DB2 Universal Database and DB2 Connect

Installation and Configuration Supplement

Version 6

GC09-2857-00

DB2 Universal Database and DB2 Connect

Installation and Configuration Supplement

Version 6

GC09-2857-00

Before using this information and the product it supports, be sure to read the general information under “Appendix H. Notices” on page 549.

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, 1999. 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
Disk Requirements 3
Estimating Fixed Disk Requirements 3
Software Requirements. 6
Client Product Requirements. 6
Possible Client-to-Server Connectivity Scenarios 10

Chapter 2. Installing DB2 Clients 13

Chapter 3. Installing DB2 Clients on Windows 32-Bit Operating Systems . . . 15
Before You Begin. 15
Installation Steps. 15

Chapter 4. Installing DB2 Clients on OS/2 Operating Systems. 19
Before You Begin. 19
Installation Steps. 20

Chapter 5. Installing DB2 Clients on UNIX Operating Systems. 21
Before You Begin. 21
Installation Steps. 21
Step 1. Update Kernel Configuration Parameters. 21
Step 2. Mount the CD-ROM 23
Step 3. Perform the Installation 26

Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant 31
Configuration Steps 32
Adding a Database Using a Profile. . . 32
Adding a Database Using Discovery 33
Adding a Database Manually 36
Creating and Using Profiles 38
Server Profiles. 38
Client Profiles. 39

Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor 43
Configuring Named Pipes on the Client 43
Step 1. Identify and Record Parameter Values 44
Step 2. Configure the Client 44
Step 3. Test the Client-to-Server Connection. 47
Configuring TCP/IP on the Client 48
Step 1. Identify and Record Parameter Values 49
Step 2. Configure the Client 50
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 . . . 62
Step 1. Identify and Record Parameter Values 63
Step 2. Configure the Client 65
Step 3. Test the Client-to-Server Connection. 68
Configuring APPC on the Client 70
Step 1. Identify and Record Parameter Values 70
Step 2. Update the APPC Profiles . . . 72
Step 3. Catalog the APPC or APPN Node 108
Step 4. Catalog the Database 109
Step 5. Test the Client-to-Server Connection. 111
Troubleshooting the Client-to-Server Connection. 112

Part 2. Setting Up Server Communications 113

© Copyright IBM Corp. 1993, 1999

Chapter 8. Using the Control Center to Configure Server Communications . . . 115
Configuring DB2 Communications for Local Instances 116
Configuring DB2 Communications for Remote Instances. 117

Chapter 9. Using the Command Line Processor to Configure Server Communications 119
Setting the DB2COMM Registry Parameter 119
Configuring Named Pipes on the Server 121
Configuring TCP/IP on the Server 122
Step 1. Identify and Record Parameter Values 122
Step 2. Configure the Server 123
Configuring NetBIOS on the Server . . . 125
Step 1. Identify and Record Parameter Values 125
Step 2. Configure the Server 126
Autostarting DB2 with NetBIOS (for Windows NT only) 129
Configuring IPX/SPX on the Server . . . 129
Step 1. Identify and Record Parameter Values 131
Step 2. Configure the Server 132
Configuring APPC on the Server 135
Step 1. Identify and Record Parameter Values 135
Step 2. Configure the Server 136

Part 3. CLI/ODBC Enablement . . 167

Chapter 10. Running Your Own Applications 169
Binding Database Utilities. 169
Running CLI/ODBC Programs 171
Platform Specific Details for CLI/ODBC Access 173
Detailed Configuration Information . . 179
Running Java Programs 182
Configuring the Environment 183
Java Applications 185
Java Applets 185

Chapter 11. DB2 CLI/ODBC Configuration Keyword Listing. 187
Configuration Keywords by Category . . . 187
CLI/ODBC Settings General Configuration Keywords 187
Compatibility Configuration Keywords 187
Data Type Configuration Keywords . . 187
Enterprise Configuration Keywords . . 188
Environment Configuration Keywords 188
Optimization Configuration Keywords 188
Service Configuration Keywords . . . 189
Transaction Configuration Keywords 189
Configuration Keyword Descriptions . . . 190
APPENDAPINAME. 190
ASYNCENABLE 190
BITDATA 191
CLISCHEMA 192
CONNECTNODE 193
CONNECTTYPE 193
CURRENTFUNCTIONPATH 194
CURRENTPACKAGESET 195
CURRENTREFRESHAGE 196
CURRENTSCHEMA 196
CURRENTSQLID 197
CURSORHOLD 197
DB2CONNECTVERSION 198
DB2DEGREE 199
DB2ESTIMATE 199
DB2EXPLAIN 200
DB2OPTIMIZATION 201
DBALIAS 202
DBNAME 203
DEFAULTPROCLIBRARY 204
DEFERREDPREPARE 204
DISABLEMULTITHREAD. 205
EARLYCLOSE. 206
GRANTEELIST 207
GRANTORLIST 207
GRAPHIC 208
IGNOREWARNINGS 209
IGNOREWARNLIST 210
KEEPCONNECT. 210
KEEPSTATEMENT 211
LOBMAXCOLUMNSIZE 212
LONGDATACOMPAT 212
MAXCONN 213
MODE 214
MULTICONNECT 214
OPTIMIZEFORNROWS 215
OPTIMIZESQLCOLUMNS 216
PATCH1 216
PATCH2 217
POPUPMESSAGE 218

PWD 218
SCHEMALIST. 219
SQLSTATEFILTER 220
SYNCPOINT 220
SYSSCHEMA 221
TABLETYPE 222
TEMPDIR 223
TRACE 224
TRACECOMM 225
TRACEFILENAME 225
TRACEFLUSH 226
TRACEPATHNAME 227
TXNISOLATION 228
UID 228
UNDERSCORE 229
WARNINGLIST 230

Part 4. Configuring DB2 Connect to Host or AS/400 Communications 231

Chapter 12. Configuring Host Communications Using the Command Line Processor 233

Chapter 13. Configuring TCP/IP Communications on the DB2 Connect Workstation 235
Step 1. Identify and Record Parameter Values 236
Step 2. Configure the DB2 Connect Workstation 238
A. Resolve the Host's IP Address . . . 238
B. Update the Services File 239
Step 3. Catalog the TCP/IP Node 240
Step 4. Catalog the Database as a Database Connection Service (DCS) Database . . . 241
Step 5. Catalog the Database 242
Step 6. Bind Utilities and Applications to the Database Server 243
Step 7. Test the Host or AS/400 Connection 244
Test the Host Connection 244

Chapter 14. Configuring APPC Communications on the DB2 Connect Workstation 247
Step 1. Identify and Record Parameter Values 247
Step 2. Update the APPC Profiles on the DB2 Connect Workstation 250
Configuring IBM eNetwork Communications Server for OS/2 . . . 251
Configuring IBM Personal Communications for Windows NT and Windows 9x 263
Configuring IBM eNetwork Communications Server for Windows NT 274
Configuring IBM eNetwork Communications Server for Windows NT SNA API Client 280
Configuring Microsoft SNA Server for Windows NT 283
Configuring Microsoft SNA Client . . . 292
Configuring IBM eNetwork Communication Server for AIX 295
Configuring Bull SNA for AIX 308
Configuring SNAPlus2 for HP-UX . . . 311
Configuring SunLink SNA for Solaris 324
Step 3. Catalog the APPC or APPN Node 327
Step 4. Catalog the Database as a Database Connection Service (DCS) Database . . . 328
Step 5. Catalog the Database 329
Step 6. Bind Utilities and Applications to the Database Server 330
Step 7. Test the Host or AS/400 Connection 331

Chapter 15. Enabling Multisite Updates (Two-Phase Commit) 333
Enabling Multisite Updates Using the Control Center 337
Starting the Multisite Update Smartguide 337
Smartguide Steps. 337
Testing the Multisite Update Feature 337

Chapter 16. Enabling Multisite Updates with Host and AS/400 Database Servers and Clients 339
Planning to Use the SPM 339
Software Prerequisites 339
Database Manager Configuration Parameters 340
Other Requirements 341
Configuring the SPM 342
Applications Which Access Host or AS/400 Database Servers 342
Host or AS/400 Applications Which Access a DB2 Universal Database Server in a Multisite Update 345

Setting up SNA Communications for the DB2 Syncpoint Manager (SPM) 346
Configuring Communications Server for OS/2. 346
Communications Server Operational Considerations 347
Configuring Communications Server for AIX 348
Configuring Communications Server for Windows NT 348
Registering CS/NT as a Windows NT Service 349
SPM Migration Considerations 350
SPM Logging 350
SPM Log File Size 351
Changing the SPM Log File Size . . . 351
Performance Characteristics 352
Memory Utilization 352
SPM Protocol Violation Records 352

Part 5. Configuring DB2 UDB as a DRDA Application Server 353

Chapter 17. Accessing DB2 Universal Database Servers from Host and AS/400 Applications 355
Supported Clients 355
PTFs Required 356
Configuration Steps for DB2 Universal Database Server 356
Configuring DB2 Universal Database Servers for Host or AS/400 Client Access 357
Using the DB2 Universal Database Server from Host or AS/400 Clients 358
Authentication 358
Troubleshooting 359
DRDA Functions Supported 359
Security and Auditability 364
Configuration Considerations 365

Part 6. Thin Client and Thin Connect Architecture 367

Chapter 18. Distributed Thin Client Installation 369
Installing and Configuring Thin Client Workstations to use a Code Server 369
Before You Begin 369
Performing the Installation 370

Chapter 19. DB2 Thin Connect Workstations in a Windows 32-bit Operating System Environment 377
Installing and Configuring Thin Connect Workstations to use a Code Server 377
Before You Begin 377
Performing the Installation 378

Part 7. Other DB2 for UNIX Installation Methods 387

Chapter 20. Other Methods to Install AIX Server 389
Installation Steps 389
Step 1. Install DB2 390
Install Products or Filesets Using SMIT 390
Install Software Bundles Using SMIT 391
Install the DB2 Product Library (Optional) 392
Install the DB2 Product Messages (Optional) 393
Step 2. Create or Assign a Group and User ID. 393
Step 3. Create a DB2 Instance 395
Step 4. Create the Administration Server 395
Step 5. Install the License Key 395
Step 6. Create Links for DB2 Files (Optional) 396

Chapter 21. Other Methods to Install HP-UX Servers 399
Installation Steps 399
Step 1. Install DB2 399
Install the DB2 Product Library (Optional) 401
Install DB2 Product Messages (Optional) 403
Install DB2 Control Center and Help (Optional) 403
Step 2. Update the Kernel Configuration Parameters 405
Step 3. Create or Assign Groups and User IDs 406
Step 4. Create a DB2 Instance 407
Step 5. Create an Administration Server 407
Step 6. Install the License Key 407
Step 7. Create Links for DB2 Files (Optional) 408

Chapter 22. Other Methods to Install DB2 for Solaris 409
Installation Steps 409
Step 1. Install DB2 409
Installing the DB2 Product Library (Optional) 411
Installing the DB2 Product Messages (Optional) 412
Step 2. Update the Kernel Configuration Parameters 413
Step 3. Create or Assign Groups and User IDs 414
Step 4. Create a DB2 Instance 415
Step 5. Create an Administration Server 415
Step 6. Install the License Key 416
Step 7. Create Links for DB2 Files (Optional) 416

Chapter 23. Contents of the DB2 Products for UNIX platforms 419
Packaging 419
Products and Selectable Components . . . 421

Part 8. Distributed Installation . . 423

Chapter 24. An Introduction to Distributed Installation 425
Types of Distributed Installation. 425
Response File 425
Available Sample Response Files 426
Important Keywords for OS/2 and Windows 32-Bit Operating Systems . . . 427
Response File Generator 429
Where Do You Go From Here? 430

Chapter 25. Distributed DB2 Installation on Windows 32-Bit Operating Systems 431
Before You Begin 431
Step 1. Make DB2 Files Available for Installation 431
Set up Shared Access 432
Create a Response File 433
Step 3. Run Setup with the Response File from the Client Workstation 434
DB2 Product Installation Using SMS . . . 436
SMS Requirements 436
Step 1. Import the DB2 Install File into SMS on the SMS Server 436
Step 2. Create the SMS Package on the SMS Server. 437
Step 3. Distribute the DB2 Installation Package from the SMS Server 438
Configuring Client Settings 440
Configuring Remote Access to a Server Database 440
Configuring db2cli.ini 442
Exporting and Importing an Instance Profile 442

Chapter 26. Distributed DB2 Installation on UNIX Operating Systems 443
Before You Begin 443
Step 1. Mount the CD-ROM 443
Step 2. Create a Response File 443
Step 3. Start an Unattended Installation with a Response File 444

Chapter 27. Distributed DB2 Installation on OS/2 Operating Systems 445
Installing DB2 Products from a Hard Disk or CD-ROM 445
Before You Begin 445
Step 1. Make DB2 Files Available for Installation 445
Step 2. Create a Response File for Distributed Installation. 446
Step 3. Run the CMD File from the Remote Workstation. 447

Part 9. Configuring Distributed Requests and Access to Heterogeneous Data Sources . . 451

Chapter 28. Creating and Configuring a Federated Database System 453
Supported Data Sources 455

Chapter 29. Setting Up a Federated System to Access DB2 Family Data Sources 457
Enabling Federated Database Functionality 457
Adding DB2 Family Data Sources to a Federated System 457
Verifying Connections to DB2 Family Data Sources 460

