2C71xx, where xx represents the two-letter language identifier of the installed documentation.
- One or more directories containing the online documentation in HTML format.

There are two versions of NetQuestion:

- The *single-byte character set* (SBCS) version is installed with documentation written in single-byte languages such as English, French, German, Italian, and Spanish.
- The *double-byte character set* (DBCS) version is installed with documentation written in Japanese, Simplified Chinese, Traditional Chinese, and Korean.

Searching the DB2 Online Information

To search the DB2 online information, launch the DB2 Information page in a browser, enter your search terms, and click **Search**.

The location of the DB2 Information page varies from platform to platform:

On Windows 9x, Windows NT and Windows 2000

Click on **Start** → **Programs** → **IBM DB2** → **Information** → **DB2 Information**. You can also open `x:\sql11ib\doc\html\index.htm`, where `x:` is your DB2 installation drive.

On OS/2

Open the **IBM DB2** folder and double-click on **DB2 Information**. You can also open `x:\sql11ib\doc\html\index.htm`, where `x:` is your DB2 installation drive.

On AIX, HP-UX and Solaris

Open `DB2DIR/doc/en_US/html/index.htm`, where `DB2DIR` represents `/usr/lpp/db2_07_01` on AIX and `/opt/IBMdb2/V7.1` on HP-UX and Solaris.

Solving Search Problems

You may occasionally encounter an error when searching the DB2 online information. If you do encounter a search error, try the following troubleshooting tips:

Is the search server running?

If you receive a return code 33 when you attempt to search the DB2 documentation, the NetQuestion search server is not running.

To start the search server on Windows 9x, Windows NT and Windows 2000, click **Start** and go to **Programs** → **IBM DB2** → **Start HTML Search Server**. You can also start the search server by entering one of the following commands:

```
x:\imnnq_nt\imnss start server //for SBCS
x:\imnnq_nt\imqss -start dbschelp //for DBCS
```

Note that the NetQuestion install directory on Windows 9x is `x:\imnnq_95`.

To start the search server on OS/2, open the **IBM DB2** folder and click on the **Start HTML Search Server** icon. You can also enter one of the following commands:

```
x:\imnnq\imnss start server //for SBCS
x:\imnnq\imqss -start dbcshelp //for DBCS
```

To start the search server on AIX, HP-UX and Solaris systems, enter one of the following commands:

On AIX

Enter the `/usr/IMNSearch/bin/imnss -start imnhelp` command for SBCS installations. The **imnss -start** command can also be executed without typing the full path name if `/usr/bin` is in your PATH.

Enter the `/usr/IMNSearch/bin/imqss -start dbcshelp` command for DBCS installations. In DBCS installations, you must ensure that the `IMQCONFIGSRV` and `IMQCONFIGCL` environment variables are set. You can set these variables by executing the `/usr/IMNSearch/bin/. imq_env` command. Again, you can execute both the **imqss -start** and **. imq_env** commands without typing the full pathname if `/usr/bin` is in your PATH.

On HP-UX

Enter the `/sbin/rc2.d/S990IMNSearch start` command for SBCS and DBCS installations. This command will also start the NetQuestion web server, if it is not already running.

On Solaris

Enter the `/etc/rc2.d/S90IMNSearch start` command for SBCS and DBCS installations. This command will also start the NetQuestion web server, if it is not already running.

To stop the search server, replace `start` with `stop` in the above commands.

Is the NetQuestion web server running?

NetQuestion comes with its own basic web server. You can also use your own web server with NetQuestion. You should make sure that the web server you are using is running. The following information applies to the NetQuestion web server. To start other web servers, consult your web server's product documentation.

On Windows NT and Windows 2000 systems, check the Task Manager to see if the `httpd1.exe` process is running. If it is not running, start it by entering the `x:\imnnq_nt\httpd1 -r httpd.cnf` command, where `x:` is your DB2 installation drive.

On OS/2, run the `x:\sqllib\bin\db2netqd start` command, where `x:` is your DB2 installation drive. The NetQuestion web server and the search server will be started if they are not already running. You can also run the `pstat | find "httpd1"` command to check if the process is running.

On AIX, HP-UX and Solaris, enter the `ps -ef | grep httpd1ite` command to see if the `httpd1ite` process is running. If it is not running, start it by entering one of the following commands:

On AIX

```
/usr/IMNSearch/httpd1ite/httpd1ite -r  
/etc/IMNSearch/httpd1ite/httpd1ite.conf
```

On HP-UX

```
/sbin/rc2.d/S990IMNSearch start
```

On Solaris

```
/etc/rc2.d/S90IMNSearch start
```

Are the document indices registered with the search server?

DB2's document indices are registered with the search server during installation. To check if the indices are registered properly:

1. Enter the `imndomap -a` or `imqdomap -a` command to determine which indices are installed, and where they are installed. These commands should return one or more indices with names similar to `DB2S71xx` or `DB2C71xx`, where `xx` is the two-letter language identifier of the installed documentation. If this is not the case, re-install your DB2 product and be sure to select to install the product documentation component.

If the indices still are not registered with the search server, you can try to register them manually. For more information on manual index registration, refer to your operating system's section later in this chapter.

2. Enter the `imnixsta` (or `imqixsta` for DBCS) `INDEX_NAME` command, where `INDEX_NAME` is one of the index names returned by the command in step 1. The output from this command should indicate, in part, that search is available.
3. DB2 indices should be located in the following directories:

```
x:\sqllib\doc\html           //Windows and OS/2  
/var/docsearch/indexes      //AIX, HP-UX, and Solaris
```

Is TCP/IP configured properly?

NetQuestion uses TCP/IP for communication on all systems except on OS/2, which uses Named Pipes (although TCP/IP is also supported). Therefore, your system must have TCP/IP installed and configured properly, and your system must be able to resolve localhost if you are

searching documentation installed locally. For more information on configuring TCP/IP, see your operating system's section later in this chapter.

Is the search CGI in the correct location, and is it named correctly?

The DB2 search CGI must be located in a specific directory. On Windows and OS/2 operating systems, this directory is the NetQuestion installation direction which can be determined by entering the **echo %IMNINSTSRV%** command for SBCS installations, or **echo %IMQINSTSRV%** command for DBCS installations. On UNIX operating systems, the search CGI is installed in `/var/docsearch/cgi-bin`.

On Windows and OS/2 operating systems, the SBCS search CGI is named `db2srsXX.exe` and the DBCS version is named `db2srdXX.exe`, where `XX` represents the two-letter language identifier of the installed documentation.

On UNIX operating systems, the SBCS search CGI is named `db2srbsbc` and the DBCS search CGI is named `db2srdbsc`. There are no language-specific NetQuestion search CGIs on UNIX operating systems.

Also, make sure that the name of your search CGI matches the one in the `<form action="http...">` tag in the DB2 search form. For example, in the English search form on Windows or OS/2 this tag should read `<form action="http://localhost:49213/cgi-bin/db2srbsen.exe" method="POST">`.

Are the correct books or help files installed, and are they in the correct directory?

If you received a "File Not Found (Error 404)" message, make sure the links on the search results page are pointing to a valid URL. If your DB2 documentation is installed locally, all URLs should begin with `file://`. For documentation that is being served from another computer, all URLs should begin with `http://`.

Are you using the correct search parameters?

When you are searching for words or phrases, you should keep the following in mind:

- To search using wildcards, use a question mark (?) for any single character or an asterisk (*) as a placeholder for zero or more characters.
- When using wildcards you may encounter a return code 22 in the search results page. This means that your search request was too complex. You should reformulate your request so that it is more specific. For example, searching for `DB*` on all of the books and online help may produce a return code 22.

- Enclose phrases in double quotes.
- To include a particular term or phrase in your search results, precede the word or phrase with a plus sign (+). You can exclude words or phrases from your search results by preceding them with a minus sign (-).
- Boolean operators such as AND, OR, and NOT are not supported. Use plus or minus signs instead.

For additional platform-specific troubleshooting information, see the following sections:

- “NetQuestion for Windows 32-bit Operating Systems”.
- “NetQuestion for OS/2 Operating Systems” on page 470.
- “NetQuestion for UNIX Operating Systems” on page 475.

NetQuestion for Windows 32-bit Operating Systems

This section describes post-installation configuration information, additional troubleshooting tips, and workarounds for known issues with NetQuestion on Windows 32-bit operating systems. It also describes how to remove NetQuestion from your system.

Diagnosing NetQuestion Install Errors

If you received error messages while installing NetQuestion, perform the following steps to diagnose and fix the problem:

1. Look in the `<temp>\imnnq\install` directory, where `<temp>` is your system's %TEMP% directory, to find the `imnnq.err` file. If it does not exist, reboot and try installing the product again. If `imnnq.err` does exist, here are the possible contents:
 - 1 - This indicates that the current PATH is too long and adding the search server into the PATH will cause the entire PATH to be erased.
 Note: The limit on Windows NT 4.0 is 512; on Windows 95 it is 255.
 It is recommended that you perform the following steps:
 - a) Rename the PATH variable in the AUTOEXEC.BAT file (PATHGOOD), save changes and reboot.
 - b) Remove the IMNNQ.ERR file from `<temp>\imnnq\install`.
 - c) Run the product install again to install the search system properly.
 - d) Merge the PATHGOOD variable with the PATH variable that was created by the latest installation.
 - 2 - Miscellaneous error, please contact IBM service.
 - 3 - Out of disk space error. Please ensure that there is at least 4.5MB of disk space for the search system plus enough space for AUTOEXEC.BAT to be changed for Windows 95.
2. If `imnnq.err` contains a message saying `<file_name>.EXE DOES NOT EXIST`, the NetQuestion executables could not be found. In this case you should try installing the DB2 product documentation again. For more troubleshooting information see “Solving Search Problems” on page 460.

TCP/IP Configuration

The following prerequisites are required to install and work with NetQuestion:

- You must have TCP/IP Version 3 or higher installed on your machine. It must be installed and configured in order for NetQuestion to function properly.

For Windows 95 and Windows 98, TCP/IP must be enabled as follows:

- For a LAN Adapter configuration:
 - You must have DNS enabled with a valid host and domain name.
 - Your LAN DNS must resolve localhost to 127.0.0.1.
 - You cannot run disconnected with a LAN adapter configuration.
- For a Dial-Up Adapter configuration:
 - You must have DNS disabled.
 - Your TCP/IP Address must be obtained automatically.

Note: These configuration options will apply to all TCP/IP adapters even though they have only been changed for this one. You will not be able to use both LAN and Dial-Up without reconfiguring.

- Dial-Up Networking (DUN) TCP/IP properties for your internet service provider (ISP) must be configured as documented your ISP. These properties will override the properties in the Dial-Up Adapter TCP/IP properties configured via the **Network** icon in the Windows 95/98 Control Panel, but only if the Dial-Up Adapter properties are configured as above.

Note: Do not enable DNS or set an IP address in the Dial-Up Adapter TCP/IP properties as this will interfere with the DUN configuration for the ISP.

For Windows NT 4.0, either TCP/IP configurations (for DUN or Dial-Up Adapter) detailed above will work. If you are running a stand-alone system not connected to a network, you can also enable the MS Loopback Adapter without the other two adapters.

Changing the Port Number for the Search Server

The search server is assigned to port 49213, a number beyond the public ports assigned for TCP/IP. If you have another product that uses this port, you should change the search server port by performing the following steps:

- Step 1. Edit the `httpd.cnf` file in the NetQuestion directory and change the port number to one you know is available, preferably one above 49000.

- Step 2. Determine where your *db2path* is by entering the **db2set db2path** command.
- Step 3. Go to the *db2path/doc/html* directory and use a text editor to edit the *index.htm* file. In the line with the **<form>** tag, change the value for **localhost:49213** to reflect the port number you selected in step 1.
- Step 4. Stop and re-start the search server. To stop the search server, click on **Start —> Programs —> IBM DB2 —> Stop HTML Search Server**. To start the search server, click on **Start —> Programs —> DB2 for Windows —> Start HTML Search Server**.

Ensuring NetQuestion Environment Variables are set on a LAN-Connected Drive

If you install DB2 on a LAN-connected drive on Windows 9x and the drive is not reconnected before *autoexec.bat* executes, NetQuestion environment variables are not set. NetQuestion sets environment variables through a batch file (*imnenv.bat* for SBCS or *imqenv.bat* for DBCS) that resides in the NetQuestion directory. To bypass this problem, copy *imnenv.bat* or *imqenv.bat* from the NetQuestion directory to another drive/directory that is connected before *autoexec.bat* is executed. Then, modify *autoexec.bat* to call this batch file on start-up. For example, if you copy *imnenv.bat* to *C:\WINDOWS\IMNNQ*, you can add the following line to *autoexec.bat*:

```
IF EXIST C:\WINDOWS\IMNNQ\IMNENV.BAT CALL IMNENV.BAT
```

Locating the NetQuestion Installation Directory

NetQuestion is stored in its own directory because it may be used by other products. For example, if you installed DB2 Universal Database along with NetQuestion on *G*: and later you installed IBM VisualAge for Java on *H*:\, only one search system is installed, the one that was installed first.