Chapter 30. Setting Up a Federated System to Access Oracle Data Sources 463

Enabling Federated Database Functionality 463
Adding Oracle Data Sources to a Federated System 463
Oracle Code Page Options 470
Verifying Connections to Oracle Data Sources 470

Part 10. Appendixes 473

Appendix A. Basic Task Knowledge . . . 475
Starting the Software Registration Tool . . 475
Starting the Client Configuration Assistant 475
Starting the Control Center 476
Entering Commands Using the Command Center 477
Entering Commands Using the Command Line Processor 478
Command Line Mode 479
Interactive Input Mode 480
Working with the System Administrative Group 480

Appendix B. Migrating from Previous Versions and Releases 483
Migrating Instances 483
Step 1. Prepare the DB2 Instance for Migration 483
Step 3. Migrate the DB2 Instance . . . 484

Appendix C. Troubleshooting NetQuestion 487
NetQuestion for Windows 32–bit Operating Systems 487
Restrictions on Use 487
Installing NetQuestion 488
Working with NetQuestion 491
Uninstalling NetQuestion 495
NetQuestion for OS/2 Operating Systems 497
Installing NetQuestion 497
Working with NetQuestion 500
Uninstalling NetQuestion 503
Troubleshooting NetQuestion Installs on UNIX operating systems 505
NetQuestion installation workarounds on UNIX platforms 505
Installing NetQuestion on HPUX-10 operating systems 506

Appendix D. Setting up DB2 Documentation on a Web Server 509
Considerations for Using a Web Server . . 509
Working with the DB2 Universal Database Documentation Files in a Client/Server Environment 510
Searching the DB2 Books from a Remote Client 511
Typical Web Server Scenarios. 518
Scenario 1: Lotus Domino Go! Web Server on OS/2 518
Scenario 2: Netscape Enterprise Web Server on Windows NT 519
Scenario 3: Microsoft Internet Information Server on Windows NT 521
Serving Documentation in Multiple Languages 522
Serving Documentation for Multiple Platforms 523
DB2 Extender Documentation 523

Appendix E. How the DB2 Library Is Structured 525
Completing Tasks with SmartGuides . . . 525
Accessing Online Help 526
DB2 Information – Hardcopy and Online 528
Viewing Online Information 535
Accessing Information with the Information Center 536
Setting Up a Document Server 537
Searching Online Information 538
Printing the PostScript Books. 538
Ordering the Printed Books 539

Appendix F. National Language Support (NLS) 541
Language and Codeset Support for UNIX Operating Systems 541
Code Page and Language Support for OS/2 and Windows Operating Environments 542

Appendix G. Naming Rules 545
General Naming Rules 545
Database, Database Alias, and Catalog Node Name Rules 546
Object Name Rules 546
Username, User ID, Group Name, and Instance Name Rules 547
Workstation Name (nname) Rules 548
DB2SYSTEM Naming Rules 548

Password Rules 548

Appendix H. Notices 549
Trademarks 550
Trademarks of Other Companies 550

Index 553

Contacting IBM 559

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 install 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.

© Copyright IBM Corp. 1993, 1999

For a complete description of the DB2 library, see “Appendix E. How the DB2 Library Is Structured” on page 525.

- 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, or Windows NT.
- The term *Windows 9x* refers to Windows 95 or Windows 98.
- The term *DB2 client* refers to a DB2 Run-Time Client or a DB2 Administration Client.
- 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

© Copyright IBM Corp. 1993, 1999

Chapter 1. Planning for Installation

If you know that your system meets all the hardware and software requirements, and you want to begin installing your DB2 product, skip this section and proceed to the appropriate DB2 client installation section.

There are many components that you might want to use in your environment. Use the product and planning information in this section to ensure that your system meets the prerequisites and to decide which components you want to install.

Before you begin your DB2 product installation, you should determine the requirements for the system that you are planning to install and configure.

Disk Requirements

This section shows the *minimum* amount of disk space that is required to install your DB2 product and components. It does not include the disk requirements necessary for the operating system, application development tools, and communications products. Consult each product's documentation for these values. Estimates for disk space requirements are listed here; the actual amounts required depend on the functions you are using.

For information about space requirements for data, refer to the *Administration Guide*.

Estimating Fixed Disk Requirements

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

Client Components

Use Table 1 on page 4 to estimate the amount of disk space you need on each of your client workstations.

If you are installing the Control Center or the online documentation on a Windows NT FAT or a Windows 9x FAT16 disk partition, you may require additional amounts of disk space.

© Copyright IBM Corp. 1993, 1999

Table 1. Disk Requirements for Client Components

	Recommended Minimum Disk (MB)
AIX	
DB2 Run-Time Client (Base)	25 MB
- Client Configuration Assistant	5 MB
or	
DB2 Administration Client (Base)	25 MB
- DB2 Administration Tools	40 MB
- Client Configuration Assistant	6 MB
Total Disk Space Required	__ MB
HP-UX	
DB2 Run-Time Client (Base)	25 MB
- Client Configuration Assistant	5 MB
or	
DB2 Administration Client (Base)	25 MB
- DB2 Administration Tools	40 MB
- Client Configuration Assistant	6 MB
Total Disk Space Required	__ MB
Linux	
DB2 Run-Time Client (Base)	25 MB
- Client Configuration Assistant	5 MB
or	
DB2 Administration Client (Base)	25 MB
- DB2 Administration Tools	40 MB
- Client Configuration Assistant	6 MB
Total Disk Space Required	__ MB
OS/2	
DB2 Run-Time Client (Base)	11 MB
- Client Configuration Assistant	6 MB
or	
DB2 Administration Client (Base)	11 MB

Table 1. Disk Requirements for Client Components (continued)

	Recommended Minimum Disk (MB)
- DB2 Administration Tools	68 MB
- Client Configuration Assistant	6 MB
Total Disk Space Required	__ MB

Silicon Graphics IRIX

DB2 Run-Time Client (Base)	25 MB
- Client Configuration Assistant	6 MB
or	
DB2 Administration Client (Base)	25 MB
- DB2 Administration Tools	40 MB
- Client Configuration Assistant	6 MB
Total Disk Space Required	__ MB

Solaris

DB2 Run-Time Client (Base)	25 MB
- Client Configuration Assistant	6 MB
or	
DB2 Administration Client (Base)	11 MB
- DB2 Administration Tools	68 MB
- Client Configuration Assistant	6 MB
Total Disk Space Required	__ MB

Windows 32-bit Operating Systems

DB2 Run-Time Client (Base)	11 MB
- Client Configuration Assistant	6 MB
or	
DB2 Administration Client (Base)	11 MB
- DB2 Administration Tools	68 MB
- Client Configuration Assistant	6 MB
Total Disk Space Required	__ MB

Software Requirements

This section outlines the software required to run DB2 products.

Client Product Requirements

Table 2 lists the software requirements needed for a DB2 Administration Client, DB2 Run-Time Client, or a DB2 Software Developer's Kit.

Table 2. Software Requirements for Clients

Component	Hardware/Software Requirements	Communications
• DB2 Run-Time Client for AIX • DB2 Administration Client for AIX • DB2 Software Developer's Kit for AIX	RISC System/6000 and the following: • AIX Version 4.2 or later	APPC or TCP/IP • For APPC connectivity, you require IBM eNetwork Communications Server Version 5.0.2.4 or later for AIX • The AIX base operating system provides TCP/IP connectivity, if selected during install. **Note:** If you plan to use DCE (Distributed Computing Environment), you require a DCE product that is provided by the AIX Version 5 operating system with its latest DCE PTF.
• DB2 Run-Time Client for HP-UX 10.20 • DB2 Administration Client for HP-UX 10.20 • DB2 Software Developer's Kit for HP-UX	HP 9000 Series 700 or 800 system and the following: • HP-UX Version 10.20 or later The following patches are required: • For HP-UX Version 10.20 – PHSS_10556 – PHSS_10436 – PHSS_10053 – PHSS_10113 • For systems with the ANSI C or C++ compilers: – PHSS_10261 – PHSS_7505 – PHSS_9096 for C++	APPC or TCP/IP • For APPC connectivity, HP-UX Version 10.20 requires the following: – SNAplus2 and the following components: - SNAplus2 Link Version A.10.10 - SNAplus2 API Version A.10.10 **Note:** If you plan to use DCE (Distributed Computing Environment) with Version 6 of the DB2 Universal Database products, you require a DCE product that is at OSF DCE level 1.1, which is provided by the HP-UX Version 11 and later operating system. With DB2 Connect, you must install DCE Directory Services on the client and the DRDA server. You do not need DCE installed on a DB2 Connect Enterprise Edition server.

Table 2. Software Requirements for Clients (continued)

Component	Hardware/Software Requirements	Communications
• DB2 Run-Time Client for HP-UX for 11.00 • DB2 Administration Client for HP-UX for 11.00 • DB2 Software Developer's Kit for HP-UX	HP 9000 Series 700 or 800 system and the following: • HP-UX Version 11.00 or later	APPC or TCP/IP • For APPC connectivity, you require either of the following: – SNAplus2 Link R6.11.00.00 – SNAplus2 API R6.11.00.00 **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.
• DB2 Run-Time Client for Linux • DB2 Administration Client for Linux • DB2 Software Developer's Kit for Linux	• Linux kernel 2.0.35 or higher; • *glibc* Version 2.0.7 or higher; • *pdksh* package (required to run the DB2 command line processor); and • *libstdc++* Version 2.8.0 or higher. To install DB2, you will need *rpm*.	TCP/IP • The Linux base operating system provides TCP/IP connectivity, if selected during install. APPC connectivity is not provided in this release.

Table 2. Software Requirements for Clients (continued)

Component	Hardware/Software Requirements	Communications
• DB2 Run-Time Client for OS/2 • DB2 Administration Client for OS/2 • DB2 Software Developer's Kit for OS/2	• OS/2 Warp Version 3 and Version 4 • OS/2 Warp Connect Version 3 • OS/2 Warp Server Version 4 • OS/2 Warp Server Advanced V4 • OS/2 Warp Server Advanced V4 with SMP Feature • OS/2 Warp Server for e-business	APPC, IPX/SPX, NetBIOS, or TCP/IP • 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 local databases. It cannot be used to connect to host or AS/400 databases. • For NetBIOS connectivity, you require IBM NTS/2 Version 1.0, IBM eNetwork Communications Server for OS/2 Warp Version 5, IBM eNetwork Personal Communications for OS/2 Warp Version 4.2, or IBM OS/2 LAN Requester. NetBIOS can only be used to connect to local databases. It cannot be used to connect to host or AS/400 databases. • For TCP/IP connectivity, you require IBM TCP/IP Version 2.0 or later. • The OS/2 base operating system provides Named Pipes (Local) connectivity. Named Pipes is supported in DOS and WIN-OS/2 sessions. **Notes:** 1. Net.Data requires OS/2 Warp Version 3 or later and a Web server such as IBM's Internet Connection Server. 2. For DCE Cell Directory Services Support (CDS) for DB2 Clients for OS/2, you must install IBM Distributed Computing Environment Cell Directory Service client, Version 2.10, on each client workstation. 3. If you are planning to use ADSM, PTF 3 for ADSTAR Distributed Storage Manager (ADSM) Version 3 is required for an OS/2 client.
• DB2 Run-Time Client for Silicon Graphics IRIX • DB2 Administration Client for Silicon Graphics IRIX • DB2 Software Developer's Kit for Silicon Graphics IRIX	• Silicon Graphics IRIX, Version 6.x, and the following filesets: – eoe.sw.oampkg – eoe.sw.svr4net The following patches are required for Versions 6.2 and 6.3: – 2791.0 – 3778.0	TCP/IP • The Silicon Graphics IRIX base operating system provides TCP/IP connectivity.

Table 2. Software Requirements for Clients (continued)

Component	Hardware/Software Requirements	Communications
• DB2 Run-Time Client for Solaris • DB2 Administration Client for Solaris • DB2 Software Developer's Kit for Solaris	Solaris SPARC-based computer and the following: • Solaris Version 2.5.1 or later The following patches are required for Solaris version 2.5.1: • 101242 Rev. 11 or higher • 103566 Rev. 08 or higher • 103600 Rev. 13 or higher • 103640 Rev. 20 or higher The following patches are required for Solaris version 2.6: • 105568 Rev. 12 or higher • 105210 Rev. 13 or higher • 105181 Rev. 06 or higher	APPC or TCP/IP • For APPC connectivity, you require SunLink SNA 9.0 or later and the following: – 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. **Note:** If you plan to use DCE (Distributed Computing Environment), you require a DCE product with Transarc DCE Version 1.1 for Solaris 2.5 and 2.5.1, patch level 18 or higher.
• DB2 Run-Time Client for Windows 9x • DB2 Administration Client for Windows 9x • DB2 Software Developer's Kit for Windows 9x	• Windows 95 4.00.950 or later • Windows 98	IPX/SPX, Named Pipes, NetBIOS, or TCP/IP • 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 servers. • If you plan to use LDAP (Lightweight Directory Access Protocol), you require the IBM eNetwork LDAP Directory Client Version 3.1. For more information, refer to the *Administration Guide*.

Table 2. Software Requirements for Clients (continued)

Component	Hardware/Software Requirements	Communications
• DB2 Run-Time Client for Windows NT • DB2 Administration Client for Windows NT • DB2 Software Developer's Kit for Windows NT	• Windows NT Version 4.0 with Service Pack 3 or later • Windows Terminal Server (can only run the DB2 Run-Time Client)	APPC, IPX/SPX, Named Pipes, NetBIOS, or TCP/IP • The Windows NT base operating system provides NetBIOS, IPX/SPX, TCP/IP, and Named Pipes connectivity. • For APPC connectivity, you require one of the following products: – IBM eNetwork Communications Server for Windows NT V5.01 or later. – IBM eNetwork Personal Communications for Windows NT V4.2 or later. – Microsoft SNA Server Version 4 Service Pack 2 or later – Wall Data Rumba • If you plan to use DCE (Distributed Computing Environment) with Version 6 of DB2 Universal Database, you will need to ensure that if you are connecting to DB2 for OS/390 V5.1 database that it is enabled for DCE support using OS/390 DCE Base Services Version 3. • If you plan to use LDAP (Lightweight Directory Access Protocol), you require the IBM eNetwork LDAP Directory Client Version 3.1. For more information, refer to the *Administration Guide*. • If you plan to use the ADSTAR Distributed Storage Manager (ADSM) facilities for backup and restore of your databases, you require the ADSM Client Version 3 or later. • If you have the IBM Antivirus program installed on your operating system, it must be Version 3.0 or later.

Possible Client-to-Server Connectivity Scenarios

The following table shows the communication protocols that can be used when connecting a specific LAN, host or AS/400 DB2 client to a specific DB2 server or DB2 Connect 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**	**Solaris**	**Windows NT**
AS/400 V4R1	SNA	N/A	N/A	SNA	SNA	SNA
AS/400 V4R2	SNA TCP/IP	TCP/IP	TCP/IP	SNA TCP/IP	SNA TCP/IP	SNA TCP/IP
AIX	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP	APPC TCP/IP
HP-UX	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP	APPC TCP/IP
Linux	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
MVS	SNA	N/A	N/A	SNA	SNA	SNA
OS/2	APPC IPX/SPX(1),(2) TCP/IP	TCP/IP	TCP/IP	APPC IPX/SPX(1),(2) NetBIOS TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) NetBIOS TCP/IP
OS/390	SNA TCP/IP	TCP/IP	TCP/IP	SNA TCP/IP	SNA TCP/IP	SNA TCP/IP
Silicon Graphics IRIX	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
SQL/DS	SNA	N/A	N/A	SNA	SNA	SNA
Solaris	APPC TCP/IP	TCP/IP	TCP/IP	APPC TCP/IP	APPC TCP/IP	APPC TCP/IP
VSE & VM V6	SNA	N/A	N/A	SNA	SNA	SNA
VSE V6	SNA	N/A	N/A	SNA	SNA	SNA
VM V6	SNA TCP/IP	TCP/IP	TCP/IP	SNA TCP/IP	SNA TCP/IP	SNA TCP/IP
Windows 9x	TCP/IP	TCP/IP	TCP/IP	NetBIOS TCP/IP	TCP/IP	IPX/SPX(1) NPIPE NetBIOS TCP/IP
Windows NT	APPC IPX/SPX(1) TCP/IP	TCP/IP	TCP/IP	APPC IPX/SPX(1) NetBIOS TCP/IP	APPC IPX/SPX(1) TCP/IP	APPC IPX/SPX(1) NPIPE NetBIOS TCP/IP

1. Direct Addressing
2. File Server Addressing

Chapter 2. Installing DB2 Clients

This section describes how to install a DB2 Administration Client, a DB2 Run-Time Client, or a DB2 Software Developer's Kit on your workstation. For information on how to deploy this product using a distributed installation see "Chapter 24. An Introduction to Distributed Installation" on page 425. For information on how to set up a Thin Client on Windows 32-bit operating systems, see "Chapter 18. Distributed Thin Client Installation" on page 369.

DB2 Administration Clients are available for the following platforms: AIX, HP-UX, Linux, OS/2, Silicon Graphics IRIX, Solaris, Windows 9x, and Windows NT.

DB2 Run-Time Clients and DB2 Software Developer's Kits are available for the following platforms: AIX, HP-UX, Linux, OS/2, Silicon Graphics IRIX, Solaris, Windows 9x, and Windows NT.

Clients that connect to a server must have the appropriate DB2 Administration Client, DB2 Run-Time Client, or DB2 Software Developer's Kit installed.

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

Go to the section that provides installation instructions for the DB2 client that you want to install:

- "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 21

To download installation packages for all supported DB2 clients, which include pre-Version 6 DB2 clients such as DOS, Macintosh, SCO OpenServer, SCO Unixware, SINIX, and Windows 3.x, connect to the IBM DB2 Client Application Enabler Web site at http://www.software.ibm.com/data/db2/db2tech/clientpak.html.

© Copyright IBM Corp. 1993, 1999

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

This section contains the information you need to install a DB2 client or DB2 Software Developer's Kit on Windows 32-bit operating systems.

Before You Begin

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 account to perform the installation.

Windows 9x
Any valid Windows 9x user.

Windows NT
Any user account that does not belong to the *Guests* group on the machine where the account is defined.

Installation Steps

To install a DB2 Client or Software Developer's Kit, perform the following steps:

Step 1. Log on to the system with the user account that you created 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 auto-start, see the tip that follows.

© Copyright IBM Corp. 1993, 1999

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

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

b. In the **Open** field, enter the following command:

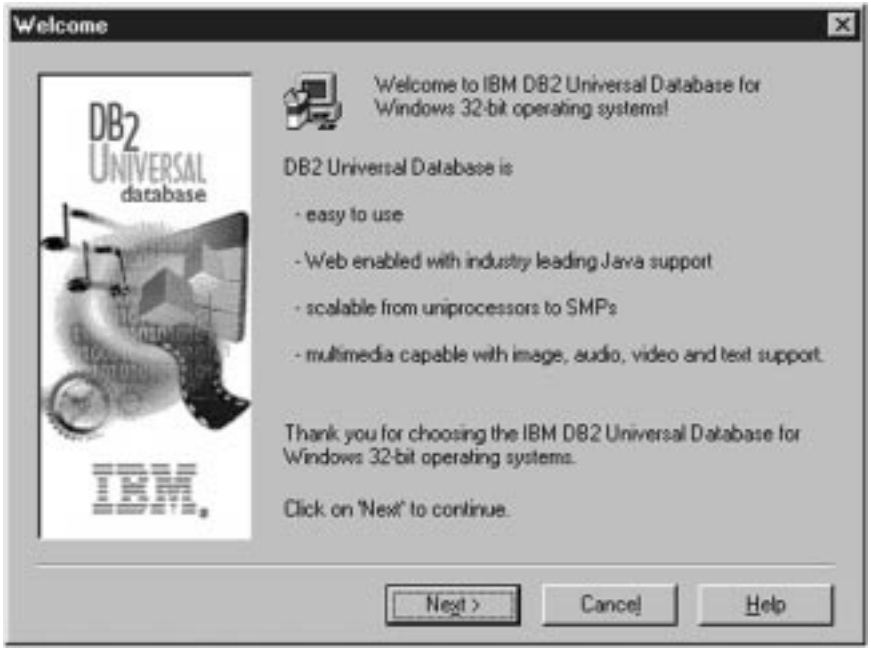
   `x:\setup /i language`

   where:

   - *x:* represents your CD-ROM drive
   - *language* represents the country code for your language (for example, EN for English). Table 36 on page 542 lists the code for each available language.

c. Click on **OK**.

Step 4. The Welcome window opens.



Step 5. Respond to the setup program's prompts. Online help is available to guide you through the remaining steps. Invoke online help by clicking on the **Help** push button, or by pressing the **F1** key at any time.

You can click on the **Cancel** push button 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 has:

- Created DB2 program groups and items (or shortcuts).
- Updated the Windows registry.
- Created 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 31.

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

This section contains the information that you need to install a DB2 client or Software Developer's Kit on OS/2 operating systems. If you have a pre-Version 6 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.software.ibm.com/data/db2/db2tech/clientpak.html.

Before You Begin

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 userid `USERID` with password `PASSWORD`.

__ 3. To verify that DB2 installed correctly, you will need to have a user account that belongs to the 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 480. For more information on valid DB2 usernames, see "Appendix G. Naming Rules" on page 545.

© Copyright IBM Corp. 1993, 1999

Installation Steps

To install a DB2 client or Software Developer's Kit 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 and set the directory to your CD-ROM drive by entering the following command:

```
x:\install
```

where *x:* represents your CD-ROM drive.

Step 3. The Welcome window opens.

Step 4. Respond to the install program's prompts. Online help is available to guide you through the remaining steps. Invoke online help by clicking on the **Help** push button, or by pressing the **F1** key at any time.

For information on errors encountered during installation, see the `l1.log` and `l2.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 31.

Chapter 5. Installing DB2 Clients on UNIX Operating Systems

This section contains the information that you need to install a DB2 client or Software Developer's Kit on UNIX operating systems.

If you want to remove a DB2 product, refer to the *Quick Beginnings for UNIX* manual.

Before You Begin

Before you begin installing a DB2 client or Software Developer's Kit using the DB2 Installer program, you need to gather the following 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 username for the default DB2 instance. We recommend that you create a new group and use it as the primary group for the DB2 instance owner.

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

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

  ```
  db2setup -d
  ```

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

Installation Steps

To install a DB2 client or Software Developer's Kit on AIX, HP-UX, Linux, Silicon Graphics IRIX, and Solaris systems, perform the following steps:

Step 1. Update Kernel Configuration Parameters

To run a DB2 client or Software Developer's Kit on a HP-UX, or Solaris system, you may have to update some kernel configuration parameters.

© Copyright IBM Corp. 1993, 1999

Go to the section that describes the kernel configuration parameters for the DB2 client that you want to install:

- "Recommended Values for HP-UX Version 10 and Version 11"
- "Recommended Values for Solaris" on page 23

You do *not* need to update any kernel configuration parameters to run a DB2 client or Software Developer's Kit on an AIX, Linux, or SGI IRIX workstation. If you are installing a DB2 client or Software Developer's Kit on AIX, Linux, or SGI IRIX, go to "Step 2. Mount the CD-ROM" on page 23.

Recommended Values for HP-UX Version 10 and Version 11

Table 4 lists the recommended values for HP-UX kernel configuration parameters. These values are valid for HP-UX 10 and HP-UX 11.

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.

Go to "Step 2. Mount the CD-ROM" to continue with the installation.

Recommended Values for Solaris

Table 5 lists the recommended values for Solaris kernel configuration parameters.

Table 5. Solaris Kernel Configuration Parameters (Recommended Values)

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

Notes:

1. Parameters msgsys:msginfo_msgmnb and msgsys:msginfo_msgmax must be set to 65535 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 changing the kernel parameters, reboot the system so that the changes can take effect.

Go to "Step 2. Mount the CD-ROM" to continue with the installation.

Step 2. Mount the CD-ROM

To install your DB2 product using the DB2 Installer program, you must first mount the CD-ROM.

Go to the section that describes the mounting instructions for a DB2 client that you want to install:

- "Mounting on AIX Systems"
- "Mounting on HP-UX Systems" on page 25
- "Mounting on Linux Systems" on page 25
- "Mounting on Silicon Graphics IRIX" on page 25
- "Mounting on Solaris Systems" on page 26

Mounting on AIX Systems

To mount the CD-ROM on AIX, perform the following steps:

Step 1. Log in as a user with `root` authority.

Step 2. Insert the CD-ROM in the drive.

Step 3. Create a directory to mount the CD-ROM by entering the following command:

```
mkdir -p /cdrom
```

where `cdrom` represents the CD-ROM mount directory.

Step 4. Allocate a CD-ROM file system by entering the following command:

```
smitty storage
```

Step 5. Select **File Systems**.

Step 6. Select **Add / Change / Show / Delete File Systems**.

Step 7. Select **CDROM File Systems**.

Step 8. Select **Add CDROM File System**.

Step 9. Select **Device Name**.

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

Step 10. In the pop-up window, enter the following as **mount point**:

```
/cdrom
```

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

```
smit mountfs
```

Step 12. Enter a value in the **FILE SYSTEM** name field. For example, the name could be `/dev/cd0`.

Step 13. Enter a value in the **Directory over which to mount** field. For example, this value could be `/cdrom`.

Step 14. Enter a value in the **Type of Filesystem** field. For example, this value could be `cdrfs`.

Step 15. Set the **Mount as READ-ONLY system** to Yes.

Step 16. Click on **OK**.

Step 17. Log out.

Go to "Step 3. Perform the Installation" on page 26 to continue with the installation.

Mounting on HP-UX Systems

To mount the CD-ROM on HP-UX, perform the following steps:

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

Step 2. Insert the CD-ROM in the drive.

Step 3. Mount it by entering the following commands:

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

where /cdrom represents the CD-ROM mount point.

Step 4. Log out.

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

Go to "Step 3. Perform the Installation" on page 26 to continue with the installation.

Mounting on Linux Systems

To mount the CD-ROM on Linux, perform the following steps:

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

Step 2. Insert the CD-ROM in the drive and mount it with a command similar to the following:

```
mount -t iso9660 -o ro /dev/cdrom /cdrom
```

where */cdrom* is the mount point of the CD-ROM.

Step 3. Log out.

Go to "Step 3. Perform the Installation" on page 26 to continue with the installation.