Some instructions in this section require you to specify the location of the NetQuestion directory. To locate this directory enter one of the following commands:

```
echo %IMNINSTSRV% //for SBCS versions
echo %IMQINSTSRV% //for DBCS versions
```

Searching With Proxies Enabled in Netscape or Internet Explorer

If you use Netscape or Internet Explorer with proxies enabled manually, you can speed up search significantly by modifying your proxy information.

In Netscape 4, perform the following steps to modify your proxy information:

1. Select **Edit —> Preferences**.
2. Double-click **Advanced** in the **Category** tree.
3. Click **Proxies** in the **Advanced** sub-tree.
4. Click **View at the Manual Proxy Configuration** selection.

5. In the **Exceptions...Do not use proxy servers for domains beginning with** box, type
localhost:49213

If you have other entries here, separate them with commas.

6. Click **OK** until all dialog boxes are closed.

In Internet Explorer 4, perform the following steps to modify your proxy configuration:

1. Select **View** → **Internet Options**.
2. Select the **Connections** tab.
3. Select the **Bypass proxy server for local addresses** box. This check box is available only if you are using a proxy or socks connection and you have selected the **Use a proxy server** box.
4. Click on **Advanced**.
5. In the **Exceptions...Do not use proxy server for addresses beginning with** box, type
localhost:49213
6. Click **OK** until all dialog boxes are closed.

In Internet Explorer 5, perform the following steps to modify your proxy configuration:

1. Select **Tools** → **Internet Options**.
2. Select the **Connections** tab.
If you are connected to a Local Area Network (LAN), click on **LAN Settings**. If you use Dial-Up Networking, click on **Settings**.
3. Select the **Bypass proxy server for local addresses** box. This check box is available only if you are using a proxy or socks connection and you have selected the **Use a proxy server** box.
4. Click on **Advanced**.
5. In the **Exceptions...Do not use proxy server for addresses beginning with** box, type
localhost:49213
6. Click **OK** until all dialog boxes are closed.

Searching with a Laptop on Windows 9x

If you use a laptop that is normally connected to a LAN as a disconnected development platform, you may run into difficulty searching the DB2 product documentation. To search successfully, you need to disable your nameserver IP address in your TCP/IP configuration. In other words, you need two separate TCP/IP configurations, one for connected operations and another for disconnected ones.

Windows 9x only allows you to have a single TCP/IP configuration. However, there are shareware utilities available on the internet that let you specify more than one setting, and then change them according to your connection status. TCPSwitch is one of these programs.

Uninstalling NetQuestion on Windows 32-bit Operating Systems

Before uninstalling NetQuestion, make sure the search server is stopped. For information on how to stop the search server, see “Solving Search Problems” on page 460.

NetQuestion is normally uninstalled when you uninstall DB2 Universal Database. If NetQuestion is still present after properly uninstalling DB2 and rebooting your system, go to “Diagnosing NetQuestion Uninstall Problems” diagnose and fix the problem.

Diagnosing NetQuestion Uninstall Problems

If NetQuestion is not uninstalled after properly uninstalling DB2 and rebooting your system, it could mean that DB2 or other IBM products are still registered with NetQuestion. To determine which products are still registered with NetQuestion, issue one of the following commands:

```
imndomap -a          //for SBCS
imqdomap -a          //for DBCS
```

If this command lists no indexes, go to “Manually Uninstalling NetQuestion on Windows 32-bit operating systems” on page 469.

If this command lists indexes that do *not* belong to DB2 Universal Database (that is, the index names begin with something other than **DB2**), other products on your system are using NetQuestion. In this case you cannot remove NetQuestion.

If the list contains any of the DB2 index file names (DB2S71xx or DB2C71xx), DB2 could not unregister the indexes therefore causing the NetQuestion uninstall to fail. This usually occurs if DB2 was incorrectly uninstalled. In this case, you need to manually unregister the indexes one at a time by performing the following steps:

1. Issue one of these commands to ensure that the search server is running:

```
imnss start server    //for SBCS
imqss -start dbcshelp //for DBCS
```

2. Issue one of these commands for each of the index files:

```
imndomap -d index_name //for SBCS
imqdomap -d index_name //for DBCS
```

where *index_name* is the name of one of the indexes returned by the **imndomap -a** or **imqdomap -a** command.

3. Issue one of these commands for each of the index files:

```
imnixdel index_name //for SBCS
imqixdel index_name //for DBCS
```

where *index_name* is the name of one of the indexes returned by the **imndomap -a** or **imqdomap -a** command.

4. Stop the search server:

```
imnss stop server //for SBCS
imqss -stop dbcshelp //for DBCS
```

5. Issue the **imndomap -a** or **imqdomap -a** command to verify that there are no DB2 indexes remaining. If there are, please contact IBM service.
6. Issue one of following commands to verify that no other indexes are active.

```
nqcounti netq_dir //for SBCS
tmcounti netq_dir //for DBCS
```

where *netq_dir* is the fully qualified path under which NetQuestion is installed. See “Locating the NetQuestion Installation Directory” on page 466 if you do not know the directory under which NetQuestion is installed.

If the **nqcounti** or **tmcounti** command return data that indicates one or more indexes are still active, NetQuestion cannot be removed because other IBM products are still registered. Please contact IBM service for assistance.

If the **nqcounti** or **tmcounti** command return data that indicate no indexes are active, go to “Manually Uninstalling NetQuestion on Windows 32-bit operating systems”.

Manually Uninstalling NetQuestion on Windows 32-bit operating systems

If the **nqcounti** or **tmcounti** command does not list any active indices, you can try removing NetQuestion manually by performing the following steps:

1. From a command prompt, enter the **uninstnq** command.
2. Remove the registry entry
\\HKEY_LOCAL_MACHINE\SOFTWARE\IBM\NetQuestion.
3. Remove the NetQuestion installation directory. For example, if you installed NetQuestion on the C:\ drive on Windows NT, remove the **c:\imnq_nt** directory.
4. Remove the environment variables **IMNINST** and **IMNINSTSRV**, and remove the NetQuestion installation path from the **PATH** environment variable.

If you still can not uninstall NetQuestion, call IBM service for assistance.

Rebooting After Uninstalling NetQuestion

After a DB2 Universal Database uninstall, it is very important to reboot before performing another DB2 installation. Some NetQuestion DLLs can be held by the operating system and are not removed until the next reboot. If a NetQuestion install happens before the reboot, the newly installed NetQuestion DLLs will be deleted on the next reboot, rendering NetQuestion unusable.

NetQuestion for OS/2 Operating Systems

This section describes post-installation configuration information, additional troubleshooting tips, and workarounds for known issues with NetQuestion on OS/2 operating systems. It also describes how to remove NetQuestion from your system.

With the release of DB2 Version 7.1, NetQuestion no longer uses TCP/IP for communication between the browser, web server and search server. Instead, NetQuestion uses Named Pipes by default for local communication. However, if you will be using NetQuestion to search DB2 documentation in a client/server environment, you can enable it to use TCP/IP for communication. For more information on enabling NetQuestion to use TCP/IP, see “Switching NetQuestion to use TCP/IP”.

Additional NetQuestion Prerequisites

In addition to meeting the general installation prerequisites for DB2, the following prerequisites are required to install and work with NetQuestion:

- You need a browser, such as Netscape 4.04 for OS/2. If a Netscape browser is not available for your language, use Web Explorer 1.1 or higher. Ensure you turn off proxy handling for **localhost** in the browser you use.
- If you are installing DB2 Universal Database on a system that has VisualAge for C++ for OS/2, you need to have CSD6 or above installed for VisualAge for C++.

Switching NetQuestion to use TCP/IP

NetQuestion in DB2 for OS/2 Version 7.1 uses Named Pipes instead of TCP/IP for local communication. This section describes how to enable NetQuestion to use TCP/IP.

Note: If NetQuestion is already installed on your system for use with another IBM application, such as Visual Age C++ for OS/2, it will likely already be set up to use TCP/IP for communication. In this case, when you install DB2 for OS/2 Version 7.1, the existing version of NetQuestion will be upgraded automatically but it will continue to use TCP/IP for communication so that your other NetQuestion-dependent applications continue to work properly.

To enable NetQuestion to use TCP/IP:

1. Stop the NetQuestion search server by entering the **imnss stop server** (for SBCS) or **imqss -stop dbcshelp** (for DBCS) command.
2. From a command prompt, go to the NetQuestion installation directory. See “Locating the NetQuestion Directory” on page 472 for more information.
3. Replace the current `netq.cfg` file with the one that enables TCP/IP communication by entering the **copy imncxtcp.cfg netq.cfg** command.
4. Start the search server by entering the **imnss start server** (for SBCS) or **imqss -start dbcshelp** (for DBCS) command.

If you want to switch back to communication using named pipes, repeat the above steps replacing `netq.cfg` with `imncxpip.cfg`. You can also switch communication to use unnamed pipes by replacing `netq.cfg` with `imncxloc.cfg`.

The search server will function with or without a network adapter installed as long as TCP/IP local loopback and localhost are enabled on your system. For more information on enabling local loopback and localhost, see “Chapter 8. Control Center Installation and Configuration” on page 115.

To verify your TCP/IP configuration, open the **IBM DB2** folder and double-click the **Start HTML Search Server** icon. If an error message appears, TCP/IP is not configured properly. Follow the instructions in “Chapter 8. Control Center Installation and Configuration” on page 115 to make sure that TCP/IP is correctly configured. Reboot OS/2 if you change any settings.

Changing the Port Number for the Search Server (TCP/IP only)

The search server is assigned to port 49213, a number beyond the public ports assigned for TCP/IP. If you have another product that uses this port, you should change the search server port number by performing the following steps:

1. Edit the `httpd.cnf` file in the search system directory and change the port number to one you know is available, preferably one above 49000.
2. Determine where your `db2path` is by entering the **db2set db2path** command.
3. Go to the `db2path/doc/html` directory and use a text editor to edit the `index.htm` file. In the line with the `<form>` tag, change the value `rlocalhost:49213` to reflect the port number you selected in Step 1.
4. Stop and re-start the search server by clicking on the appropriate icons in the **IBM DB2** folder.

Error Detected When Starting NetQuestion

If you encounter this error:

```
EHS0410 (or EHS0411): An error was detected when starting the search service.  
Stop the service and start it again. Press Enter to continue...
```

your hostname might have changed since doing the install, and you have to record it again. From a command prompt, switch to the NetQuestion installation directory and enter:

```
type netq.cfg
```

to see which hostname NetQuestion is using. If you have changed the hostname, for example while configuring DHCP and DDNS, enter:

```
netqinit data_path
```

where *data_path* is the data directory under the NetQuestion installation directory. For example, if you installed NetQuestion on the D:\ drive, *data_path* would resolve to d:\imnq\data.

Searching while Disconnected from the Network

If, for example, you are using a laptop computer and are temporarily away from a network connection, you must enable localhost to search the documentation. For more information on enabling localhost, see “Configuring TCP/IP on OS/2” on page 123.

Locating the NetQuestion Directory

NetQuestion is installed in its own directory because it may be used by other products. For example, if you installed DB2 Universal Database along with NetQuestion on G:\, and later you installed IBM VisualAge for Java on H:\, the search system is only installed in one place.

Some instructions in this section require you to specify the location of this directory. To locate this directory enter one of the following commands:

```
echo %IMNINSTSRV% //for SBCS  
echo %IMQINSTSRV% //for DBCS
```

You can also run the `x:\sqllib\bin\db2netqd dir` command, where x: is your DB2 installation drive.

Uninstalling NetQuestion on OS/2

Before uninstalling NetQuestion, make sure the search server is stopped by using the appropriate icon in the **IBM DB2** folder.

To uninstall NetQuestion, enter the following command:

```
uninstnq.cmd
```

If NetQuestion is still present after executing this command, continue reading this section to diagnose and fix the problem.

Diagnosing NetQuestion Uninstall Problems

If you executed `uninstnq.cmd` and NetQuestion does not uninstall, the cause may be one of the following:

- TCP/IP was not configured properly. See “Configuring TCP/IP on OS/2” on page 123 to ensure that TCP/IP is configured properly and the search server is initialized with a properly configured TCP/IP setup. Make sure you reboot the system if you make any changes to the configuration.
- DB2 Universal Database or another product is using NetQuestion.

To determine which products are still registered with the search system, issue one of the following commands:

```
imndomap -a           //for SBCS
imqdomap -a           //for DBCS
```

If this command lists no indices, go to “Manually Uninstalling NetQuestion on OS/2” on page 474.

If this command lists indices that do *not* belong to DB2 Universal Database (that is, the index names begin with something other than **DB2**), other products on your system are using NetQuestion. In this case you cannot remove NetQuestion.

If the list contains any of the DB2 index file names (DB2S71xx or DB2C71xx), DB2 could not unregister the indices therefore causing the NetQuestion `uninstnq.cmd` to fail. This usually occurs if DB2 was incorrectly uninstalled. In this case, you need to manually unregister the indices one at a time by performing the following steps:

1. Issue one of these commands to ensure that the search server is running:

```
imnss start server    //for SBCS
imqss -start dbcshelp //for DBCS
```

2. Issue one of these commands for each of the index files:

```
imndomap -d index_name //for SBCS
imqdomap -d index_name //for DBCS
```

where *index_name* is the name of one of the indices returned by the `imndomap -a` or `imqdomap -a` command.