Mounting on Silicon Graphics IRIX

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

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

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

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

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

3. Log out.

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

Mounting on Solaris Systems

To mount the CD-ROM on Solaris, perform the following steps:

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

Step 2. If the Volume Manager is not running on your system, enter the following commands to mount the CD-ROM:

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

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

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
```

Step 3. Log out.

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

Step 3. Perform the Installation

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

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.

Step 1. Log in as user with `root` authority.

Step 2. Insert the appropriate CD-ROM into the drive.

Step 3. Change to the directory where the CD-ROM is mounted by entering the following command:

```
cd /cdrom
```

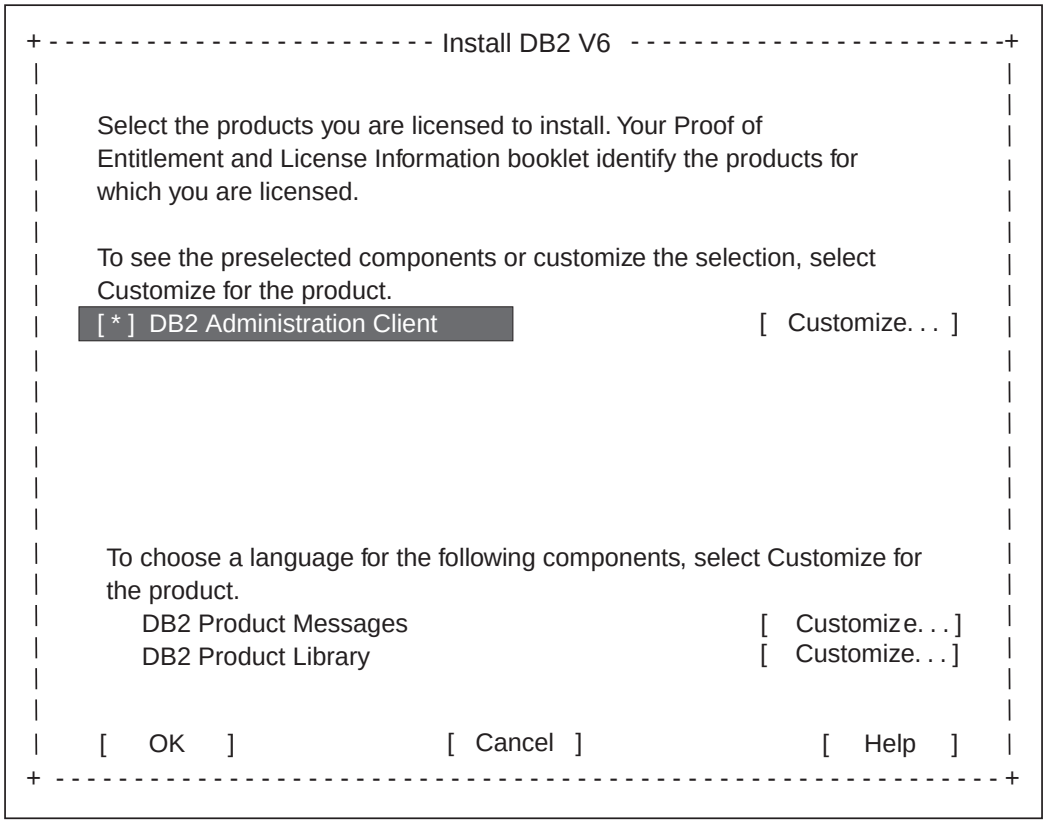
where `/cdrom` is the mount point of the CD-ROM drive.

Step 4. Change to the directory where the install image for the DB2 product that you want to install is located.

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

AIX	/cdrom/db2/aix
HP-UX Version 10	/cdrom/db2/hpux10
HP-UX Version 11	/cdrom/db2/hpux11
Linux	/cdrom/db2/linux
SGI	/cdrom/db2/sgi
Solaris	/cdrom/unnamed_cdrom/db2/solaris

Step 5. Enter the **./db2setup** command to start the DB2 Installer program. The Install DB2 V6 window opens.



Step 6. From the product list on the **Install DB2 V6** window, select the product that you want to install and select **OK**.

Press the Tab key to change the highlighted option and the Enter key to select or deselect the option you want.

To display the required and optional components for a product you want to install, select **Customize**. To go back to a previous window at any time, select **Cancel**. You can invoke the online help by clicking on **Help**.

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

This completes the installation. The DB2 software is installed in the *DB2DIR* directory,

where *DB2DIR*	= /usr/lpp/db2_06_01	on AIX
	= /opt/IBMdb2/V6.1	on HP-UX, Solaris, or SGI IRIX
	= /usr/IBMdb2/V6.1	on Linux

You can use the DB2 Installer program to create an additional instance, create an Administration Server, or add additional products or components after your initial installation. To create or add a new instance, an Administration Server, or additional DB2 products and components, enter the following command:

On AIX

```
/usr/lpp/db2_06_01/install/db2setup
```

On HP-UX, Solaris, or SGI IRIX

```
/opt/IBMdb2/V6.1/install/db2setup
```

On Linux

```
/usr/IBMdb2/V6.1/install/db2setup
```

To configure your client to access a remote DB2 server, see "Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor" on page 43.

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

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.

If you have added a new protocol to your network, or wish to modify any of the default settings, go to "Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor" on page 43.

The information in this section describes how to use the Client Configuration Assistant (CCA) to configure your OS/2, Windows 9x, or Windows NT clients to access remote servers. 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 475.

With the CCA, you can:

- Configure database connections that applications can use.
- Update or delete existing configured database connections.
- Display the information for existing configured connections.
- Test a connection to a database.
- Enable or disable databases to be configured as CLI or ODBC data sources.
- Export client profiles, or import client and server profiles which contain information for the setup of a client.
- Update client configuration settings.
- Discover remote databases (if enabled).
- Bind user applications and utilities to databases.
- Change your server password.

The CCA provides three methods to set up a database connection to a server:

- Using a profile.
- Searching the network for databases.
- Manually entering database and communication protocol information for a server.

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

© Copyright IBM Corp. 1993, 1999

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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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 475.

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 Database** or **Add** push button to configure a connection.

Go to the section that describes the configuration method that you want to use:

- "Adding a Database Using a Profile".
- "Adding a Database Using Discovery" on page 33.
- "Adding a Database Manually" on page 36.

Adding a Database Using a Profile

A profile contains information about instances and databases on a system, and databases within each instance. For information on profiles, see "Creating and Using Profiles" on page 38.

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

Step 1. Select the **Use an access profile** radio button and click on the **Next** push button.

Step 2. Click on the **...** push button and select a profile.

Step 3. Enter a local database alias name in the **Database alias** field and add a comment that describes this database in the **Comment** field. Click on the **Next** push button.

If you do not specify a database alias name, the default will be the same as the remote database alias name.

Step 4. Register this database as an ODBC data source.

If you are not planning to use ODBC, click on the **Done** push button and go to the next step.

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.

c. If you would like to create an ODBC data source file to share database access, select the **Create a file data source** check box and enter the path and filename for this file in the **File data source name** field.

d. Click on the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.

e. Click on the **Done** push button to add the database that you selected. The Confirmation window opens.

Step 5. Click on the **Test Connection** push button to test the connection. The Connect to DB2 Database window opens.

If the database that you want to add was not added successfully, click on the **Change** push button to change any settings that you may have incorrectly specified or click on the **Help** push button for more information. For even more detailed information, refer to the *Troubleshooting Guide*.

Step 6. Enter your user ID and password and click on **OK**. If the connection is successful, a message confirming the connection appears.

Step 7. You are now able to use this database. To finish using the Add Database SmartGuide, click on the **Close** push button; you can then add more databases by clicking on the **Add** push button or exit the CCA.

Adding a Database Using Discovery

This option cannot return information about pre-Version 5 DB2 systems 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 the 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 on the **Next** push button.

Step 2. Click on the **[+]** sign beside the **Known Systems** icon to list all the systems known to your client.

Step 3. Click on the **[+]** sign beside a system to get a list of the instances and databases on it. Select the database that you want to add, click on 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 on the **Next** push button, 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 on the **Add System** push button

b. Enter the required communication protocol parameters for the remote Administration Server and click on **OK**. A new system is added. For more information, click on the **Help** push button.

c. Select the database that you want to add and click on the **Next** push button.

Step 4. Enter a local database alias name in the **Database alias** field and add a comment that describes this database in the **Comment** field. Click on the **Next** push button.

If you do not specify a database alias name, the default will be the same as the remote database alias name.

Step 5. Register this database as an ODBC data source.

If you are not planning to use ODBC, click on the **Done** push button and go to the next step.

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.

c. If you would like to create an ODBC data source file to share database access, select the **Create a file data source** check box and enter the path and filename for this file in the **File data source name** field.

d. Click on the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.

e. Click on the **Done** push button to add the database that you selected. The Confirmation window opens.

Step 6. Click on the **Test Connection** push button to test the connection. The Connect to DB2 Database window opens.

If the database that you want to add was not added successfully, click on the **Change** push button to change any settings that you may have incorrectly specified or click on the **Help** push button for more information. For even more detailed information, refer to the *Troubleshooting Guide*.

Step 7. Enter your user ID and password and click on **OK**. If the connection is successful, a message confirming the connection appears.

Step 8. You are now able to use this database. To finish using the Add Database SmartGuide, click on the **Close** push button; you can then add more databases by clicking on the **Add** push button or exit the CCA.

Adding a Database Manually

If you have the protocol information for the server that you want to connect to, 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 DB2 database** radio button and click on the **Next** push button.

If you are not using Lightweight Directory Access Protocol (LDAP), this tab will not be shown, skip this step and go to Step 3.

Step 2. 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 on the **Next** push button.
- If you would like to maintain the DB2 directories globally at an LDAP server, select the **Add database using LDAP** radio button and click on the **Next** push button.

Step 3. Select the radio button that corresponds to the protocol that you want to use from the **Protocol** list and click on the **Next** push button.

If you have installed a DB2 Connect product or the DB2 Connect Server Support feature available with DB2 Enterprise Edition or DB2 Enterprise - Extended Edition, and have selected the **TCP/IP** or **APPC** radio button, select the radio button from the **Target operating system** list that corresponds to the type of system where the database that you are trying to connect to resides.

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

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. Add a comment that describes this database in the **Comment** field and click on the **Next** push button.

If you are adding a database that resides on a host or AS/400 system, the **Database name** field will appear as follows:

OS/390 or MVS
Location name

AS/400
RDB name

VM or VSE
DBNAME

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

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

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.

c. If you would like to create an ODBC data source file to share database access, select the **Create a file data source** check box and enter the path and filename for this file in the **File data source name** field.

d. Click on the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.

e. Click on the **Done** push button to add the database that you selected. The Confirmation window opens.

If you are adding a database using the *manual* method, you can also select the **Node Options** tab, the **Security Options** tab, or the **Host or AS/400 Options** tab (if you are adding a host or AS/400 database) to specify more detailed catalog information. Refer to the online help in each tab for more information.

If you want to use the default settings for the node, security, and host or AS/400 (if applicable) options, click on the **Done** push button and go to the next step.

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

If the database that you want to add was not added successfully, click on the **Change** push button to change any settings that you may have incorrectly specified or click on the **Help** push button for more information. For even more detailed information, refer to the *Troubleshooting Guide*.

Step 8. Enter your user ID and password and click on **OK**. If the connection is successful, a message confirming the connection appears.

Step 9. You are now able to use this database. To finish using the Add Database SmartGuide, click on the **Close** push button; you can then add more databases by clicking on the **Add** push button or exit the CCA.

You can use the Export feature 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 feature 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".

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 24. An Introduction to Distributed Installation" on page 425.

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. You can use either a *Server Profile* or a *Client Profile* to configure database connections on a client.

Go to the section that describes the type of profile that you want to use:

- "Server Profiles".
- "Client Profiles" on page 39.

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.

To generate a Server Profile, use the Export Server Profile function provided in the Control Center. When a profile is generated, it includes instances that have the *discover_inst* database manager configuration parameter and databases with the *discover_db* database configuration parameter set to *ENABLE*. The *discover* parameter in the Administration Server's configuration file must be set to either *SEARCH* or *KNOWN* to generate a profile for a server system.

The default settings for these configuration parameters enable you to discover all instances and databases on your network. For information on setting the *discover_inst*, *discover_db*, and *discover* configuration parameters, refer to the *Administration Guide*.

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 Control Center" on page 476.

Step 2. Select the system that you want to create a profile for and click on the right mouse button.

If the system that you want to create a profile for is not shown, select the **Systems** icon, click on the right mouse button, and select the **Add** option. Click on 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 click on the **Export** push button.

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 32.

Client Profiles

Information in a Client Profile can be used to configure clients using the Import function in the Client Configuration Assistant (CCA). These clients can import all or a subset of the configuration information in a profile.

This scenario assumes that the database connections configured on one client will be exported and used to set up one or more clients.

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, 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 475.

Step 2. Click on the **Export** push button. 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 on **OK**, and go to Step 6.
- 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 on **OK**, and go to Step 6.
- 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 on **OK**, and go to the next step.

Step 4. Select the databases to be exported from the **Available DB2 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 on 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 on the **Help** push button.

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

Step 7. Enter a path and file name for this Client Profile and click on **OK**. The DB2 Message window opens.

Step 8. Click on **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 475.

Step 2. Click on the **Import** push button. The Select Profile window opens.

Step 3. Select a Client Profile to import and click on **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 all the databases defined 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 on **OK**.

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

Step 7. Enter a local database alias name in the **Database alias** field and add a comment that describes this database in the **Comment** field. Click on the **Next** push button.

If you do not specify a database alias name, the default will be the same as the remote database alias name.

Step 8. Register this database as an ODBC data source.

If you are not planning to use ODBC, click on the **Done** push button and go to the next step.

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.

c. If you would like to create an ODBC data source file to share database access, select the **Create a file data source** check box and enter the path and filename for this file in the **File data source name** field.

d. Click on the **Optimize for application** drop down box and select the application for which you want to tune the ODBC settings.

e. Click on the **Done** push button to add the database that you selected. The Confirmation window opens.

Step 9. Click on the **Test Connection** push button to test the connection. The Connect to DB2 Database window opens.

If the database that you want to add was not added successfully, click on the **Change** push button to change any settings that you may have incorrectly specified or click on the **Help** push button for more information. For even more detailed information, refer to the *Troubleshooting Guide*.

Step 10. Enter your user ID and password and click on **OK**. If the connection is successful, a message confirming the connection appears.

Step 11. You are now able to use this database. To finish using the Add Database SmartGuide, click on the **Close** push button; you can then add more databases by clicking on the **Add** push button or exit the CCA.

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

To configure a client to communicate with a server, the remote server must be configured to accept in-bound requests for the communication protocol that you want to use. By default, the installation program automatically detects and configures any 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, refer to the *Installation and Configuration Supplement*.

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

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 31 for more information.

For instructions on entering DB2 commands, see "Entering Commands Using the Command Center" on page 477 or "Entering Commands Using the Command Line Processor" on page 478.

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".
- TCP/IP - see "Configuring TCP/IP on the Client" on page 48.
- NetBIOS - see "Configuring NetBIOS on the Client" on page 56.
- IPX/SPX - see "Configuring IPX/SPX on the Client" on page 62.
- APPC - see "Configuring APPC on the Client" on page 70.

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 6 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 10 for the supported communication protocols for your particular client and server.

© Copyright IBM Corp. 1993, 1999

To set up a client to use Named Pipe 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 6. Named Pipe Values Required at the Client

Parameter	Description	Sample Value	Your Value
Computer name (*computer_name*)	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 (*instance_name*)	The name of the instance on the server to which you are connecting.	`db2`	
Node name (*node_name*)	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 this protocol on the client. 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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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:

```
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:

```
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:

```
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. 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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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 7. Worksheet: Parameter Values for Cataloging Databases

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

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

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

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

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

```
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

When you have finished configuring the client for communications, perform the following steps to test the connection:

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. Enter the following command to connect the client to the remote database:

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

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 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.

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 (*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 6 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 10 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 8. TCP/IP Values Required at the Client

Parameter	Description	Sample Value	Your Value

Host Name
• Hostname (*hostname*) or
• IP address (*ip_address*)
	Use the *hostname* or *ip_address* of the remote server workstation.
	
	To resolve this parameter:
	• Enter the **hostname** command at the server to obtain the *hostname*.
	• Contact your network administrator to obtain the *ip_address* or enter the **ping** *hostname* command.
		`serverhost`
		
		or
		
		`9.21.15.235`	

Service Name
• Connection Service name (*svcename*) or
• Port number / Protocol (*port_number/tcp*)
	Values required in the `services` file.
	
	The Connection Service name is an arbitrary name that represents the Connection port number (*port_number*) on the client.
	
	The port number for the client must be the same as the port number that the *svcename* parameter maps to in the `services` file at the server. (The *svcename* parameter is located in the database manager configuration file on the server.) This value must not be in use by any other applications, and must be unique within the `services` file.
	
	Contact your database administrator for the values used to configure the server.
		`server1`
		
		`3700/tcp`	

Table 8. TCP/IP Values Required at the Client (continued)

Parameter	Description	Sample Value	Your Value
Node name (*node_name*)	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 this protocol on the client. Replace the sample values with your worksheet values.

A. Resolve the Server's Host Address

If your network has a name server, or you are planning to directly specify the IP address (*ip_address*) of the server, skip this step and proceed to "Step B. Update the Services File" on page 51.

The client must know the address of the server to which it is attempting to establish communications. If a name server does not exist on your network, you may directly specify a hostname that maps to the IP address (*ip_address*) of the server in the local `hosts` file. See Table 9 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 9. Location of the Local Hosts and Services Files

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

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

tcp represents the communication protocol that you are using

represents a comment describing the entry

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

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

The file called `services` is located in the same directory as the local `hosts` file that you may have edited in "A. Resolve the Server's Host Address" on page 50.

See Table 9 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 *svcename* (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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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/sqllib/db2profile    (for Bash, Bourne or Korn shell)
source INSTHOME/sqllib/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 [svcename|port_number]
terminate
```

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

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

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

```
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. 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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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 10. Worksheet: Parameter Values for Cataloging Databases

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

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

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

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

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

```
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

When you have finished configuring the client for communications, perform the following steps to test the connection:

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. Enter the following command to connect the client to the remote database:

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

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server 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 **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 `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 (*svcename*) 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 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.

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 must match, or the service name must map 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 (*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 6 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 10 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 11. NetBIOS Values Required at the Client

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

Table 11. NetBIOS Values Required at the Client (continued)

Parameter	Description	Sample Value	Your Value
Node name (*node_name*)	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 this protocol on the client. Replace the sample values with your worksheet values.

A. Record the Logical Adapter Number 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 **MPTS** icon.
  2. Click on the **Configure** push button.
  3. Select the **LAN adapters and protocols** radio button and click on the **Configure** push button.
  4. Record the Logical adapter number associated with the **IBM OS/2 NETBIOS** entry in the Current Configuration window.
  5. Click on the **Cancel** push button.
  6. Click on the **Close** push button.
  7. Click on the **Exit** push button.
- For Windows 9x:

There is no interface on Windows 9x operating systems that you can use to record the adapter number used for the NetBIOS connection. Complete the following steps to have the NetBIOS protocol default to use the adapter number *0*.

1. Click on **Start** and select **Settings->Control Panel**.
2. Double-click on the **Network** icon.
3. Select the **NetBEUI** icon from the following Network components are installed window.
4. Click on the **Properties** push button.
5. Select the **Advanced** tab.

Step 6. Select the **Set this protocol to be the default** check box.

Step 7. Click on **OK** to exit this window.

Step 8. Click on **OK**.

Step 9. Record the value *0* as the logical adapter number on your worksheet.

Step 10. Shut down and reboot your system for these changes to take effect.

- For Windows NT:

The logical adapter number (*adapter_number*) that you use *must* be associated with the **Nbf** Network Route for native NetBIOS.

1. Click on **Start** and select **Settings->Control Panel**.
2. Double-click on the **Network** icon and select the **Services** tab.
3. Select the **NetBIOS Interface** icon from the Network Services window and click on the **Properties** push button.
4. Scroll through the network routes until you find the logical adapter number associated with **Nbf** and record it on your worksheet.
5. Click on **OK**.
6. Click on the **Close** push button.

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 480.

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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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:

```
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. 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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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 12. Worksheet: Parameter Values for Cataloging Databases

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

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

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

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

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

```
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

When you have finished configuring the client for communications, perform the following steps to test the connection:

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. Enter the following command to connect the client to the remote database:

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

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server 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 **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 servers only).

__ 5. The security service was started. Enter the **net start db2ntsecserver** command (for Windows NT 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.

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 (*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 6 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 10 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 13 on page 63 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 129.

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 13 for a list of client specific IPX/SPX supported addressing methods.

Table 13. 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	*	

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 14. IPX/SPX Values Required at the Client

Parameter	Description	Sample Value	Your Value
File server name (*FILESERVER*)	**Direct Addressing**: A * value indicates that you are using Direct Addressing. **File Server Addressing:** The name of the NetWare file server where the database server instance is registered. This parameter must be entered in uppercase. Locate this parameter in the database manager configuration file on the server.	**Direct Addressing** * **File Server Addressing** NETWSRV	
DB2 server object name (*OBJECTNAME*)	**Direct Addressing**: The server's IPX/SPX internetwork address of the form: `netid.nodeid.socket#` where `netid` is 8 bytes, `nodeid` is 12 bytes, and `socket#` is 4 bytes. To resolve this parameter, enter the **db2ipxad** command at the server. See "A. Catalog the IPX/SPX Node" on page 65 for more information. **File Server Addressing:** The database manager server instance, represented as the object *OBJECTNAME* on the NetWare file server. The server's IPX/SPX internetwork address is stored and retrieved from this object. This parameter must be entered in uppercase and be unique on the NetWare file server system. Locate this parameter in the database manager configuration file on the server.	**Direct Addressing** 09212700.400011527745.879E **File Server Addressing** DB2INST1	

Table 14. IPX/SPX Values Required at the Client (continued)

Parameter	Description	Sample Value	Your Value
Node name (*node_name*)	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 this protocol on the client. Replace the sample values with your worksheet values.

A. Catalog the IPX/SPX Node

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

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

Step 1. Log on to the system with a valid DB2 user ID. For more information, see "Appendix G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/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:*\sqllib\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:

```
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. 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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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 15. Worksheet: Parameter Values for Cataloging Databases

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

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

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

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

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

```
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

When you have finished configuring the client for communications, perform the following steps to test the connection:

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. Enter the following command to connect the client to the remote database:

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

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server 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 **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 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.

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 (*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 6 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 10 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 the following servers:

- AIX
- OS/2
- Solaris
- Windows NT

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:

  Step a. Update the APPC profiles.

  Step b. Catalog the APPC or APPN Node.

  Step 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 16 on page 71 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 16. 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	NYX1GWOA	
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)	
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 16 on page 71 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 9x and Windows NT" on page 78
- "Configuring IBM eNetwork Communications Server for Windows NT" on page 82
- "Configuring IBM eNetwork Communications Server for Windows NT SNA API Client" on page 87
- "Configuring Microsoft SNA Server for Windows NT" on page 89
- "Configuring Microsoft SNA Client" on page 92
- "Configuring IBM eNetwork Communications Server for AIX" on page 93
- "Configuring Bull SNA for AIX" on page 98
- "Configuring SNAPlus2 for HP-UX" on page 101
- "Configuring SunLink SNA for Solaris" on page 105

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 Comminucation 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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 and the sections that follow.

Configuring IBM eNetwork Personal Communications for Windows 9x and Windows NT

This section describes how to configure IBM Personal Communications for Windows 95, Windows 98 and Windows NT 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 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 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 on the **Start** push button 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 9x or Windows NT 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 on the **Start** push button 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 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.

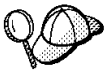
d. In the **Local Node ID** fields, type in the block ID (**13**) and the physical unit ID (**14**).

e. Click on **OK**.

Step 2. Configure the device

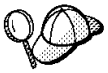
a. In the **Configuration options** box, select the **Configure devices** option and click on the **New** push button.

b. Select the appropriate DLC from **DLCs** field.



These instructions in this section use the **LAN** DLC.

c. Click on the **New** push button. The appropriate window opens with default values displayed.



In our example, 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, the select **Configure connections** option.

b. Ensure that the **LAN** option is selected in the **DLCs** field.

c. Click on the **New** push 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 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 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. 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 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. 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 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. 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 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. 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 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 dialog box 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

IBM Personal Communications 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 enter the **set appcllu=***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, perform the following steps:

a. Click on the **Start** 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 `appcllu` in the **Variable** field.

e. Type your local LU name (**11**) 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** 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, 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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 and the sections that follow.

Configuring IBM eNetwork Communications Server for Windows NT

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.

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 asssume 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 *appcllu* 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 appcllu=***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 `appcllu` 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 31.

For manual configuration instructions, and for UNIX platforms, see “Step 3. Catalog the APPC or APPN Node” on page 108 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 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 accout. 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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 and the sections that follow.

Configuring Microsoft SNA Server for Windows NT

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.

For instructions on how to configure Microsoft SNA Client for Windows NT, go to "Configuring Microsoft SNA Client" on page 92.

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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 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 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 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" on page 89.

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" on page 89.

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. Install the DB2 client.

Step 2. Click on the **Start** button and select **Programs->DB2 for Windows NT->Client Configuration Assistant**.

Step 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 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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 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**.

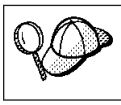
Step 3. Define a port

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 Sytems** 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 TP** box:

   Select the **Service TP (hex)** option, and enter the hexadecimal TP number (**17**)

   or:

   Select the **Application TP** field and enter the application TP name. (**17**)

e. 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.

f. Select the type of conversation level security you want to use in the **Security** group box. This is usually None.

g. Click on **OK**. The new destination name appears in the Destination names window.

h. 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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 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 93 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 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.NYX1GWOA
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_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, 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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 and the sections that follow.

Configuring SNAPlus2 for HP-UX

This section describes how to configure SNAPlus2 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 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.
- The DB2 client has been installed.
- The user is logged on as root.

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 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 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** 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 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 that 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 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** 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 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 same name in the **LU alias** field (**12**).

e. Click on **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 on the **Add** button. The Add to Node window opens.

c. Select **Define remote node**.

d. Click on **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 on **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 Sytems** 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 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:

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 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 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 on **OK**. The new destination name appears in the Destination names window.

h. Click on **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 on the **Start** push button.

e. Start the session. Select the LU you defined previously in the **Independant Local LUs** window, and click on the **Start** push button. A session activation window opens. Select or enter the Partner LU and Mode desired.

f. 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 31.

For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 108 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"

Step 2. "Create the SNA Server Configuration File" on page 106

Step 3. "Define the Environment Variables Required by SunLink SNA" on page 107

Step 4. "Start the SunLink SNA Subsystem" on page 107

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=NYX1GW0A           // 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 105

Step 4. Ensure you have created the SNA server configuration file as described in "Create the SNA Server Configuration File" on page 106.