3. Issue one of these commands for each of the index files:

```
imnixdel index_name //for SBCS
imqixdel index_name //for DBCS
```

where *index_name* is the name of one of the indices returned by the `imndomap -a` or `imqdomap -a` command.

4. Stop the search server:

```
imnss stop server           //for SBCS
imqss -stop dbcshelp       //for DBCS
```

5. Issue **imndomap -a** (or **imqdomap -a**) and verify that there are no DB2 indices remaining. If there are, please contact IBM service.
6. Issue the following command to verify that no other indices are active.

```
nqcounti netq_dir //for SBCS
tmcounti netq_dir //for DBCS
```

where *netq_dir* is the fully qualified path under which NetQuestion is installed. See "Locating the NetQuestion Directory" on page 472 if you do not know the directory under which NetQuestion is installed.

If the **nqcounti** or **tmcounti** command returns data that indicate no indices are active, go to "Manually Uninstalling NetQuestion on Windows 32-bit operating systems" on page 469.

If the **nqcounti** or **tmcounti** command returns data that indicate one or more indices are still active, NetQuestion cannot be removed because other products are still registered. Please contact IBM service for assistance.

Manually Uninstalling NetQuestion on OS/2

If the **nqcounti** or **tmcounti** command does not list any active indices, you can try removing NetQuestion manually by performing the following steps:

1. From a command prompt, enter:

```
uninstnq.cmd
```

Ignore this message: "SYS0016: The directory cannot be removed."

2. Remove the NetQuestion directory and all of its subtrees. For example, if you installed NetQuestion on the D:\ drive, remove the d:\imnq directory.
3. Restore CONFIG.SYS from a backup that was created during the installation of DB2.

Note: DB2 install creates numbered backups of CONFIG.SYS in the form CONFIG.*xyz*, where *xyz* is the first available number from 000 to 100. The backups will have a time and date stamp corresponding to the time and date of the install. Two backups are created during any DB2 Universal Database install: one for the DB2 install, and one for the NetQuestion install.

- To remove only NetQuestion settings from CONFIG.*xyz*, restore the second backup (the one with a higher value for *xyz*). Your DB2 settings will remain in the restored file.

- To remove both DB2 Universal Database and NetQuestion settings from CONFIG.xyz, restore the first backup (the one with a lower value for xyz).
4. Reboot your system.

NetQuestion for UNIX Operating Systems

This section describes how to work with NetQuestion on AIX, HP-UX and Solaris operating systems. NetQuestion is not currently supported on other UNIX-based operating systems. This section also describes workarounds to problems that might be encountered while installing NetQuestion on UNIX platforms.

Installing NetQuestion on UNIX Systems

If you installed your DB2 product using the db2setup utility and you selected to install the Product Documentation or Control Center components, NetQuestion was automatically installed and configured for you.

If you installed DB2 manually, you need to perform certain post-installation configuration tasks to set up the online search system. For more information, see your platform's server installation chapter in the *DB2 for UNIX Quick Beginnings* manual.

NetQuestion Installation Workarounds on UNIX Platforms

In some environments, previous installs or uninstalls of NetQuestion make it impossible to install the NetQuestion again. In short, NetQuestion requires that special user ID and group ID names exist to install correctly. Occasionally (usually when NetQuestion has been incompletely removed) one or the other of these IDs does not exist. In other cases, users sometimes do not have the right access needed to create these group IDs and user IDs successfully at installation. Both of these conditions result in NetQuestion failing to install properly.

Problems installing NetQuestion for the first time on the machine

If you experience a problem installing NetQuestion, and you do not have a previous version of the product already installed, then please completely remove NetQuestion and then re-install it. Note that this includes removing the imnadm user ID and group ID before reinstalling.

Problems installing NetQuestion when another NetQuestion component is already installed, or was previously installed

If you experience a problem installing NetQuestion, and you have another version of NetQuestion already installed (or have had it installed in the past),

then do not remove the existing NetQuestion installation. Instead, check to see if there is a group ID called `imnadm`. If there is, then you can either:

- Create the `imnadm` user ID and reinstall the NetQuestion component.
- Delete the `imnadm` group ID and reinstall the NetQuestion component.

Problems installing NetQuestion in an AFS/DCE or NIS/NIS+ environment

In environments where user ID and group ID management is centralized, that is, IDs are created on a server by an authorized administrative user, who then distributes the ID information to client machines, the `imnadm` user IDs and group IDs have to be created prior to users installing NetQuestion. Since these user IDs and group IDs are shared among all products that make use of NetQuestion, this step needs to be performed once prior to installing the first product that uses NetQuestion. After the last product that uses NetQuestion is uninstalled, then the user ID and group ID should be removed.

Appendix C. Setting up DB2 Documentation on a Web Server

This section describes how you can serve the DB2 Universal Database online documentation files from a central machine.

Without a web server, you must store the files on each machine where you want to read the DB2 documentation, or make them available on a network drive.

Considerations for Using a Web Server

You can decide whether to set up a centralized web server for the DB2 documentation, install the documentation on each machine, or use a combination of both:

Advantages of a Web Server

- Any installation and customization of HTML files need only be done once.
- You can use the search facility of the web server, and can create search indices for specific sets of documents that your users need.
- Less storage is required overall. With files stored centrally, client machines only require a web browser.
- Information can be viewed from any kind of client machine that supports a web browser.
- You can supply links to items of key interest to your users, and links to local information such as coding guidelines and operations manuals.
- It is easier to direct people to specific URLs to find the information they need.
- You can collect books for multiple different products, not just those installed locally.

Disadvantages of a Web Server

- It is not easy to use the Information Center from a remote machine to access files on the web server.
- You cannot search the pre-defined DB2 indices from remote clients. You can only search these indices on the machine on which they physically reside..
- An administrator might need to manage network load on the web server.

Working with the DB2 Universal Database Documentation Files in a Client/Server Environment

To set up the DB2 Universal Database documentation on a central server, you should perform the following steps:

1. Install a DB2 server, DB2 client, or DB2 Connect product and select the option to install documentation.
2. As an option, you can customize the HTML files you have installed. The DB2 Information page is named `index.htm` and is located in the product's `doc\html` sub-directory. Depending on the set of products that you install, some of the links in this file might point to non-existent books. You can remove these links if you do not intend to install the books on the central server. You can also add your own links to local resources such as your own operations manual or help desk. The DB2 Information page also contains a **Search** field in which you can enter search terms or phrases.

Note: Keep in mind that DB2 service releases might include an updated version of the DB2 Information page, so be sure to keep a backup copy of any changes you make.

3. As an option, you might want to move or copy the DB2 Universal Database HTML files:
 - To a network drive.
 - To a directory underneath the root directory of a web server.
 - To another machine that acts as a web server.

To copy the files on the same machine, you can use a command that preserves the directory structure, such as `xcopy` or `cp -r`. It is important to preserve the original directory structure, because the book files use relative links to return to the list of books, and the list of books uses relative links to each book.

To copy the files to a different machine, you can use an archiving tool such as `pkzip` or `tar`. Again, use options that preserve the directory structure in the new location.

If you just wanted to transfer a few books, you might use `ftp` to transfer the `index.htm` file and all the files in some of the book subdirectories. The files should be transferred in binary so that the `*.gif` files and any national characters such as accented letters are preserved.

4. As an option on Windows and OS/2 operating systems, you might want to customize the Information Center.

The DB2 Information Center uses a set of mapping files to locate the online information that it displays. By default, it looks for the online HTML information in the `doc\html` sub-directory tree in the directory where DB2 is installed. If you move the HTML files, such as to a web

server or network drive, the Information Center can only find them if the *.ndx files are updated. In the sql11ib\help directory of DB2, look for files with the *.ndx extension. Each one represents a page in the Information Center notebook.

To update the Information Center to find the online information on the x: drive, edit each *.ndx file in turn. Be sure to use an editor that handles Unicode. Change all occurrences of the string file:///DB2PATH%/doc/html to reflect the new location of the HTML files. For example, for a web server you might change the string to http://servername/path/to/db2/docs. For a network drive, you might change the string to file://x:/path/to/db2/docs. It is important to preserve the directory structure wherever the files are located, because the Information Center expects that, for example, the *SQL Reference* files are in a directory named db2s0, and so on.

Typical Web Server Scenarios

Although you can use the basic web server that comes with NetQuestion, for security reasons we *strongly* recommend that you use a "real" web server, such as Apache or Lotus Domino Go!, to serve DB2 documentation to remote clients.

The following sections describe some of the ways that you might put the DB2 documentation on a central server, with the details for particular platforms and web servers. If you use a different platform or web server, you should find it has corresponding features that allow you to achieve similar results.

Scenario 1: Lotus Domino Go! Web Server on OS/2

In this scenario, you are using your OS/2 machine as the DB2 server, and decide to use the Lotus Domino Go! web server to make the DB2 online documentation available to the rest of your intranet.

Install the DB2 Universal Database documentation

You install the DB2 documentation along with the DB2 Enterprise Edition server, on the E: drive. The HTML files are located in the subdirectory e:\sql11ib\doc\html. The server documentation includes such books as the *SQL Reference*, *Command Reference*, *SQL Getting Started*, *Administration Guide*, and the *Quick Beginnings* book for that particular server. Because the Enterprise Server includes the DB2 Connect function and its documentation, you also get the *DB2 Connect User's Guide* and the *DB2 Connect Enterprise Edition Quick Beginnings for OS/2 and Windows NT* manual. Assume that your company does not do any application development, so you do not install the DB2 Personal Developer's Edition, which includes the programming books such as the *Administrative API Reference*.

Install the Web Server

Install the Lotus Domino Go web server from its installation CD-ROM. In this scenario, assume the web server is installed in `c:\www` and that its TCP/IP host name is `udbserv`.

Make the DB2 UDB Documentation Available through the Web Server

To serve files from the DB2 directory, add a line like the following to the `c:\www\httpd.cnf` file:

```
Pass /db2docs/* e:\sql11ib\doc\html\*
```

Make sure that it comes before the line:

```
Pass /* document_root\*
```

Then, restart the web server.

Or, to copy the files to the web server directory, use the `xcopy` command with the `/s` option to copy `e:\sql11ib\doc\html*.*` to the `c:\www\html\db2docs` directory. Your users can now access the DB2 documentation by going to the URL `http://udbserv/db2docs/index.htm`.

Customize (Optional)

Later, you get a phone call from a user who tried to select the *API Reference* and received a "file not found" error because that book was not present on the web server. Another user tried to select the *Quick Beginnings* book for UNIX and got a similar error. You might edit the file `e:\sql11ib\doc\html\index.htm`, removing the links to the programming books, and changing the links to the Windows and UNIX *Quick Beginnings* books to point to a different intranet web server that has those books installed.

Because the company has its own help desk for database problems, you add the help desk's phone number and a `mailto:` link to the `index.htm` file. Keep in mind that corrective service releases might include an updated version of `index.htm`, so be sure to keep a backup copy of any changes you make.

Scenario 2: Netscape Enterprise Web Server on Windows NT

You are a system administrator running Netscape Enterprise server on Windows NT. You are starting to roll out DB2 Universal Database and have not yet decided if you will combine the database server and the web server on a single machine, or use separate servers.

Install the DB2 Universal Database documentation

On this system, DB2 is installed on the E: drive, so the HTML files are all stored under `e:\sql11ib\doc\html`.

Install the Web Server

To test the performance of a combined web and database server, you might install the Netscape Enterprise web server on the same system, named `udbserv` on the company's intranet. The web server is installed on the `H:` drive.

Make the DB2 Documentation Available

Initially, the web server only contains the DB2 documentation, so you want the default home page of `http://udbserv/` to display the file `e:\sql11ib\doc\html\index.htm`, the DB2 Information page. To make the DB2 documentation available:

1. Run the **Administer Netscape Servers** program from the Netscape folder, and choose `udbserv` as the server.
2. Under **Content Mgmt**, change the **Primary Document Directory** to `e:\sql11ib\doc\html`.
3. Use **Document Preferences** to add `index.htm` as one of the default document names so that the browser displays `index.htm` instead of a list of files in the directory.

Later, you decide to store other HTML files on the web server, and configure the server so that the DB2 books are located at `http://udbserv/db2docs/`:

1. Change the **Primary Document Directory** back to a general directory, such as `h:\netscape\server\docs`.
2. Under **Content Mgmt**, select **Additional Document Directories** and map the `db2docs` subdirectory to `e:\sql11ib\doc\html`.
3. From the new default page for the web server, add a link to the `db2docs` directory:

```
<a href="db2docs/index.htm">DB2 Information</a>
```

Transfer Files (Optional)

Eventually, the load on the combined web and database server becomes too great and you decide to transfer the web server to a different machine. You use an archiving tool such as **pkzip** or a Windows version of **tar** to package the directory tree underneath `e:\sql11ib\doc\html`. You restore all the files somewhere on the web server machine, then configure the web server as before to serve the HTML files.

On a different system, you decide to uninstall the DB2 server and keep the HTML files available for the web server. Before uninstalling the server, you use the **xcopy** command to copy the `e:\sql11ib\doc\html` directory tree to `h:\netscape\server\docs\db2docs`, then remove the web server mapping for the additional document directory named `db2docs`.