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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 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, Windows 9x, and Windows NT, 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 G. Naming Rules" on page 545.

If you are adding a database to a system that has a DB2 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 480.

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/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/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 26 on page 248:

```
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:

```
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. 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 G. Naming Rules” on page 545.

If you are adding a database to a system that has a DB2 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 480.

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 17. Worksheet: Parameter Values for Cataloging Databases

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

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/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/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
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:

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

```
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

When you have finished configuring the client for communications, perform the following steps to test the connection:

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. Enter the following command to connect the client to the remote database:

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

The values for *userid* and *password* must be valid for the system on which they are authenticated. By default, authentication takes place on the server 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 **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 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.

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 (*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*.

Part 2. Setting Up Server Communications

© Copyright IBM Corp. 1993, 1999

Chapter 8. Using the Control Center to Configure Server Communications

If you want to set up communications using the command line processor, refer to "Chapter 9. Using the Command Line Processor to Configure Server Communications" on page 119.

To complete the steps in this section, you should be familiar with how to start the Control Center. For more information, see "Appendix A. Basic Task Knowledge" on page 475.

When you add a new protocol to your network, you must install the appropriate support on the server.

The installation program automatically detects and configures most communication protocols on your system. Use the instructions in this section to update configuration settings, or to add support for a new communications protocol.

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 on your Windows 32–bit or OS/2 workstation. On UNIX workstations, the installation program does not automatically create or configure an Administration Server. For more information, refer to the *Administration Guide*.

The Control Center is a graphical tool used to administer DB2 databases. Use the Control Center's setup communications function to configure communications on a DB2 server. The Control Center allows you to:

- Display the protocols and configuration parameters that a server instance is configured to use.
- Maintain the configured protocols:
  - You can modify the parameter values of a configured protocol.
  - You can add or delete a protocol.

    When you add support for a new protocol to the server system, the setup communications function detects and generates server instance parameter values for the new protocol; you can accept or modify them before use. When you remove support for an existing protocol from the server system, the setup communications function detects the protocol that has been removed and disables its use by the server instance.

© Copyright IBM Corp. 1993, 1999

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 31.
- Using the command line processor on the client, 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 43

Configuring DB2 Communications for Local Instances

You cannot use the Control Center to set up communications on a DB2 Universal Database Enterprise - Extended Edition server.

To configure communications for local instances, perform the following steps:

Step 1. Start the Control Center. For more information, see "Starting the Control Center" on page 476.

Step 2. Click on the **[+]** beside a system's name to get the instances folder.

Step 3. 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 Control Center" on page 476.

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 servers.
  - Click on the **Refresh** push button.
  - Select the system you want to add from the **System name** drop-down list.

  or

- Enter Administration Server information.
  - Enter the hostname or the IP address of the remote DB2 Administration Server in the **Host name** field.
  - Click on the **Retrieve** push button to obtain this system's information.

    For more information, click on the **Help** 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 9. 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 during the installation;
- You have added a communication protocol to your network since running the DB2 setup program; or
- You are using a communications protocol that could not be detected by the DB2 setup program.

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 8. Using the Control Center to Configure Server Communications" on page 115.

For instructions on entering DB2 commands, see "Entering Commands Using the Command Center" on page 477 or "Entering Commands Using the Command Line Processor" on page 478.

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, if you have:

- Deselected a detected protocol during installation;
- Added a communications protocol to your network since installation;
- Want to use a communications protocol that could not be detected by the DB2 setup program.

© Copyright IBM Corp. 1993, 1999

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:

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. 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.

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" on page 122.
- NetBIOS - see "Configuring NetBIOS on the Server" on page 125.
- IPX/SPX - see "Configuring IPX/SPX on the Server" on page 129.
- APPC - see "Configuring APPC on the Server" on page 135.

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 6 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 10 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 119, 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 6 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 10 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in “Setting the DB2COMM Registry Parameter” on page 119.

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.

Step 1. Identify and Record Parameter Values

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

Table 18. TCP/IP Values Required at the Server

Parameter	Explanation	Sample Value	Your Value
Connection Port • Connection Service name (*svcename*) • Port number / Protocol (port_number / tcp)	Values required in the services file. 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). The Connection Port must be unique within the services file.	server1 3700/tcp	
Service name (*svcename*)	The name used to update the Service name (*svcename*) parameter in the database manager configuration file at the server. This value must be the same as the Connection Service name specified in the services file.	server1	

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample values with your values and record them on your worksheet.

A. Update the Services File

The TCP/IP services file specifies the ports that server applications can listen on for client requests. You need to update the services file and specify the ports that you want the server to listen on for incoming client requests. The location of the services file depends on the operating system:

OS/2 path is specified by the ETC environment variable

UNIX /etc

Windows NT \winnt\system32\drivers\etc

Windows 9x \windows

If you are using Network Information Services (NIS) on your network (for UNIX servers only) you must update the services file located on your master server.

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 for the Connection Ports

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 480.

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. 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 6 for the communication protocol requirements for your platform. See “Possible Client-to-Server Connectivity Scenarios” on page 10 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in “Setting the DB2COMM Registry Parameter” on page 119.

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.

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 19. NetBIOS Values Required at the Server

Parameter	Description	Sample Value	Your Value
Adapter number (*adapter_number*)	The local logical adapters that will be used for the NetBIOS connection. The server uses adapter 0 if this parameter is not configured.	0	

Table 19. NetBIOS Values Required at the Server (continued)

Parameter	Description	Sample Value	Your Value
Workstation name (*nname*)	The NetBIOS name of the server workstation. *nname* is a name, chosen by the user, which must be unique among all NetBIOS server nodes in the network. If you are using DB2 Enterprise - Extended Edition, ensure that the last *4* characters are unique among all NetBIOS server nodes in the network. For more information on workstation names, see "Workstation Name (nname) Rules" on page 548	server1	

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample values with your values and record them on your worksheet.

A. Configure the NetBIOS Interface

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:

  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 on **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 has a built in checking facility that will not allow you to specify the same adapter number for different network routes. If a Network route already exists that is using the adapter number *0*, assign a different number to that route. (The valid range for adapter numbers is *0* to *255*.) This will allow you to select *0* as the adapter number that corresponds to **Nbf**. Approve the changes by clicking on **OK**.

B. Update the Database Manager Configuration File

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

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

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 480.

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:\sqllib\misc directory, where x: is 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 IPX/SPX, you must first have installed and configured communication software for both the client and server workstations. See “Software Requirements” on page 6 for the communication protocol requirements for your platform. See “Possible

Client-to-Server Connectivity Scenarios" on page 10 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in "Setting the DB2COMM Registry Parameter" on page 119.

Your server can be set up to support IPX/SPX client communications via Direct Addressing or File Server Addressing.

Direct Addressing:
The client connects to the 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 through an 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 20. For a list of the supported IPX/SPX addressing methods for DB2 clients, see Table 13 on page 63 .

Table 20. 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 9x	*	

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).

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 21. IPX/SPX Values Required at the Server

Parameter	Explanation	Our Example	Your Value
File server name (*FILESERVER*)	**Direct Addressing:** A * value indicates that you are using Direct Addressing. **File Server Addressing:** The name of the NetWare file server where the database server instance is registered. This parameter must be entered in uppercase.	**Direct Addressing** * **File Server Addressing** DB2INST1	
DB2 server object name (*OBJECTNAME*)	**Direct Addressing:** A * value indicates that you are using Direct Addressing. **File Server Addressing:** The database manager server instance represented as the object *OBJECTNAME* on the NetWare file server. The server's IPX/SPX internetwork address is stored and retrieved from this object. This parameter must be entered in uppercase and be unique on the NetWare file server system.	**Direct Addressing** * **File Server Addressing** NETWSRV	

Table 21. IPX/SPX Values Required at the Server (continued)

Parameter	Explanation	Our Example	Your Value
Socket number (*IPX_SOCKET*)	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. DB2 has registered well known sockets with Novell in the range 0x879E to 0x87A1. If you run more than 4 instances on the server machine, you must prevent socket collisions for instances 5 and up by choosing a socket number that is not *0x0000*, in the dynamic socket range *0x4000* to *0x7FFF*, or in the range *0x8000* to *0x9100* (these are well known sockets that are registered to various applications). The maximum value for this parameter is *0xFFFF*.	879E (default)	
IPX/SPX Internetwork Address	The internetwork address required when configuring a client to communicate with a server using Direct Addressing.	09212700.400011527745.879E	

The following characters are not valid in the File server (*FILESERVER*) or the DB2 server object name (*OBJECTNAME*) parameters: / \ : ; , * ?

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample values with your values and record them on your worksheet.

A. Update the Database Manager Configuration File

You must update the database manager configuration file with the File server (*FILESERVER*), DB2 server object name (*OBJECTNAME*), and Socket number (*IPX_SOCKET*) parameters.

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

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 480.

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
```

Direct Addressing

For example, if you have chosen a value of *879E* for the Socket number (*IPX_SOCKET*) parameter, use:

```
update dbm cfg using fileserver * objectname * ipx_socket 879E
db2stop
db2start
```

File Server Addressing

For example, if the name of the File server (*FILESERVER*) is NETWSRV, the DB2 server's object name (*OBJECTNAME*) is DB2INST1, and you have chosen a value of 879E for the Socket number (*IPX_SOCKET*) parameter, use:

```
update dbm cfg using fileserver NETWSRV objectname DB2INST1 \
ipx_socket 879E
db2stop
db2start
```

After the database manager is stopped and started again, view the database manager configuration file to ensure that these changes have taken effect. View the database manager configuration file by entering the following command:

```
get database manager configuration
```

If you are updating the Administration Server, you cannot change the ipx-socket parameter. The Administration Server always uses the registered port 87A2.

If you are planning to only support clients that use Direct Addressing, you will be required to enter the server's IPX/SPX internetwork address as the DB2 server object name when cataloging the node on the client.

Determine the value for the *OBJECTNAME* parameter by entering the **db2ipxad** command on the server. This command is located in the `sqllib/misc/` directory for UNIX platforms, or the `sqllib\misc\` 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

To access a remote server through APPC, you must first have installed and configured communication software on both the client and server workstations. This section describes how to set up APPC communications for inbound client connections on a DB2 Connect or DB2 Universal Database server. These connections can be from workstations running DB2 clients, host database clients, or AS/400 database clients. See "Software Requirements" on page 6 for the communication protocol requirements for your platform. See "Possible Client-to-Server Connectivity Scenarios" on page 10 for the supported communication protocols for your particular client and server.

Before completing the steps in this section, be sure that you have completed the instructions in "Setting the DB2COMM Registry Parameter" on page 119.

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.

Step 1. Identify and Record Parameter Values

Before you configure the server, fill in copies of the worksheet in Table 22.

After you fill in the *Your Value* entries, you can use the worksheet to configure APPC communications for inbound connections. 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.

Table 22. 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	

Table 22. APPC Values Required at the Server (continued)

Ref. #	Name on the Server	Sample Value	Your Value
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 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 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' or and `DB2DRDA`.

You will need to 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:

1. Network ID (1)
2. Local Control Point name (2)
3. Local LU name (4)
4. Mode name (6)
5. Transaction program name (7) or (8).

Step 2. Configure the Server

The following steps describe how to configure this protocol on the server. Replace the sample value with your value as recorded on your worksheet.

A. Update the Database Manager Configuration File

If you want to use only the default transaction programs 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 138.

The default transaction program 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 TP name in 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 transaction program 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 480.

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* 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 server's transaction program 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, use:

```
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 transaction program 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, or AIX). Set this value to the name of the instance you want to listen for these default TPs.

For example, to ensure that the instance MYINST listens for default transaction programs, enter the following command:

```
db2set 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'X'07'6DB'. The MYINST2 instance will continue to listen for the MYTP2 TP.

If this registry value is not set and multiple instances are configured to support APPC connections, then unpredictable results will occur.

B. Configure the APPC Communications Subsystem

To configure your DB2 server to accept remote clients using APPC, you need to update the APPC communications subsystem to support the transaction program name 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 V5.0.2.4"
- "Configuring IBM eNetwork Communications Server for Windows NT V5.01" on page 145
- "Configuring IBM eNetwork Communications Server for OS/2" on page 149
- "Configuring IBM eNetwork Personal Communications for Windows NT" on page 154
- "Configuring Microsoft SNA Server for Windows NT" on page 159
- "Configuring SunLink SNA for Solaris" on page 162

Configuring IBM eNetwork Communications Server for AIX V5.0.2.4: This chapter describes how to configure your IBM eNetwork Communications Server V5.0.2.4 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 Values* entries in the worksheet in Table 22 on page 135 to complete the steps below.