Scenario 3: Microsoft Internet Information Server on Windows NT

In this scenario, you are using Windows NT 4.0 as the DB2 server, and decide to use Internet Information Server (IIS) to make the DB2 online documentation available to the rest of your workgroup. IIS is available with Windows NT 4.0 Server, but it also runs on Windows NT 4.0 Workstation.

You can also set up Peer Web Services (PWS), which is available with Windows NT 4.0 Workstation but can be installed on Windows NT 4.0 Server, using the following instructions.

Install the DB2 Universal Database Documentation

Install the DB2 documentation into the same directory as in the previous scenarios: `e:\sql11ib\doc\html`.

Install the Web Server

If IIS is not installed at the same time as the Windows NT system, install it from your Windows NT installation CD-ROM. Open the **Services** dialog in the **Control Panel** to start this service automatically.

Make the DB2 Documentation Available through the Web Server

Run the **Internet Service Manager** from the Microsoft Internet Information Server folder of the **Start** menu, going to the **WWW** section and then the **Directories** subsection. Edit the home directory entry to change the path from `c:\inetpub\wwwroot` to `e:\sql11ib\doc\html` and the default document to `index.htm`. Later, if you need to serve other documents as well as the DB2 books, change the home document back to `c:\inetpub\wwwroot` and add a new virtual directory named `db2docs`.

Customize (Optional)

If you want to use the built-in search facility of IIS, you must replace the file `e:\sql11ib\doc\html\index.htm` with the file `inetpub\samples\isapi\srch.htm`, and copy the DB2 HTML files as explained below.

Copy Files (Optional)

To use the built-in search facility of IIS, copy the DB2 HTML files to the `c:\inetpub\wwwroot` directory tree. Create a directory named `db2docs`, and use the `xcopy /s` command to preserve the directory structure during the copy. Remove the virtual directory `db2docs` so that the files under the real `db2docs` directory will be used.

Serving Documentation in Multiple Languages

To serve translated versions of the DB2 online documentation on the web, you might prefer to start with a UNIX server, where you can use symbolic links instead of making multiple copies of files and directories.

When you install DB2 on a UNIX server, you can select one or more languages for the online product library. English (en_US) HTML books are always installed with books in any other language, because some books are available only in English.

The installation through the `db2setup` utility creates a `doc/html` subdirectory that contains a link to each translated book or individual HTML file, or the English one if no translated version is available.

The `doc/html` subdirectory can be found in the following paths on UNIX systems:

- On AIX systems: `/usr/lpp/db2_07_01/doc/html`
- On HP-UX and Solaris systems: `/opt/IBMdb2/V7.1/doc/html`

This provides a well-known location for the DB2 documentation, regardless of the language chosen.

If you install through the `smit` or another native operating system installation tool instead of the `db2setup` utility, you must follow the instructions in the *DB2 for UNIX Quick Beginnings* manual for running the `db2insthtml` command. You must run this command for each language in which you installed DB2 documentation.

For example, if you used `smit` on AIX to install English, German, and French documentation, you would run these commands afterward:

```
cd /usr/lpp/db2_07_01/doc
db2insthtml en_US
db2insthtml de_DE
db2insthtml fr_FR
```

Specifying the `fr_FR` locale last means that the links in `/usr/lpp/db2_07_01/doc/html` would point to the French versions of files that are translated into French, and English files for everything else. When running the `db2insthtml` command for multiple languages, always run it for `en_US` first.

Example 1: English and Japanese

You want to set up a web server for users who speak only Japanese, or a mixture of Japanese and English. Because only two languages are involved, you can select both the English and Japanese languages for the Product Library and set up a virtual directory on the server that points to the `doc/html` subdirectory, which has links to all the Japanese books, and to English books where no translated version is available. The `index.htm` file in this directory provides a list of all the product manuals, and is the place to begin when looking for information.

Example 2: English, Japanese and Swedish

Now, you want to set up a web server for users who speak a mixture of Japanese, English, and Swedish. The Japanese books are in the `doc/ja_JP/html` subdirectory, the English books are in the `doc/en_US/html` subdirectory, and the Swedish books are in the `doc/sv_SE/html` subdirectory. The Japanese and Swedish directories have symbolic links so that any books that are not translated are displayed in English.

Serving Documentation for Multiple Platforms

If your users have a combination of OS/2, Windows NT, and UNIX workstations, you might want to collect the books for several platforms on a single web server. Again, installing DB2 on a UNIX-based server is the simplest starting point, because it comes with more books than do the DB2 for OS/2 or DB2 for Windows NT server products

Each DB2 platform or product comes with its own *Quick Beginnings* book. To gather all of these books, you must get them from the various products. You can find these books in the `doc/html` subdirectories of the product CD-ROMs, so that you can get them without actually installing the servers.

Appendix D. Using the DB2 Library

The DB2 Universal Database library consists of online help, books (PDF and HTML), and sample programs in HTML format. This section describes the information that is provided, and how you can access it.

To access product information online, you can use the Information Center. For more information, see “Accessing Information with the Information Center” on page 501. You can view task information, DB2 books, troubleshooting information, sample programs, and DB2 information on the Web.

DB2 PDF Files and Printed Books

DB2 Information

The following table divides the DB2 books into four categories:

DB2 Guide and Reference Information

These books contain the common DB2 information for all platforms.

DB2 Installation and Configuration Information

These books are for DB2 on a specific platform. For example, there are separate *Quick Beginnings* books for DB2 on OS/2, Windows, and UNIX-based platforms.

Cross-platform sample programs in HTML

These samples are the HTML version of the sample programs that are installed with the Application Development Client. The samples are for informational purposes and do not replace the actual programs.

Release notes

These files contain late-breaking information that could not be included in the DB2 books.

The installation manuals, release notes, and tutorials are viewable in HTML directly from the product CD-ROM. Most books are available in HTML on the product CD-ROM for viewing and in Adobe Acrobat (PDF) format on the DB2 publications CD-ROM for viewing and printing. You can also order a printed copy from IBM; see “Ordering the Printed Books” on page 497. The following table lists books that can be ordered.

On OS/2 and Windows platforms, you can install the HTML files under the `sql1lib\doc\html` directory. DB2 information is translated into different

languages; however, all the information is not translated into every language. Whenever information is not available in a specific language, the English information is provided

On UNIX platforms, you can install multiple language versions of the HTML files under the `doc/%L/html` directories, where `%L` represents the locale. For more information, refer to the appropriate *Quick Beginnings* book.

You can obtain DB2 books and access information in a variety of ways:

- “Viewing Information Online” on page 500
- “Searching Information Online” on page 504
- “Ordering the Printed Books” on page 497
- “Printing the PDF Books” on page 496

Table 37. DB2 Information

Name	Description	Form Number PDF File Name	HTML Directory
DB2 Guide and Reference Information			
<i>Administration Guide</i>	<i>Administration Guide: Planning</i> provides an overview of database concepts, information about design issues (such as logical and physical database design), and a discussion of high availability.	SC09-2946 db2d1x70	db2d0
	<i>Administration Guide: Implementation</i> provides information on implementation issues such as implementing your design, accessing databases, auditing, backup and recovery.	SC09-2944 db2d2x70	
	<i>Administration Guide: Performance</i> provides information on database environment and application performance evaluation and tuning.	SC09-2945 db2d3x70	
	You can order the three volumes of the <i>Administration Guide</i> in the English language in North America using the form number SBOF-8934.		
<i>Administrative API Reference</i>	Describes the DB2 application programming interfaces (APIs) and data structures that you can use to manage your databases. This book also explains how to call APIs from your applications.	SC09-2947 db2b0x70	db2b0

Table 37. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Application Building Guide</i>	Provides environment setup information and step-by-step instructions about how to compile, link, and run DB2 applications on Windows, OS/2, and UNIX-based platforms.	SC09-2948 db2axx70	db2ax
<i>APPC, CPI-C, and SNA Sense Codes</i>	Provides general information about APPC, CPI-C, and SNA sense codes that you may encounter when using DB2 Universal Database products.	No form number db2apx70	db2ap
	Available in HTML format only.		
<i>Application Development Guide</i>	Explains how to develop applications that access DB2 databases using embedded SQL or Java (JDBC and SQLJ). Discussion topics include writing stored procedures, writing user-defined functions, creating user-defined types, using triggers, and developing applications in partitioned environments or with federated systems.	SC09-2949 db2a0x70	db2a0
<i>CLI Guide and Reference</i>	Explains how to develop applications that access DB2 databases using the DB2 Call Level Interface, a callable SQL interface that is compatible with the Microsoft ODBC specification.	SC09-2950 db2l0x70	db2l0
<i>Command Reference</i>	Explains how to use the Command Line Processor and describes the DB2 commands that you can use to manage your database.	SC09-2951 db2n0x70	db2n0
<i>Connectivity Supplement</i>	Provides setup and reference information on how to use DB2 for AS/400, DB2 for OS/390, DB2 for MVS, or DB2 for VM as DRDA application requesters with DB2 Universal Database servers. This book also details how to use DRDA application servers with DB2 Connect application requesters.	No form number db2h1x70	db2h1
	Available in HTML and PDF only.		

Table 37. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Data Movement Utilities Guide and Reference</i>	Explains how to use DB2 utilities, such as import, export, load, AutoLoader, and DPROP, that facilitate the movement of data.	SC09-2955 db2dmx70	db2dm
<i>Data Warehouse Center Administration Guide</i>	Provides information on how to build and maintain a data warehouse using the Data Warehouse Center.	SC26-9993 db2ddx70	db2dd
<i>Data Warehouse Center Application Integration Guide</i>	Provides information to help programmers integrate applications with the Data Warehouse Center and with the Information Catalog Manager.	SC26-9994 db2adx70	db2ad
<i>DB2 Connect User's Guide</i>	Provides concepts, programming, and general usage information for the DB2 Connect products.	SC09-2954 db2c0x70	db2c0
<i>DB2 Query Patroller Administration Guide</i>	Provides an operational overview of the DB2 Query Patroller system, specific operational and administrative information, and task information for the administrative graphical user interface utilities.	SC09-2958 db2dwx70	db2dw
<i>DB2 Query Patroller User's Guide</i>	Describes how to use the tools and functions of the DB2 Query Patroller.	SC09-2960 db2wwx70	db2ww
<i>Glossary</i>	Provides definitions for terms used in DB2 and its components. Available in HTML format and in the <i>SQL Reference</i> .	No form number db2t0x70	db2t0
<i>Image, Audio, and Video Extenders Administration and Programming</i>	Provides general information about DB2 extenders, and information on the administration and configuration of the image, audio, and video (IAV) extenders and on programming using the IAV extenders. It includes reference information, diagnostic information (with messages), and samples.	SC26-9929 dmbu7x70	dmbu7
<i>Information Catalog Manager Administration Guide</i>	Provides guidance on managing information catalogs.	SC26-9995 db2dix70	db2di

Table 37. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Information Catalog Manager Programming Guide and Reference</i>	Provides definitions for the architected interfaces for the Information Catalog Manager.	SC26-9997 db2bix70	db2bi
<i>Information Catalog Manager User's Guide</i>	Provides information on using the Information Catalog Manager user interface.	SC26-9996 db2aix70	db2ai
<i>Installation and Configuration Supplement</i>	Guides you through the planning, installation, and setup of platform-specific DB2 clients. This supplement also contains information on binding, setting up client and server communications, DB2 GUI tools, DRDA AS, distributed installation, the configuration of distributed requests, and accessing heterogeneous data sources.	GC09-2957 db2iyx70	db2iy
<i>Message Reference</i>	Lists messages and codes issued by DB2, the Information Catalog Manager, and the Data Warehouse Center, and describes the actions you should take. You can order both volumes of the Message Reference in the English language in North America with the form number SBOF-8932.	Volume 1 GC09-2978 db2m1x70 Volume 2 GC09-2979 db2m2x70	db2m0
<i>OLAP Integration Server Administration Guide</i>	Explains how to use the Administration Manager component of the OLAP Integration Server.	SC27-0787 db2dpx70	n/a
<i>OLAP Integration Server Metaoutline User's Guide</i>	Explains how to create and populate OLAP metaoutlines using the standard OLAP Metaoutline interface (not by using the Metaoutline Assistant).	SC27-0784 db2upx70	n/a
<i>OLAP Integration Server Model User's Guide</i>	Explains how to create OLAP models using the standard OLAP Model Interface (not by using the Model Assistant).	SC27-0783 db2lpx70	n/a
<i>OLAP Setup and User's Guide</i>	Provides configuration and setup information for the OLAP Starter Kit.	SC27-0702 db2ipx70	db2ip
<i>OLAP Spreadsheet Add-in User's Guide for Excel</i>	Describes how to use the Excel spreadsheet program to analyze OLAP data.	SC27-0786 db2epx70	db2ep