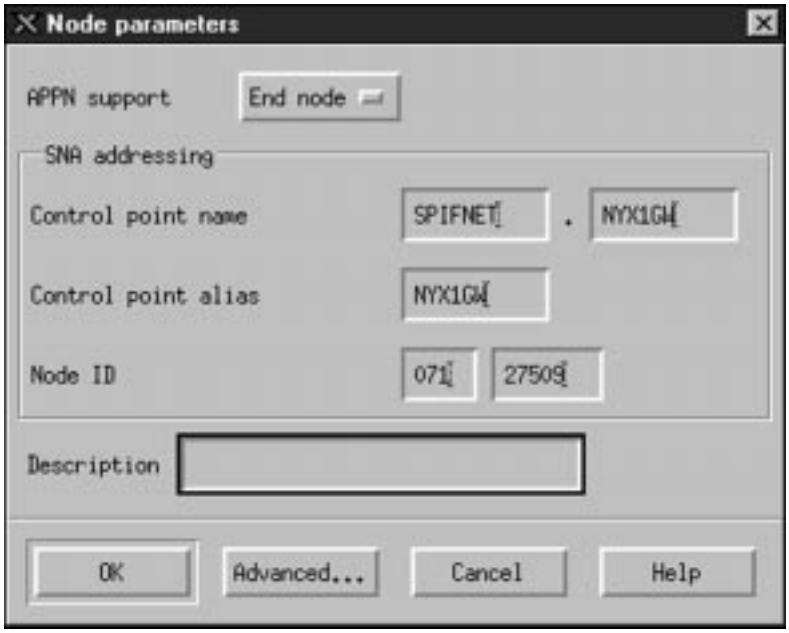
DB2 automatically configures CS/AIX with the transaction program parameters required to accept inbound APPC connections. These instructions describe the remaining steps required to configure your APPC communications for inbound connections.

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.



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.

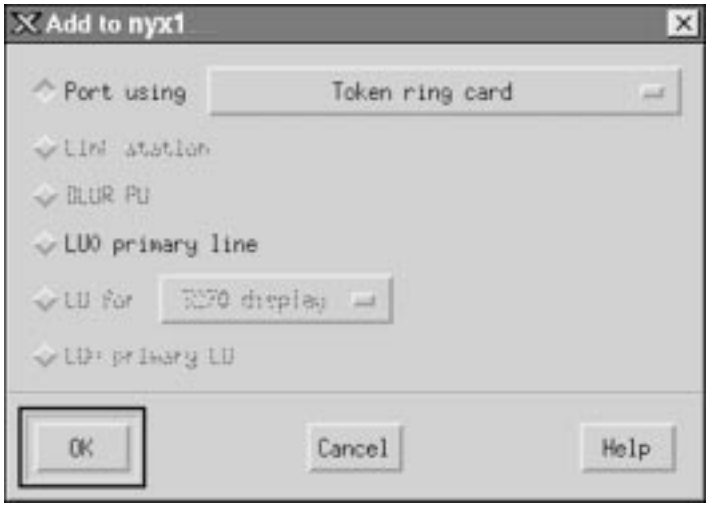
e. Enter your Node ID (**3**) in the **Node ID** fields.

f. Click 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.

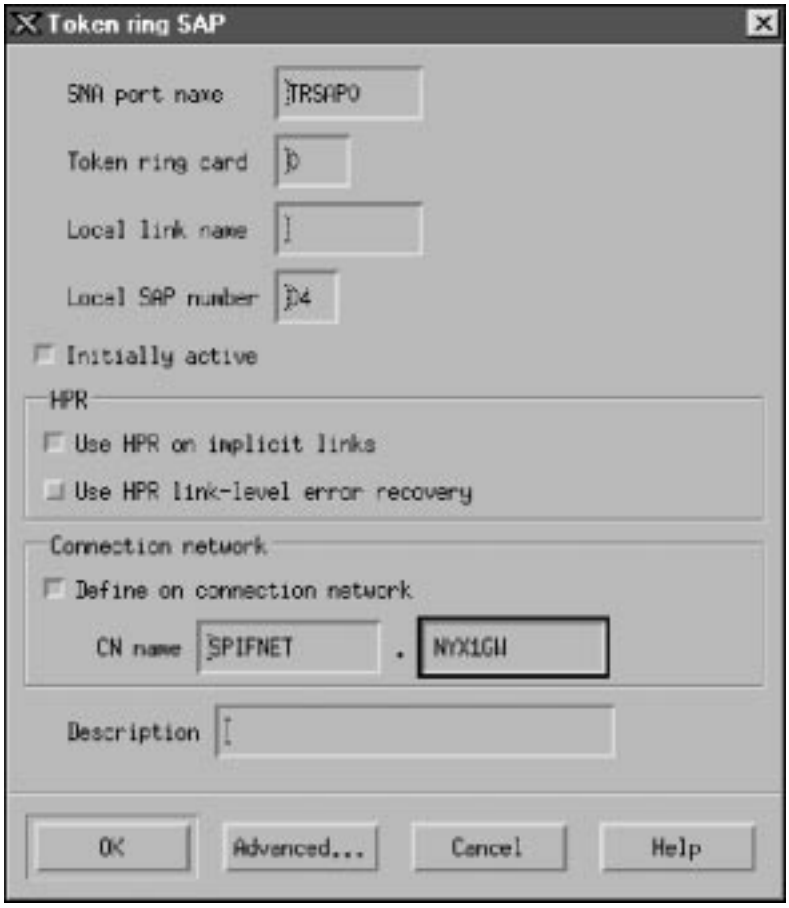


c. Select the **Port using** radio button.

d. Click on the **Port using** drop down box and select the appropriate port.

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** checkbox.

h. From the **Connection network** box, select the **Define on a connection network** checkbox.

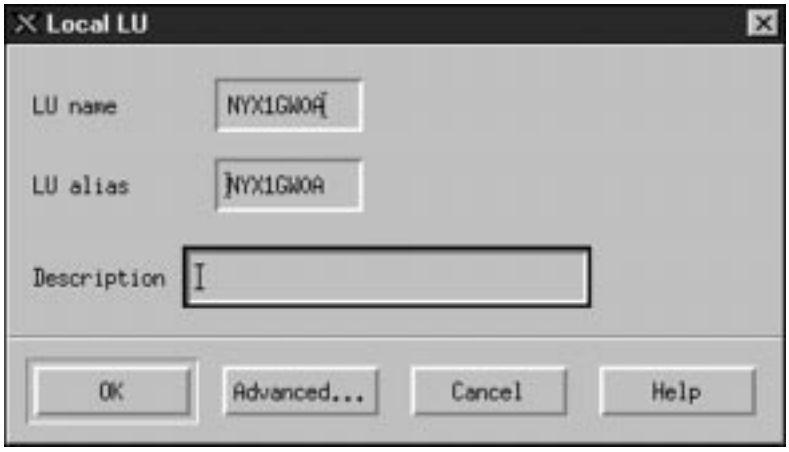
i. Enter your Network ID (**1**) and Control Point name (**2**) in the **CN name** fields.

j. Click on **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.



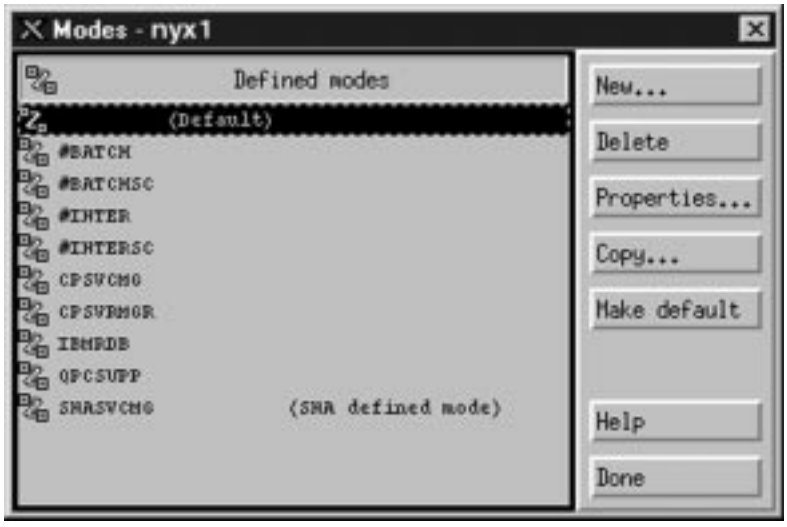
c. Enter your independent Local LU name (**4**) in the **LU name** field.

d. Enter the same name in the **LU alias** (**5**) field.

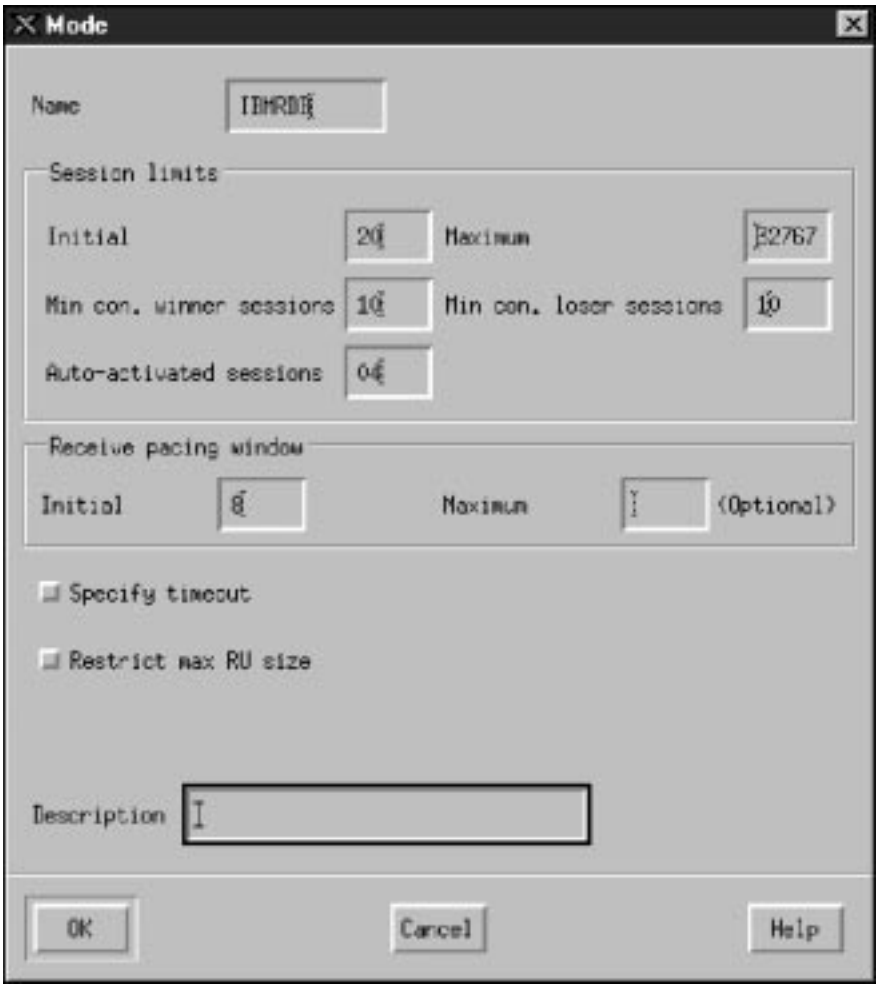
e. Click on **OK**. The new LU appears in the **Independent local LUs** window.

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.



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 session**: 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 on **OK**. The new mode appears in the Mode window.

f. Click on **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 the **Start** button.

You have now finished setting up your workstation for inbound APPC communications.

Configuring IBM eNetwork Communications Server for Windows NT V5.01: This section describes how to configure IBM eNetwork Communications Server for Windows NT V5.01 (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 (this is 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 the **Start** push button, then select **Settings->Control Panel**.

Step b. Double-click on the **Network** icon.

Step 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 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 assumptions have been made:

- The basic installation of the IBM eNetwork Communication Server V5.01 for Windows NT package has already been completed.

Use the *Your Values* entries in the worksheet in Table 22 on page 135 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 in the **CP alias** field (**2**).

d. Enter your Node ID (**3**) in the **Local Node ID** field.

e. Select the **End Node** radio button.

f. Click on **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 on **OK** to accept the default values.

Step 4. 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 NT SNA API Client, as described in "Configuring IBM eNetwork Communications Server for Windows NT SNA API Client" on page 280.

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 on **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 on **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.
e. Click on **OK**.

Step 7. 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. 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 on **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 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.

j. Click on **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 on **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* 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 **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 on **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 on **OK**.