Table 37. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>OLAP Spreadsheet Add-in User's Guide for Lotus 1-2-3</i>	Describes how to use the Lotus 1-2-3 spreadsheet program to analyze OLAP data.	SC27-0785 db2tpx70	db2tp
<i>Replication Guide and Reference</i>	Provides planning, configuration, administration, and usage information for the IBM Replication tools supplied with DB2.	SC26-9920 db2e0x70	db2e0
<i>Spatial Extender User's Guide and Reference</i>	Provides information about installing, configuring, administering, programming, and troubleshooting the Spatial Extender. Also provides significant descriptions of spatial data concepts and provides reference information (messages and SQL) specific to the Spatial Extender.	SC27-0701 db2sbx70	db2sb
<i>SQL Getting Started</i>	Introduces SQL concepts and provides examples for many constructs and tasks.	SC09-2973 db2y0x70	db2y0
<i>SQL Reference, Volume 1 and Volume 2</i>	Describes SQL syntax, semantics, and the rules of the language. This book also includes information about release-to-release incompatibilities, product limits, and catalog views. You can order both volumes of the <i>SQL Reference</i> in the English language in North America with the form number SBOF-8933.	Volume 1 SC09-2974 db2s1x70 Volume 2 SC09-2975 db2s2x70	db2s0
<i>System Monitor Guide and Reference</i>	Describes how to collect different kinds of information about databases and the database manager. This book explains how to use the information to understand database activity, improve performance, and determine the cause of problems.	SC09-2956 db2f0x70	db2f0
<i>Text Extender Administration and Programming</i>	Provides general information about DB2 extenders and information on the administration and configuring of the text extender and on programming using the text extenders. It includes reference information, diagnostic information (with messages) and samples.	SC26-9930 desu9x70	desu9

Table 37. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Troubleshooting Guide</i>	Helps you determine the source of errors, recover from problems, and use diagnostic tools in consultation with DB2 Customer Service.	GC09-2850 db2p0x70	db2p0
<i>What's New</i>	Describes the new features, functions, and enhancements in DB2 Universal Database, Version 7.	SC09-2976 db2q0x70	db2q0
DB2 Installation and Configuration Information			
<i>DB2 Connect Enterprise Edition for OS/2 and Windows Quick Beginnings</i>	Provides planning, migration, installation, and configuration information for DB2 Connect Enterprise Edition on the OS/2 and Windows 32-bit operating systems. This book also contains installation and setup information for many supported clients.	GC09-2953 db2c6x70	db2c6
<i>DB2 Connect Enterprise Edition for UNIX Quick Beginnings</i>	Provides planning, migration, installation, configuration, and task information for DB2 Connect Enterprise Edition on UNIX-based platforms. This book also contains installation and setup information for many supported clients.	GC09-2952 db2cyx70	db2cy
<i>DB2 Connect Personal Edition Quick Beginnings</i>	Provides planning, migration, installation, configuration, and task information for DB2 Connect Personal Edition on the OS/2 and Windows 32-bit operating systems. This book also contains installation and setup information for all supported clients.	GC09-2967 db2c1x70	db2c1
<i>DB2 Connect Personal Edition Quick Beginnings for Linux</i>	Provides planning, installation, migration, and configuration information for DB2 Connect Personal Edition on all supported Linux distributions.	GC09-2962 db2c4x70	db2c4
<i>DB2 Data Links Manager Quick Beginnings</i>	Provides planning, installation, configuration, and task information for DB2 Data Links Manager for AIX and Windows 32-bit operating systems.	GC09-2966 db2z6x70	db2z6

Table 37. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>DB2 Enterprise - Extended Edition for UNIX Quick Beginnings</i>	Provides planning, installation, and configuration information for DB2 Enterprise - Extended Edition on UNIX-based platforms. This book also contains installation and setup information for many supported clients.	GC09-2964 db2v3x70	db2v3
<i>DB2 Enterprise - Extended Edition for Windows Quick Beginnings</i>	Provides planning, installation, and configuration information for DB2 Enterprise - Extended Edition for Windows 32-bit operating systems. This book also contains installation and setup information for many supported clients.	GC09-2963 db2v6x70	db2v6
<i>DB2 for OS/2 Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database on the OS/2 operating system. This book also contains installation and setup information for many supported clients.	GC09-2968 db2i2x70	db2i2
<i>DB2 for UNIX Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database on UNIX-based platforms. This book also contains installation and setup information for many supported clients.	GC09-2970 db2ixx70	db2ix
<i>DB2 for Windows Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database on Windows 32-bit operating systems. This book also contains installation and setup information for many supported clients.	GC09-2971 db2i6x70	db2i6
<i>DB2 Personal Edition Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database Personal Edition on the OS/2 and Windows 32-bit operating systems.	GC09-2969 db2i1x70	db2i1
<i>DB2 Personal Edition Quick Beginnings for Linux</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database Personal Edition on all supported Linux distributions.	GC09-2972 db2i4x70	db2i4

Table 37. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>DB2 Query Patroller Installation Guide</i>	Provides installation information about DB2 Query Patroller.	GC09-2959 db2iwx70	db2iw
<i>DB2 Warehouse Manager Installation Guide</i>	Provides installation information for warehouse agents, warehouse transformers, and the Information Catalog Manager.	GC26-9998 db2idx70	db2id
Cross-Platform Sample Programs in HTML			
Sample programs in HTML	Provides the sample programs in HTML format for the programming languages on all platforms supported by DB2. The sample programs are provided for informational purposes only. Not all samples are available in all programming languages. The HTML samples are only available when the DB2 Application Development Client is installed. For more information on the programs, refer to the <i>Application Building Guide</i> .	No form number	db2hs
Release Notes			
<i>DB2 Connect Release Notes</i>	Provides late-breaking information that could not be included in the DB2 Connect books.	See note #2.	db2cr
<i>DB2 Installation Notes</i>	Provides late-breaking installation-specific information that could not be included in the DB2 books.	Available on product CD-ROM only.	
<i>DB2 Release Notes</i>	Provides late-breaking information about all DB2 products and features that could not be included in the DB2 books.	See note #2.	db2ir

Notes:

1. The character *x* in the sixth position of the file name indicates the language version of a book. For example, the file name *db2d0e70* identifies the English version of the *Administration Guide* and the file name *db2d0f70* identifies the French version of the same book. The following letters are used in the sixth position of the file name to indicate the language version:

Language	Identifier
Brazilian Portuguese	b

Bulgarian	u
Czech	x
Danish	d
Dutch	q
English	e
Finnish	y
French	f
German	g
Greek	a
Hungarian	h
Italian	i
Japanese	j
Korean	k
Norwegian	n
Polish	p
Portuguese	v
Russian	r
Simp. Chinese	c
Slovenian	l
Spanish	z
Swedish	s
Trad. Chinese	t
Turkish	m

2. Late breaking information that could not be included in the DB2 books is available in the Release Notes in HTML format and as an ASCII file. The HTML version is available from the Information Center and on the product CD-ROMs. To view the ASCII file:
 - On UNIX-based platforms, see the `Release.Notes` file. This file is located in the `DB2DIR/Readme/%L` directory, where `%L` represents the locale name and `DB2DIR` represents:
 - `/usr/lpp/db2_07_01` on AIX
 - `/opt/IBMDB2/V7.1` on HP-UX, PTX, Solaris, and Silicon Graphics IRIX
 - `/usr/IBMDB2/V7.1` on Linux.
 - On other platforms, see the `RELEASE.TXT` file. This file is located in the directory where the product is installed. On OS/2 platforms, you can also double-click the **IBM DB2** folder and then double-click the **Release Notes** icon.

Printing the PDF Books

If you prefer to have printed copies of the books, you can print the PDF files found on the DB2 publications CD-ROM. Using the Adobe Acrobat Reader, you can print either the entire book or a specific range of pages. For the file name of each book in the library, see Table 37 on page 488.

You can obtain the latest version of the Adobe Acrobat Reader from the Adobe Web site at <http://www.adobe.com>.

The PDF files are included on the DB2 publications CD-ROM with a file extension of PDF. To access the PDF files:

1. Insert the DB2 publications CD-ROM. On UNIX-based platforms, mount the DB2 publications CD-ROM. Refer to your *Quick Beginnings* book for the mounting procedures.
2. Start the Acrobat Reader.
3. Open the desired PDF file from one of the following locations:
 - On OS/2 and Windows platforms:
x:\doc\language directory, where *x* represents the CD-ROM drive and *language* represent the two-character country code that represents your language (for example, EN for English).
 - On UNIX-based platforms:
/cdrom/doc/%L directory on the CD-ROM, where */cdrom* represents the mount point of the CD-ROM and *%L* represents the name of the desired locale.

You can also copy the PDF files from the CD-ROM to a local or network drive and read them from there.

Ordering the Printed Books

You can order the printed DB2 books either individually or as a set (in North America only) by using a sold bill of forms (SBOF) number. To order books, contact your IBM authorized dealer or marketing representative, or phone 1-800-879-2755 in the United States or 1-800-IBM-4Y0U in Canada. You can also order the books from the Publications Web page at <http://www.elink.ibm.com/pbl/pbl>.

Two sets of books are available. SBOF-8935 provides reference and usage information for the DB2 Warehouse Manager. SBOF-8931 provides reference and usage information for all other DB2 Universal Database products and features. The contents of each SBOF are listed in the following table:

Table 38. Ordering the printed books

SBOF Number	Books Included
SBOF-8931	<ul style="list-style-type: none"> • Administration Guide: Planning • Administration Guide: Implementation • Administration Guide: Performance • Administrative API Reference • Application Building Guide • Application Development Guide • CLI Guide and Reference • Command Reference • Data Movement Utilities Guide and Reference • Data Warehouse Center Administration Guide • Data Warehouse Center Application Integration Guide • DB2 Connect User's Guide • Installation and Configuration Supplement • Image, Audio, and Video Extenders Administration and Programming • Message Reference, Volumes 1 and 2 • OLAP Integration Server Administration Guide • OLAP Integration Server Metaoutline User's Guide • OLAP Integration Server Model User's Guide • OLAP Integration Server User's Guide • OLAP Setup and User's Guide • OLAP Spreadsheet Add-in User's Guide for Excel • OLAP Spreadsheet Add-in User's Guide for Lotus 1-2-3 • Replication Guide and Reference • Spatial Extender Administration and Programming Guide • SQL Getting Started • SQL Reference, Volumes 1 and 2 • System Monitor Guide and Reference • Text Extender Administration and Programming • Troubleshooting Guide • What's New
SBOF-8935	<ul style="list-style-type: none"> • Information Catalog Manager Administration Guide • Information Catalog Manager User's Guide • Information Catalog Manager Programming Guide and Reference • Query Patroller Administration Guide • Query Patroller User's Guide

DB2 Online Documentation

Accessing Online Help

Online help is available with all DB2 components. The following table describes the various types of help.

Type of Help	Contents	How to Access...
<i>Command Help</i>	Explains the syntax of commands in the command line processor.	<p>From the command line processor in interactive mode, enter:</p> <p style="padding-left: 40px;"><i>? command</i></p> <p>where <i>command</i> represents a keyword or the entire command.</p> <p>For example, <i>? catalog</i> displays help for all the CATALOG commands, while <i>? catalog database</i> displays help for the CATALOG DATABASE command.</p>
<i>Client Configuration Assistant Help</i>	Explains the tasks you can perform in a window or notebook. The help includes overview and prerequisite information you need to know, and it describes how to use the window or notebook controls.	From a window or notebook, click the Help push button or press the F1 key.
<i>Command Center Help</i>		
<i>Control Center Help</i>		
<i>Data Warehouse Center Help</i>		
<i>Event Analyzer Help</i>		
<i>Information Catalog Manager Help</i>		
<i>Satellite Administration Center Help</i>		
<i>Script Center Help</i>	<p>From the command line processor in interactive mode, enter:</p> <p style="padding-left: 40px;"><i>? XXXnnnnn</i></p> <p>where <i>XXXnnnnn</i> represents a valid message identifier.</p> <p>For example, <i>? SQL30081</i> displays help about the SQL30081 message.</p> <p>To view message help one screen at a time, enter:</p> <p style="padding-left: 40px;"><i>? XXXnnnnn more</i></p> <p>To save message help in a file, enter:</p> <p style="padding-left: 40px;"><i>? XXXnnnnn > filename.ext</i></p> <p>where <i>filename.ext</i> represents the file where you want to save the message help.</p>	

Type of Help	Contents	How to Access...
<i>SQL Help</i>	Explains the syntax of SQL statements.	<p>From the command line processor in interactive mode, enter:</p> <pre>help <i>statement</i></pre> <p>where <i>statement</i> represents an SQL statement.</p> <p>For example, help SELECT displays help about the SELECT statement.</p> <p>Note: SQL help is not available on UNIX-based platforms.</p>
<i>SQLSTATE Help</i>	Explains SQL states and class codes.	<p>From the command line processor in interactive mode, enter:</p> <pre>? <i>sqlstate</i> or ? <i>class code</i></pre> <p>where <i>sqlstate</i> represents a valid five-digit SQL state and <i>class code</i> represents the first two digits of the SQL state.</p> <p>For example, ? 08003 displays help for the 08003 SQL state, while ? 08 displays help for the 08 class code.</p>

Viewing Information Online

The books included with this product are in Hypertext Markup Language (HTML) softcopy format. Softcopy format enables you to search or browse the information and provides hypertext links to related information. It also makes it easier to share the library across your site.

You can view the online books or sample programs with any browser that conforms to HTML Version 3.2 specifications.

To view online books or sample programs:

- If you are running DB2 administration tools, use the Information Center.
- From a browser, click **File** —> **Open Page**. The page you open contains descriptions of and links to DB2 information:
 - On UNIX-based platforms, open the following page:

```
INSTHOME/sql1lib/doc/%L/html/index.htm
```

where %L represents the locale name.

- On other platforms, open the following page:

```
sql1lib\doc\html\index.htm
```

The path is located on the drive where DB2 is installed.

If you have not installed the Information Center, you can open the page by double-clicking the **DB2 Information** icon. Depending on the system you are using, the icon is in the main product folder or the Windows Start menu.

Installing the Netscape Browser

If you do not already have a Web browser installed, you can install Netscape from the Netscape CD-ROM found in the product boxes. For detailed instructions on how to install it, perform the following:

1. Insert the Netscape CD-ROM.
2. On UNIX-based platforms only, mount the CD-ROM. Refer to your *Quick Beginnings* book for the mounting procedures.
3. For installation instructions, refer to the `CDNAVnn.txt` file, where *nn* represents your two character language identifier. The file is located at the root directory of the CD-ROM.

Accessing Information with the Information Center

The Information Center provides quick access to DB2 product information. The Information Center is available on all platforms on which the DB2 administration tools are available.

You can open the Information Center by double-clicking the Information Center icon. Depending on the system you are using, the icon is in the Information folder in the main product folder or the Windows **Start** menu.

You can also access the Information Center by using the toolbar and the **Help** menu on the DB2 Windows platform.

The Information Center provides six types of information. Click the appropriate tab to look at the topics provided for that type.

Tasks	Key tasks you can perform using DB2.
Reference	DB2 reference information, such as keywords, commands, and APIs.
Books	DB2 books.
Troubleshooting	Categories of error messages and their recovery actions.
Sample Programs	Sample programs that come with the DB2 Application Development Client. If you did not install the DB2 Application Development Client, this tab is not displayed.
Web	DB2 information on the World Wide Web. To access this information, you must have a connection to the Web from your system.

When you select an item in one of the lists, the Information Center launches a viewer to display the information. The viewer might be the system help viewer, an editor, or a Web browser, depending on the kind of information you select.

The Information Center provides a find feature, so you can look for a specific topic without browsing the lists.

For a full text search, follow the hypertext link in the Information Center to the **Search DB2 Online Information** search form.

The HTML search server is usually started automatically. If a search in the HTML information does not work, you may have to start the search server using one of the following methods:

On Windows

Click **Start** and select **Programs** → **IBM DB2** → **Information** → **Start HTML Search Server**.

On OS/2

Double-click the **DB2 for OS/2** folder, and then double-click the **Start HTML Search Server** icon.

Refer to the release notes if you experience any other problems when searching the HTML information.

Note: The Search function is not available in the Linux, PTX, and Silicon Graphics IRIX environments.

Using DB2 Wizards

Wizards help you complete specific administration tasks by taking you through each task one step at a time. Wizards are available through the Control Center and the Client Configuration Assistant. The following table lists the wizards and describes their purpose.

Note: The Create Database, Create Index, Configure Multisite Update, and Performance Configuration wizards are available for the partitioned database environment.

Wizard	Helps You to...	How to Access...
<i>Add Database</i>	Catalog a database on a client workstation.	From the Client Configuration Assistant, click Add .
<i>Backup Database</i>	Determine, create, and schedule a backup plan.	From the Control Center, right-click the database you want to back up and select Backup → Database Using Wizard .

Wizard	Helps You to...	How to Access...
<i>Configure Multisite Update</i>	Configure a multisite update, a distributed transaction, or a two-phase commit.	From the Control Center, right-click the Databases folder and select Multisite Update .
<i>Create Database</i>	Create a database, and perform some basic configuration tasks.	From the Control Center, right-click the Databases folder and select Create —> Database Using Wizard .
<i>Create Table</i>	Select basic data types, and create a primary key for the table.	From the Control Center, right-click the Tables icon and select Create —> Table Using Wizard .
<i>Create Table Space</i>	Create a new table space.	From the Control Center, right-click the Table Spaces icon and select Create —> Table Space Using Wizard .
<i>Create Index</i>	Advise which indexes to create and drop for all your queries.	From the Control Center, right-click the Index icon and select Create —> Index Using Wizard .
<i>Performance Configuration</i>	Tune the performance of a database by updating configuration parameters to match your business requirements.	From the Control Center, right-click the database you want to tune and select Configure Performance Using Wizard . For the partitioned database environment, from the Database Partitions view, right-click the first database partition you want to tune and select Configure Performance Using Wizard .
<i>Restore Database</i>	Recover a database after a failure. It helps you understand which backup to use, and which logs to replay.	From the Control Center, right-click the database you want to restore and select Restore —> Database Using Wizard .

Setting Up a Document Server

By default, the DB2 information is installed on your local system. This means that each person who needs access to the DB2 information must install the same files. To have the DB2 information stored in a single location, perform the following steps:

1. Copy all files and subdirectories from `\sql11ib\doc\html` on your local system to a Web server. Each book has its own subdirectory that contains all the necessary HTML and GIF files that make up the book. Ensure that the directory structure remains the same.

2. Configure the Web server to look for the files in the new location. For information, refer to the NetQuestion Appendix in the *Installation and Configuration Supplement*.
3. If you are using the Java version of the Information Center, you can specify a base URL for all HTML files. You should use the URL for the list of books.
4. When you are able to view the book files, you can bookmark commonly viewed topics. You will probably want to bookmark the following pages:
 - List of books
 - Tables of contents of frequently used books
 - Frequently referenced articles, such as the ALTER TABLE topic
 - The Search form

For information about how you can serve the DB2 Universal Database online documentation files from a central machine, refer to the NetQuestion Appendix in the *Installation and Configuration Supplement*.

Searching Information Online

To find information in the HTML files, use one of the following methods:

- Click **Search** in the top frame. Use the search form to find a specific topic. This function is not available in the Linux, PTX, or Silicon Graphics IRIX environments.
- Click **Index** in the top frame. Use the index to find a specific topic in the book.
- Display the table of contents or index of the help or the HTML book, and then use the find function of the Web browser to find a specific topic in the book.
- Use the bookmark function of the Web browser to quickly return to a specific topic.
- Use the search function of the Information Center to find specific topics. See “Accessing Information with the Information Center” on page 501 for details.

Appendix E. National Language Support (NLS)

This section contains information about the National Language Support (NLS) provided by DB2, including information about supported languages and code pages on OS/2 and Windows operating environments, and supported locales and code sets on UNIX operating systems. For information on developing applications that use NLS, refer to the *Application Development Guide*.

Language and Codeset Support for UNIX Operating Systems

DB2 supports many code sets and locales without translating the messages for the corresponding languages. Supporting a locale means that you can create and use a database in that locale, but you may have to view all panels and messages in a different language, if translated messages are not available in DB2. For a complete list of locales supported, refer to the *Administration Guide*.

If you want to operate in a different language environment, perform the following steps:

Step 1. Ensure that the appropriate message option for the desired language has been installed.

Step 2. Set the *LANG* environment variable to the desired locale.

For example, to use fr_FR messages on DB2 for AIX, you must have the fr_FR message option installed and must set *LANG* to fr_FR.

The selected message catalog filesets are placed in the following directories on the target workstation:

DB2 for AIX

/usr/lpp/db2_07_01/msg/%L

DB2 for HP-UX, PTX, and Solaris

/opt/IBMDB2/V7.1/msg/%L

DB2 for Linux

/usr/IBMDB2/V7.1/msg/%L

where %L is equal to the locale name of the message catalog.

Code Page and Language Support for OS/2 and Windows Operating Environments

During installation of DB2, the country, codepage, and regional settings are established. However, you can change these settings after installing DB2: including regional settings such as code page, country language (for monetary, date, and numeric formatting), and time zone. When a new connection to a database is made, the database manager uses these new values.

You must ensure that your regional settings are set correctly. DB2 may not produce the expected results if the country, code page, or regional settings are incorrect for the intended language. Table 39 shows the languages into which the DB2 messages are translated. If the setup is run on a machine that is set up using a non-supported language, then English is the default unless the user has specified otherwise.

Table 39. Languages and Code Pages

Country Code	Language
bg	Bulgarian
br	Brazilian Portuguese
cn	Simplified Chinese (PRC)
cz	Czech
de	German
dk	Danish
en	English
es	Spanish
fi	Finnish
fr	French
gr	Greek
hu	Hungarian
il	Hebrew
it	Italian
jp	Japanese
kr	Korean
nl	Dutch
no	Norwegian
pl	Polish
pt	Portuguese
ru	Russian

Table 39. Languages and Code Pages (continued)

Country Code	Language
se	Swedish
si	Slovenian
tr	Turkish
tw	Traditional Chinese (Taiwan)

Appendix F. Naming Rules



Go to the section that describes the naming rules that you require information on:

- “General Naming Rules”
 - “Database, Database Alias, and Catalog Node Name Rules”
 - “Object Name Rules” on page 510
 - “Username, User ID, Group Name, and Instance Name Rules” on page 511
 - “Workstation Name (rname) Rules” on page 512
 - “DB2SYSTEM Naming Rules” on page 512
 - “Password Rules” on page 512
-

General Naming Rules

Unless otherwise specified, all names can include the following characters:

- A through Z. When used in most names, characters A through Z are converted from lowercase to uppercase.
- 0 through 9
- @, #, \$, and _ (underscore)

Unless otherwise specified, all names must begin with one of the following characters:

- A through Z
- @, #, and \$

Do not use SQL reserved words to name tables, views, columns, indexes, or authorization IDs. For a list of SQL reserved words, refer to *SQL Reference*.

Database, Database Alias, and Catalog Node Name Rules

Database names are the identifying names assigned to databases in the database manager. *Database alias names* are synonyms given to remote databases. Database aliases must be unique within the System Database Directory in which all aliases are stored. *Catalog node names* are the identifying names that are assigned to entries in the node directory. Each entry in the node directory is an alias name for a computer on your network. To avoid confusions that could arise from multiple names for the same server, we recommend that you use the same catalog node name as the network name for the server.

When naming a database, database alias, or catalog node name, see “General Naming Rules” on page 509. In addition, the name you specify can *only* contain 1 to 8 characters.



To avoid potential problems, do not use the special characters @, #, and \$ in a database name if you intend to have a client remotely connect to a host database. Also, because these characters are not common to all keyboards, do not use them if you plan to use the database in another country.

On Windows NT and Windows 2000 systems, ensure that no instance name is the same as a service name.

Object Name Rules

Database objects include:

- Tables
- Views
- Columns
- Indexes
- User-defined functions (UDFs)
- User-defined types (UDTs)
- Triggers
- Aliases
- Table spaces
- Schemas

When naming database objects, see “General Naming Rules” on page 509.

In addition, the name you specify:

- Can contain 1 to 18 characters *except* for the following:
 - Table names (including view names, summary table names, alias names, and correlation names), which can contain up to 128 characters
 - column names, which can contain up to 30 characters
 - schema names, which can contain up to 30 characters
- Cannot be any of the SQL reserved words that are listed in the *SQL Reference*.

Using delimited identifiers, it is possible to create an object that violates these naming rules; however, subsequent use of the object could result in errors.

For example, if you create a column with a + or – sign included in the name and you subsequently use that column in an index, you will experience problems when you attempt to reorganize the table. To avoid potential problems with the use and operation of your database, *do not* violate these rules.

Username, User ID, Group Name, and Instance Name Rules

Usernames or *User IDs* are the identifiers assigned to individual users. When naming users, groups, or instances, see “General Naming Rules” on page 509.

In addition to the general naming rules:

- User IDs on OS/2 can contain 1 to 8 characters. They cannot start with a numeric digit or end with \$.
- Usernames on UNIX can contain 1 to 8 characters.
- Usernames on Windows can contain 1 to 30 characters. The Windows NT and Windows 2000 operating systems currently have a limit of 20 characters.
- Group and instance names can contain 1 to 8 characters.
- Names cannot be any of the following:
 - USERS
 - ADMINS
 - GUESTS
 - PUBLIC
 - LOCAL
- Names cannot begin with:
 - IBM
 - SQL
 - SYS
- Names cannot include accented characters.
- In general, when naming users, groups, or instances:
 - OS/2** Use uppercase names.
 - UNIX** Use lowercase names.
 - Windows 32-bit operating systems**
Use any case.

Workstation Name (nname) Rules

A *workstation* name specifies the NetBIOS name for a database server or client that resides on the local workstation. This name is stored in the database manager configuration file. The workstation name is known as the *workstation nname*. When naming workstations, see “General Naming Rules” on page 509.

In addition, the name you specify:

- Can contain 1 to 8 characters
- Cannot include &, #, and @
- Must be unique within the network

DB2SYSTEM Naming Rules

DB2 uses the *DB2SYSTEM* name to identify a physical DB2 machine, system, or workstation within a network. On UNIX, the *DB2SYSTEM* name defaults to the TCP/IP hostname. On OS/2, you must specify the *DB2SYSTEM* name during install. On Windows 32-bit operating systems, you do not need to specify a *DB2SYSTEM* name; the DB2 setup program detects the Windows Computer name and assigns it to *DB2SYSTEM*.

When creating a *DB2SYSTEM* name, see “General Naming Rules” on page 509.

In addition, the name you specify:

- Must be unique within a network
- Can contain a maximum of 21 characters

Password Rules

When determining passwords, consider the following rules:

OS/2 A maximum of 14 characters.

UNIX A maximum of 8 characters.

Windows 32-bit operating systems
A maximum of 14 characters.

Appendix G. Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make

improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licenses 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
Office of the Lab Director
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 information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

Trademarks

The following terms, which may be denoted by an asterisk(*), are trademarks of International Business Machines Corporation in the United States, other countries, or both.

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

The following terms are trademarks or registered trademarks of other companies:

Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation.

Java or all Java-based trademarks and logos, and Solaris are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Tivoli and NetView are trademarks of Tivoli Systems Inc. in the United States, other countries, or both.

UNIX is a registered trademark in the United States, other countries or both and is licensed exclusively through X/Open Company Limited.

Other company, product, or service names, which may be denoted by a double asterisk(**) may be trademarks or service marks of others.

Index

Special Characters

.INI file

- db2cli.ini 196
- ODBC 196

A

access profiles

- adding databases 30
- client 36
- creating 36
- server 36
- using 36

accessing DB2 servers

- configuring communications
 - IBM eNetwork
 - Communication Server V5 for AIX 95
 - SNAPplus2 for HP-UX 102
 - SunLink SNA PU 2.1 for Solaris 106
 - using IBM Personal Communications for Windows 32-Bit Operating Systems 79
- IPX/SPX 63
- Named Pipes 42
- NetBIOS 42, 56
- overview 41
- TCP/IP 47, 257

accessing host servers

- configuring communications
 - for IBM Personal Communications for Windows 32-Bit Operating Systems 90
- IBM eNetwork
 - Communication Server V5 for AIX 315
- IBM Personal
 - Communications for Windows 32-Bit Operating Systems 284, 303
- SNA API Client 300
- SNAP-IX for Sparc Solaris 336
- SNAPplus2 for HP-UX 327
- SunLink SNA Version 9.1 for Solaris 344
- using the SNA API Client 88

accessing multiple servers

- setting configuration parameters 41
- add database wizard 502, 503
- adding databases
- manually 34
 - using access profiles 30
 - using Discovery 32

AIX

- configuring Stored Procedure Builder 130
- mounting CD-ROM 453

APPC

- Bull SNA 324
 - Communications Manager for OS/2 6, 271
 - Communications Server for OS/2 6, 271
 - Communications Server for Windows NT SNA Client 300
 - configuring Bull SNA for 99
 - configuring Communications Manager for OS/2 74
 - configuring IBM eNetwork Communications Server for OS/2 74
 - configuring SNAPplus2 for HP-UX 102
 - configuring SunLink SNA for Solaris 106
 - configuring the Communications Server for Windows SNA Client 88
 - manually configuring 267
 - Microsoft SNA Client 93
 - server 152
 - SNAPplusLink 95, 284, 315, 327, 344
 - software requirements 6, 8
 - SunLink SNA 7
 - support on OS/2
 - Communications Manager for OS/2 166
 - Communications Server for OS/2 166
 - verifying the connection 113
- APPENDAPINAME keyword 205
- APPL 270

application development

- using ODBC 187
- ASYNCCENABLE keyword 206
- Asynchronous ODBC, enabling 206
- authentication
 - server 137

B

backup database wizard 502

binding

- utilities 186
- BITDATA keyword 206
- books 487, 497

C

catalog node name

- naming rules 509
- cataloging
- APPC node 109, 110, 347, 348
 - databases 43, 44, 52, 59, 67, 110, 262, 348, 349
 - IPX/SPX node 66, 67, 151
 - Named Pipe node 43
 - NetBIOS node 58, 147
 - nodes
 - in a federated environment 428
 - TCP/IP node 51, 52, 143, 261
- CLI/ODBC keywords 203
- client profiles
- creating 37
 - definition 36
 - importing 38
 - using 37
- clients
- configuring 41
 - installing 11
- CLIPKG keyword 207
- CLISHEMA keyword 208
- Command Center
 - entering DB2 commands 448
 - entering SQL statements 448
- commands
- db2cc 120
 - db2insthtml 485
 - db2jstrt 118
 - db2sampl 120
 - db2set 138
 - db2setup 23, 27
 - rlogin 27

- commands (*continued*)
 - sniffle 124
- communication protocols
 - APPC 5, 6, 7, 8, 71, 74, 152, 166, 267, 271
 - configuring 47, 257
 - IPX/SPX 5, 6, 7, 8, 63, 147
 - Named Pipes 6, 8, 42, 139
 - NetBIOS 6, 8, 56, 143
 - TCP/IP 5, 6, 7, 8, 47, 139, 257
- communications
 - APPC 71
 - configuring the client 42, 47, 56, 63, 71
 - configuring the server 133, 137
 - Control Center 133
 - IPX/SPX 147
 - managing 41, 137
 - Named Pipes 42, 139
 - NetBIOS 56, 143
 - TCP/IP 47, 63, 139
- Communications Server for Windows NT SNA Client
 - configuring manually 300
 - version required 300
- Communications Server for Windows SNA Client
 - configuring manually 88
 - version required 88
- configuration parameters
 - setting DB2 137, 452
 - SYSADM_GROUP 452
- configure multisite update wizard 502
- configuring
 - APPC 71, 152
 - for OS/2 166
 - application server 268
 - AS/400 268
 - configuring SunLink SNA for Solaris 109
 - DB2 clients
 - using the Client Configuration Assistant (CCA) 29
 - DRDA server 268
 - IBM eNetwork Communication Server for AIX 315
 - IBM eNetwork Communications Server for OS/2 74
 - IBM eNetwork Communications Server for Windows NT SNA API Client 300
 - IBM Personal Communications for Windows 32-Bit Operating Systems 79
- configuring (*continued*)
 - IBM Personal Communications for Windows NT and Windows 9x 284
 - IPX/SPX 63
 - Microsoft SNA Server for Windows 303
 - Microsoft SNA Server for Windows using APPC 90
 - MVS 268
 - Named Pipes 42, 139
 - NetBIOS 56, 143
 - ODBC driver 190, 192
 - OS/2 271
 - servers 137
 - SNAP-IX Version 6.0.1 for SPARC Solaris 336
 - SNAPPlus 315, 327, 345
 - SNAPPlus2 for HP-UX 102, 327
 - SQL/DS 268
 - steps to configure SNAPPlus 95
 - SunLink SNA for Solaris 344
 - TCP/IP 47, 139
 - VM 268
 - VSE 268
- configuring client communications
 - setting configuration parameters 41
 - using the Command Line Processor 41
- configuring communications
 - using Control Center 133
- configuring server communications
 - setting DB2COMM 137
 - using the command line processor 137
 - using the Control Center 133
- configuring servers 133
- connection
 - testing APPC 351
- CONNECTNODE keyword 209
- CONNECTTYPE keyword 210
- Control Center
 - administering DB2 Connect Enterprise Edition 125
 - administering DB2 for OS/390 125
 - as a Java applet 115
 - as a Java application 115
 - configuring to work with a web server 122
 - customizing db2cc.htm 121
 - functional considerations 122
 - JDBC Applet Server 118
 - machine configurations 116
- Control Center (*continued*)
 - running as an applet 120
 - running as an application 120
 - setting up to run as an applet 118
 - supported browsers 117
 - supported Java Runtime Environments (JRE) 117
 - troubleshooting information 124
 - UNIX installation tips 122
 - control point name 73, 153, 270
 - create database wizard 503
 - CREATE NICKNAME statement
 - for DB2 family data sources 430
 - for Oracle data sources 440
 - CREATE SERVER statement
 - for DB2 family data sources 429
 - for Oracle data sources 438
 - create table space wizard 503
 - create table wizard 503
 - CREATE USER MAPPING statement
 - for DB2 family data sources 430
 - for Oracle data sources 439
 - CREATE WRAPPER statement
 - for DB2 family data sources 429
 - for Oracle data sources 438
 - creating profiles
 - clients 37
 - server 36
 - creating the sample database
 - cataloging a database 44, 52, 59, 67, 110
 - cataloging a node 43, 51, 58, 66, 109
 - connecting to a database 46, 54, 61, 69, 113, 265
 - updating the database manager configuration 58, 142, 146, 150
 - CURRENTFUNCTIONPATH
 - keyword 210
 - CURRENTPACKAGESET
 - keyword 211
 - CURRENTREFRESHAGE
 - keyword 212
 - CURRENTSCHEMA keyword 212
 - CURRENTSQLID keyword 213
 - CURSORHOLD keyword 213
- D**
 - database alias
 - definition 44, 53, 60, 68, 111
 - naming rules 509
 - DATABASE keyword 214
 - database objects
 - naming rules 510

- databases
 - cataloging 262, 349
 - creating a sample database 262, 349
 - naming rules 509
 - DB2 clients
 - changing privileges 452
 - installing 11
 - installing on UNIX workstations 27
 - licensing 11
 - OS/2 19
 - pre-version 7 11
 - software requirements 4
 - supported platforms 11
 - updating kernel parameters on
 - HP-UX, NUMA-Q/PTX and Solaris 24
 - WIN-OS/2 support 19
 - Windows 32-bit operating systems 15
 - DB2 Connect
 - configuring for OS/2 74
 - configuring on OS/2 166, 271
 - DB2 Connect Enterprise Edition
 - configuring to accept APPC clients 156
 - DB2_DJ_COMM environment variable
 - setting for DB2 family data sources 429
 - setting for Oracle data sources 438
 - DB2 Enterprise Edition
 - configuration planning 3
 - memory requirements 3
 - DB2 family data sources in a federated system
 - cataloging nodes 428
 - creating nicknames 430
 - creating server mappings 429
 - creating user mappings 430
 - creating wrappers 429
 - setting DB2_DJ_COMM 429
 - verifying connections 431
 - DB2 library
 - books 487
 - Information Center 501
 - language identifier for books 495
 - late-breaking information 496
 - online help 498
 - ordering printed books 497
 - printing PDF books 496
 - DB2 library (*continued*)
 - searching online
 - information 504
 - setting up document server 503
 - structure of 487
 - viewing online information 500
 - wizards 502
 - DB2 Relational Connect
 - installing 433
 - DB2 Security Server
 - starting on Windows NT or Windows 2000 119
 - DB2 Universal Database
 - software requirements 4
 - DB2 Workgroup Edition
 - configuration planning 3
 - memory requirements 3
 - db2classes.exe 120
 - db2classes.tar.Z 120
 - db2cli.ini 196
 - DB2COMM 137
 - DB2CONNECTVERSION
 - keyword 215
 - DB2DEGREE keyword 215
 - DB2DJ.ini file 435
 - DB2ESTIMATE keyword 216
 - DB2EXPLAIN keyword 217
 - DB2NBADAPTERS 144
 - DB2NODE 209
 - DB2OPTIMIZATION keyword 218
 - db2set command
 - using 138
 - db2setup
 - using to install DB2 clients 23
 - DB2SYSTEM
 - naming rules 512
 - DBALIAS keyword 219
 - DBNAME (VSE or VM) 270
 - DBNAME keyword 219
 - DEFAULTPROCLIBRARY
 - keyword 220
 - DEFERREDPREPARE keyword 221
 - DISABLEMULTITHREAD
 - keyword 222
 - Discovery
 - adding databases 32
 - disk requirements
 - client 3
 - server 3
 - Distributed Computing Environment
 - software requirements 5, 6
 - documentation
 - setting up on a web server 479
- ## E
- EARLYCLOSE keyword 222
 - environment variables
 - setting for DB2 family data sources 429
 - setting for Oracle data sources 435
 - Export function 35, 37
- ## F
- federated database system
 - DB2 family data sources
 - cataloging nodes 428
 - creating nicknames 430
 - creating server mappings 429
 - creating user mappings 430
 - creating wrappers 429
 - setting DB2_DJ_COMM 429
 - installation considerations 427, 433
 - OLE DB data sources 443
 - Oracle data sources
 - creating nicknames 440
 - creating server mappings 438
 - creating user mappings 439
 - creating wrappers 438
 - setting DB2_DJ_COMM 438
 - overview 423
 - supported data sources 424
 - File DSN
 - database to connect to 214
 - host name 226
 - IP address 226
 - protocol used 235
 - service name 238
 - fixed disks
 - hardware requirements 3
- ## G
- GRANTEELIST keyword 223
 - GRANTORLIST keyword 224
 - GRAPHIC keyword 225
- ## H
- hardware requirements
 - fixed disk 3
 - HOSTNAME keyword 226
 - HP-UX
 - mounting CD-ROM 454
 - HP-UX client
 - updating kernel components 24
 - HTML
 - sample programs 495

- I**
- IGNOREWARNINGS keyword 226
- IGNOREWARNLIST keyword 227
- Import function 35
- importing profiles
 - client 38
- IN DATABASE command 219
- index wizard 503
- Information Center 501
- initialization file, ODBC 196
- installing
 - CID using SystemView LAN 17, 20
 - client 3
 - DB2 Application Development Clients 11
 - OS/2 19
 - Windows 32-bit operating systems 15
 - DB2 clients 11
 - OS/2 19
 - DB2 clients on UNIX workstations 27
 - Distributed Join for DB2 Databases option 427
 - Distributed Join for Oracle Databases option 433
 - errors 17, 20
 - federated database system considerations 427, 433
 - log 17, 20
 - Netscape browser 501
 - OS/2 375
 - OS/2 client 19
 - remote clients 27
 - server 3
 - updating kernel components 25
 - Windows 3x 375
- instances
 - naming restrictions 511
- IPX/SPX
 - client 63
 - configuring 63
 - server 147
 - software requirements 6, 7
 - supported platforms 63, 148
 - verifying the connection 46, 61, 69, 113, 265
- J**
- Java
 - running programs 198
- Java Runtime Environment (JRE) defined 115
- Java Virtual Machine (JVM) 115
- JDBC
 - running programs 198
- JDBC Applet Server 118
- JRE
 - supported levels for Control Center 117
- K**
- KEEPCONNECT keyword 227
- KEEPSTATEMENT keyword 228
- kernel configuration parameters
 - updating on UNIX clients 24
- keywords
 - CLI/ODBC 203
- L**
- LANG environment variable 505
- language identifier
 - books 495
- late-breaking information 496
- Licensed Program Products
 - default value 146
- Linux
 - mounting CD-ROM 455
- LOBMAXCOLUMNSIZE keyword 229
- local adapter address 73, 270
- local control point name 73, 270
- local LU name 73, 154, 270
- LOCATION NAME (MVS, OS/390) 270
- LONGDATACOMPAT keyword 229
- LU 73, 154, 270
- M**
- managing connections
 - overview 41
 - using the Command Line Processor 41, 137
- manually adding databases 34
- MAXCONN keyword 230
- memory requirements
 - client 3
 - estimating 3
 - recommended 3
 - server 3
- Microsoft ODBC Driver Manager 189
- Microsoft SNA Client
 - configuring 93, 312
 - version required 93, 312
- Microsoft SNA Server
 - version required 8
- MODE keyword 231
- mode name 73, 154, 270
- MODEENT 154, 270
- mounting CD-ROM
 - AIX 453
 - HP-UX 454
 - Linux 455
 - PTX 455
 - Solaris 455
- MULTICONNECT keyword 231
- multisite updates 353
- N**
- Named Pipes
 - client 42
 - configuring 42
 - server 139
 - software requirements 6
- naming rules
 - database alias 509
 - database objects 510
 - databases 509
 - general 509
 - groups 511
 - instance names 511
 - password 512
 - userids 511
 - username 511
- National Language Support (NLS)
 - Oracle code page options 441
- Nbf
 - network route required by DB2 146
- net8 wrapper 438
- NetBIOS
 - autostartingNetBIOS 147
 - code set 505
 - codepage support 505, 506
 - determining codepage 509
 - on the client 6, 56
 - on the server 137, 143
- Netfinity Server 6
- NetQuestion 459
 - additional installation considerations on UNIX 475
 - additional installation prerequisites on OS/2 470
 - browser settings on Windows 32 466
 - changing the port number on OS/2 471
 - double-byte character set (DBCS) 460
 - for OS/2 470
 - for UNIX operating systems 475
 - for Windows 32 464
 - installation workarounds on UNIX 475

- NetQuestion 459 *(continued)*
 - locating install directory on OS/2 472
 - locating install directory on Windows 32 466
 - locating the DB2 Information page 460
 - manually uninstalling on OS/2 474
 - manually uninstalling on Windows 32 469
 - netqinit command 472
 - overview 459
 - search system components 459
 - searching DB2 online information 460
 - searching while disconnected on OS/2 472
 - searching with a laptop on Windows 9x 467
 - searching with proxies enabled in Netscape or Internet Explorer 466
 - single-byte character set (SBCS) 460
 - solving search problems 460
 - starting the search server 460
 - starting the web server 461
 - switching communication types on OS/2 470
 - TCP/IP configuration on Windows 32 465
 - troubleshooting 460
 - uninstalling on OS/2 472
 - uninstalling on Windows 32 468
- Netscape browser
 - installing 501
- network
 - ID 73, 270
 - name 73, 270
- network route
 - setting to Nbf 144, 146
- nicknames
 - for DB2 family data sources 430
 - for Oracle data sources 440
- nodes
 - cataloging in a federated environment 428
- NUMA-Q/PTX client
 - updating kernel components 25
- O**
- ODBC
 - catalog for DB2 Connect 208
 - odbc.ini file 196
- ODBC *(continued)*
 - odbcinst.ini file 196
 - registering the driver manager 189
 - running applications on OS/2 21
 - running ODBC-enabled applications 187
 - running programs 187
- odbcad32.exe 189
- OLE DB data sources 443
- online help 498
- online information
 - searching 504
 - viewing 500
- optimize for N rows
 - configuration keyword 232
- OPTIMIZEFORNROWS
 - keyword 232
- OPTIMIZESQLCOLUMNS
 - keyword 233
- ORA_NLS 436
- ORACLE_BASE 436
- Oracle data sources
 - code page options 441
 - creating nicknames 440
 - creating server mappings 438
 - creating user mappings 439
 - creating wrappers 438
 - installing client software 435
 - setting DB2_DJ_COMM 438
 - setting environment variables 435
 - tnsnames.ora file 439, 440
 - verifying connections 441
- ORACLE_HOME 436
- P**
- Parallelism, Setting degree of 215
- parameters
 - SYSADM_GROUP 452
- partner
 - LU name 73, 270
 - node name 73, 270
- passwords
 - naming rules 512
- PATCH1 keyword 233
- PATCH2 keyword 234
- PDF 496
- performance configuration
 - wizard 503
- planning
 - DB2 configuration 3
 - DB2 Connect configuration 3
- POPUPMESSAGE keyword 235
- printing PDF books 496
- privileges
 - required 452
- profiles
 - client 35, 36
 - export 35
 - server 36
- PROTOCOL keyword 235
- protocols
 - APPC 71, 152, 267
 - IPX/SPX 63, 147
 - Named Pipes 42, 139
 - NetBIOS 56, 143
 - TCP/IP 47, 139, 257
- PTX
 - mounting CD-ROM 455
- PU 73, 153, 270
- PWD keyword 236
- Q**
- QUERYTIMEOUTINTERVAL
 - keyword 236
- R**
- RDB name (AS/400) 270
- REFRESH DEFERRED 212
- REFRESH IMMEDIATE 212
- registering
 - ODBC driver manager 189
- registry values
 - db2comm 137
 - db2nbadapters 144
- relational database name 73, 270
- release notes 496
- remote
 - link address 73, 270
 - transaction program 73, 270
- response file keywords
 - DB2.AUTOSTART 381
 - DB2.DB2SATELLITEAPPVER 382
 - DB2.DB2SATELLITEID 382
 - DB2.SATCTLDB_PASSWORD 382
 - DB2.SATCTLDB_USERNAME 382
 - DB2.USERDB_NAME 382
 - DB2.USERDB_RECOVERABLE 383
 - DB2.USERDB_REP_SRC 383
- response files
 - OS/2 404
 - Windows 3.x 404
- restore wizard 503
- restrictions
 - instance name 511
- running applications
 - database client 185
 - ODBC considerations 187

- S**
- sample programs
 - cross-platform 495
 - HTML 495
 - SCHEMALIST keyword 237
 - searching
 - DB2 information using
 - NetQuestion 459
 - online information 502, 504
 - server mappings
 - for DB2 family data sources 429
 - for Oracle data sources 438
 - server profiles
 - creating 36
 - definition 36
 - SERVICENAME keyword 238
 - SET CURRENT SCHEMA 212
 - setting configuration
 - parameters 137
 - setting up client communications
 - using the Command Line Processor 41
 - setting up document server 503
 - setting up server communications
 - using the command line processor 137
 - using the Control Center 133
 - SmartGuides
 - wizards 502
 - SNA
 - configuring Communications Server for Windows NT SNA Client manually 300
 - configuring Microsoft SNA Client manually on Windows 93, 312
 - configuring SNAPlus 315, 345
 - configuring SunLink SNA for Solaris 109
 - configuring the Communications Server for the Windows SNA Client manually 88
 - steps to configure SNAPlus 95
 - snapmanage 109
 - software requirements
 - communication protocols 4
 - DB2 Application Development Clients 4, 5, 6, 7, 8
 - DB2 clients 4, 5, 6, 7, 8
 - DB2 Connect 4
 - DB2 Universal Database 4
 - Net.Data 4, 5, 6, 7, 8
 - Solaris
 - configuring Stored Procedure Builder 130
 - mounting CD-ROM 455
 - Solaris client
 - updating kernel components 26
 - sqlnet wrapper 438
 - SQLSTATEFILTER keyword 239
 - SSCP 270
 - STATICCAPFILE keyword 239
 - STATICLOGFILE keyword 240
 - STATICMODE keyword 240
 - STATICPACKAGE keyword 241
 - Stored Procedure Builder 129
 - configuring as an add-in with Visual Basic 129
 - configuring as an add-in with Visual Studio 129
 - configuring on Solaris 130
 - SunLink SNA subsystem
 - checking status 109
 - starting and stopping 109
 - symbolic destination name 73, 270
 - SYNCPPOINT keyword 242
 - SYSADM
 - controlling 452
 - SYSADM_GROUP parameter 452
 - SYSSCHEMA keyword 243
- T**
- TABLETYPE keyword 244
 - target database name 73, 270
 - target logical node 209
 - TCP/IP
 - client 47
 - client-to-server connection 257
 - configuring 47
 - configuring on OS/2 123
 - enabling localhost on OS/2 123
 - enabling loopback on OS/2 123
 - hostname 259
 - preventing socket collisions 48, 140, 257
 - resolving host address 143
 - server 139
 - setting up client-to-server 47
 - software requirements 5, 6, 7
 - supports only one-phase commit 363
 - testing database connection 266
 - troubleshooting 48, 140, 257
 - verifying on OS/2 124
 - TEMPDIR keyword 245
 - TNS_ADMIN 436
 - tnsnames.ora file 439, 440
 - TRACE keyword 246
 - TRACECOMM keyword 246
 - TRACEFILENAME keyword 247
 - TRACEFLUSH keyword 248
 - TRACEPATHNAME keyword 249
 - two-phase commit 353
 - TXNISOLATION keyword 250
- U**
- UID keyword 250
 - UNDERSCORE keyword 251
 - unit of work
 - distributed 353
 - user mappings
 - for DB2 family data sources 430
 - for Oracle data sources 439
 - username
 - naming rules 511
 - using the command line processor (CLP)
 - cataloging a node 261, 347
 - utilities
 - binding 185
- V**
- verifying the connection
 - APPC 71
 - IPX/SPX 63, 147
 - Named Pipes 42, 139
 - NetBIOS 56, 143
 - TCP/IP 47, 139, 152, 257
 - viewing
 - online information 500
 - VTAM
 - application name is Partner LU name 270
- W**
- WARNINGLIST keyword 252
 - web server
 - advantages 479
 - considerations when serving DB2 documentation 479
 - disadvantages 479
 - Lotus Domino Go! 481
 - Microsoft Internet Information Server 484
 - Microsoft Peer Web Services 484
 - Netscape Enterprise Web Server 482
 - serving documentation for multiple platforms 486
 - serving documentation in multiple languages 484
 - setting up DB2 documentation 480
 - typical scenarios 481
 - Windows 2000
 - starting the security server 119
 - Windows NT
 - starting the security server 119

wizards

- add database 502, 503
- backup database 502
- completing tasks 502
- configure multisite update 502
- create database 503
- create table 503
- create table space 503
- index 503
- performance configuration 503
- restore database 503

workstation name (nname)

- naming rules 512

wrappers

- for DB2 family data sources 429
- for Oracle data sources 438

Contacting IBM

If you have a technical problem, please review and carry out the actions suggested by the *Troubleshooting Guide* before contacting DB2 Customer Support. This guide suggests information that you can gather to help DB2 Customer Support to serve you better.

For information or to order any of the DB2 Universal Database products contact an IBM representative at a local branch office or contact any authorized IBM software remarketer.

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-237-5511 for customer support
- 1-888-426-4343 to learn about available service options

Product Information

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-IBM-CALL (1-800-426-2255) or 1-800-3IBM-OS2 (1-800-342-6672) to order products or get general information.
- 1-800-879-2755 to order publications.

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

The DB2 World Wide Web pages provide current DB2 information about news, product descriptions, education schedules, and more.

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

The DB2 Product and Service Technical Library provides access to frequently asked questions, fixes, books, and up-to-date DB2 technical information.

Note: This information may be in English only.

<http://www.elink.ibm.com/pbl/pbl/>

The International Publications ordering Web site provides information on how to order books.

<http://www.ibm.com/education/certify/>

The Professional Certification Program from the IBM Web site provides certification test information for a variety of IBM products, including DB2.

ftp.software.ibm.com

Log on as anonymous. In the directory /ps/products/db2, you can find demos, fixes, information, and tools relating to DB2 and many other products.

comp.databases.ibm-db2, bit.listserv.db2-l

These Internet newsgroups are available for users to discuss their experiences with DB2 products.

On Compuserve: GO IBMDB2

Enter this command to access the IBM DB2 Family forums. All DB2 products are supported through these forums.

For information on how to contact IBM outside of the United States, refer to Appendix A of the *IBM Software Support Handbook*. To access this document, go to the following Web page: <http://www.ibm.com/support/>, and then select the IBM Software Support Handbook link near the bottom of the page.

Note: In some countries, IBM-authorized dealers should contact their dealer support structure instead of the IBM Support Center.



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC09-2957-00