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 the following command. Either:

```
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. Go to "Chapter 2. Installing DB2 Clients" on page 13 for more information.

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 Values* entries in the worksheet in Table 22 on page 135 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. 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 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 intructions 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 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 (**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.

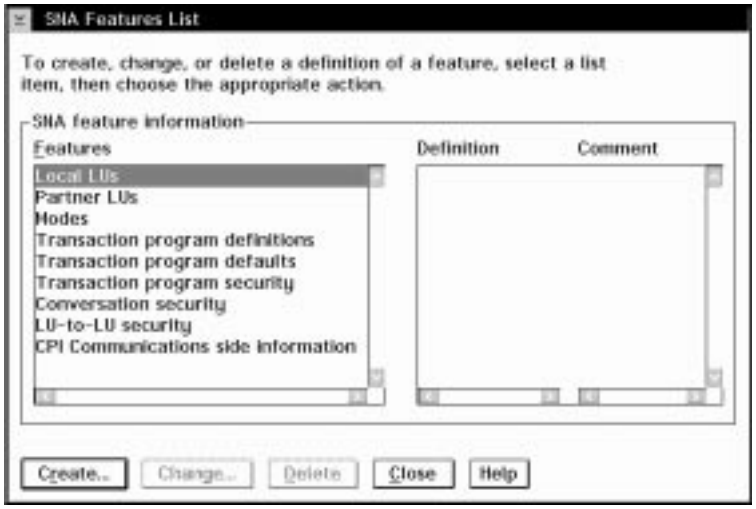
d. Enter your Node ID (**3**) in the **Local node ID (hex)** field.

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 on **OK**.

Step 5. 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 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:

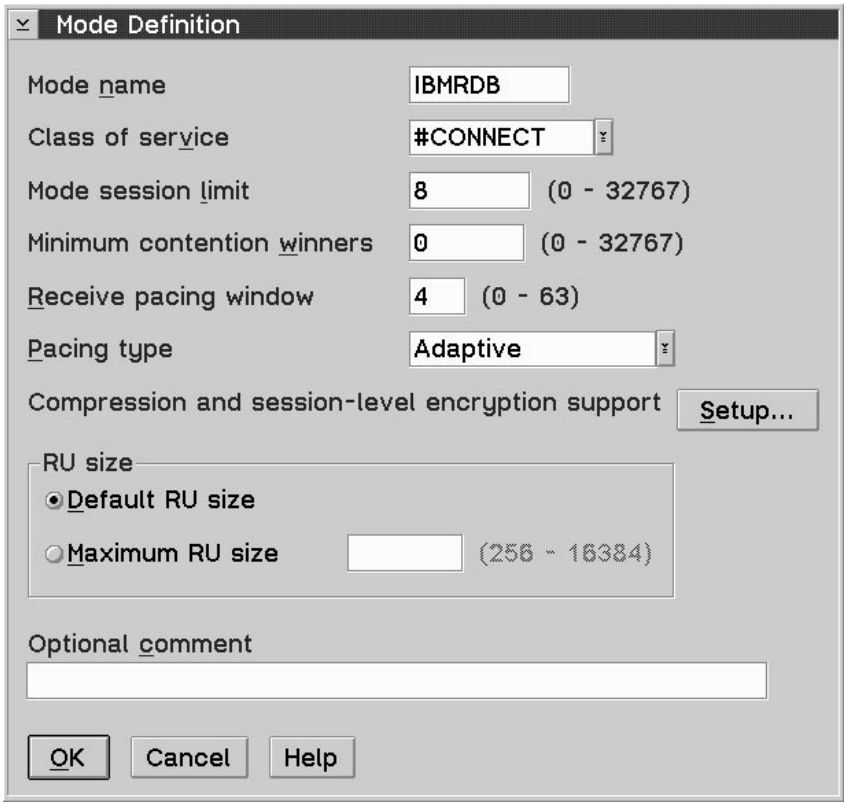
a. On the SNA Features List window, select **Local LUs->Create** from the action menu bar.

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 on **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.



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 on **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 on **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 on **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 Comminucation 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 chapter describes how to configure IBM eNetwork Personal Communications V4.30 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 the **Start** push button, then select **Settings->Control Panel**.

Step b. Double-click on the Network icon. The Network window opens.

Step c. On 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:

- *IBM DB2 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 for Windows NT has been installed.

Use the *Your Values* entries in the worksheet in Table 22 on page 135 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 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 **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 on **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 on **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. Accept the defaults for the other fields.
d. 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.
h. 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 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 on **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 on **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 **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 typing `set appcllu=`*`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, complete 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 `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.

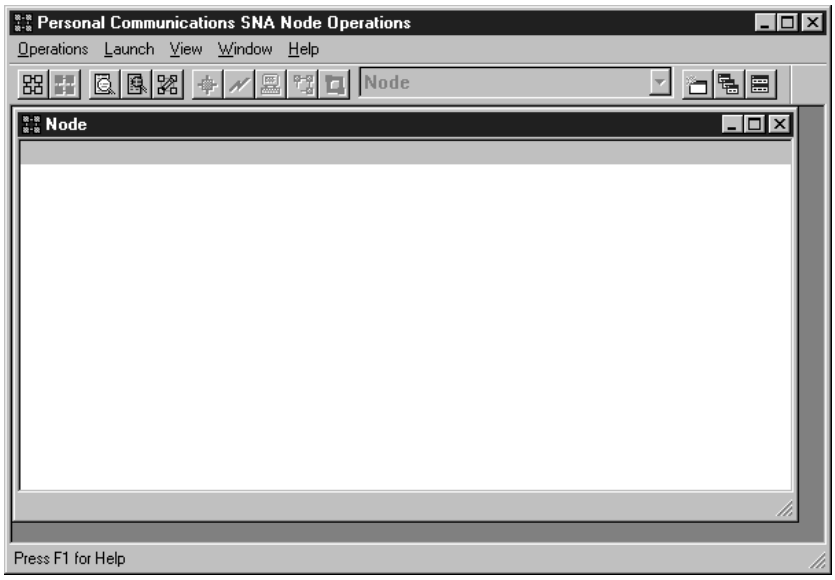
g. Click on **OK** to exit the System Properties window.

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 on **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.

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 for Windows NT has been installed.

Use the *Your Values* entries in the worksheet in Table 22 on page 135 to complete the steps below.

The transaction program name required for inbound APPC connections is automatically defined for the worksation 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 Explorer. Figure 1 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.

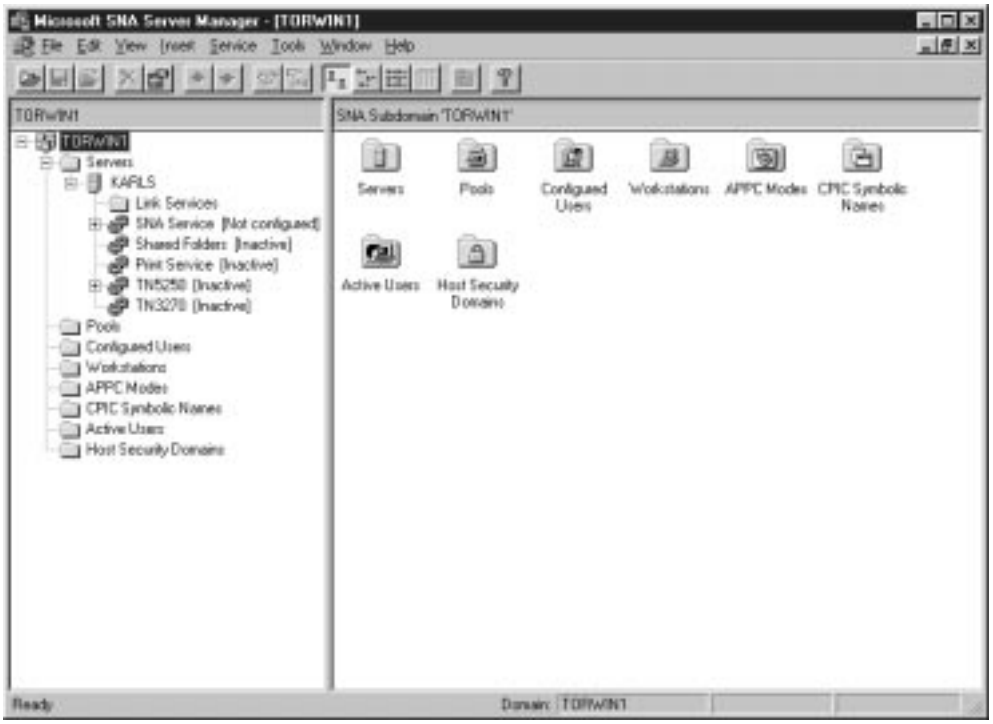


Figure 1. The Microsoft SNA Server Manager.

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 the **Start** button 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 on **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:

1) The **LU alias** (**5**).

2) The **NETID**(**1**).

3) The **LU name** (**4**).

c. Select the **Advanced** tab.

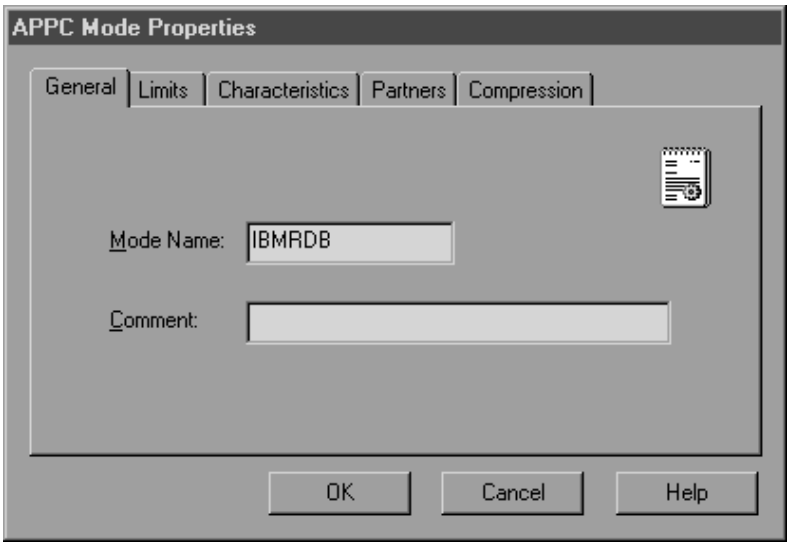
d. Select the **Member of Default Outgoing Local APPC LU Pool** option. Accept the other defaults.

e. Click on **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 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 on **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 chapter 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 theSunLink 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 Values* entries in the worksheet in Table 22 on page 135 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 it must be placed in `/opt/SUNWpu21`, or the directory where SunLink SNA PU 2.1 Server is installed. You can use any plain text editor to create the configuration file.

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
        TERMID=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:

```
kill -9 sunpu2.pid
```

f. Start SunLink using 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

© Copyright IBM Corp. 1993, 1999

Chapter 10. Running Your Own Applications

Various types of applications can access DB2 databases:

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

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

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

1. Ensure the server is configured and running.

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

Binding Database Utilities

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

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

© Copyright IBM Corp. 1993, 1999

The bind files are grouped together in different .lst files in the bnd directory, under the installation directory (typically sqllib\win on Windows 3.x systems and sqllib 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 on the **Bind** push button.
  Step 4. Select the **Bind DB2 Utilities** radio button.
  Step 5. Click on the **Continue** push button.
  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.
  Step 7. Select the utilities you want to bind and click on **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/sqllib/bnd

  For all other platforms
  x:\sqllib\bnd, where x: is the drive where you installed DB2

  Step 2. Connect to the database using the command:

  ```
  connect to database_alias
  ```

  where *database_alias* is the name of the database to which you want to connect.

  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 Software Developer's Kit (DB2 SDK) 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 169 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. (Your local system is represented by **Local** icon.) If you do not have access to this program you can use the **catalog** command in the command line processor.

Step 2. On all platforms other than OS/2 the DB2 CLI/ODBC driver is an optional component during the DB2 client install. 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.

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 179
- "How to Set CLI/ODBC Configuration Keywords" on page 180
- "Configuring db2cli.ini" on page 180

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 175
- "UNIX Client Access to DB2 using CLI/ODBC" on page 177

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 on-line 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.

To verify that they both exist on the machine:

a. Run the Microsoft ODBC Administrator from the icon in the Control Panel, or issue the appropriate command from the command line: **odbcad32.exe.**

b. Click on the **ODBC 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 190.

   For information on manually editing this file (`db2cli.ini`), see "Configuring db2cli.ini" on page 180.

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 179:

- "How to Bind the DB2 CLI/ODBC Driver to the Database" on page 179
- "How to Set CLI/ODBC Configuration Keywords" on page 180
- "Configuring db2cli.ini" on page 180

OS/2 Client Access to DB2 using CLI/ODBC

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

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

   For more information see the on-line help in the CCA (or 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 190.

      For information on manually editing this file (`db2cli.ini`), see "Configuring db2cli.ini" on page 180.

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 179:

- "How to Bind the DB2 CLI/ODBC Driver to the Database" on page 179
- "How to Set CLI/ODBC Configuration Keywords" on page 180
- "Configuring db2cli.ini" on page 180

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 43 or the **CATALOG DATABASE** and **CATALOG NODE** commands in the *Command Reference*.

2. The DB2 CLI/ODBC driver is an optional component 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 (db2.o). For example, if the home directory of your end user is /u/thisuser/ and the sqllib directory is installed there, then the correct entry would be:

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

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:

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

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

- in the [ODBC] stanza:

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

  Indicates that /u/thisuser/sqllib/odbclib 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 the sample file in the sqllib/odbclib subdirectory for an example. You can also see "How to Configure ODBC.INI" on page 180 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 180. 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" on page 180
- "Configuring db2cli.ini" on page 180

Detailed Configuration Information

The section "Platform Specific Details for CLI/ODBC Access" on page 173 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" on page 180
- "Configuring db2cli.ini" on page 180

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 169 for more information.

How to Set CLI/ODBC Configuration Keywords

DB2 CLI can be configured further by using either the CCA or the DB2 Client Setup administration tool, whichever is applicable for your platform, or by manually editing the `db2cli.ini` file.

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

By default, the location of the CLI/ODBC configuration keyword file is located in the `sqllib` directory on Intel platforms, and in the `sqllib/cfg` directory of the database instance running the CLI/ODBC applications on UNIX platforms.

The environment variable *DB2CLIINIPATH* can also be used to override the default and specify a different location for the file.

The configuration keywords enable you to:

- Configure general features such as data source name, user name, and password.
- Set options that will affect performance.
- Indicate query parameters such as wild card characters.
- Set patches or work-arounds for various ODBC applications.
- Set other, more specific features associated with the connection, such as code pages and IBM Graphic data types.

For a complete description of all the keywords and their usage, refer to “Configuration Keyword Descriptions” on page 190.

Configuring db2cli.ini: The `db2cli.ini` initialization file is an ASCII file which stores values for the DB2 CLI configuration options. A sample file is shipped to help you get started. Refer to the *CLI Guide and Reference* for information on each keyword.

See “Platform Specific Details for CLI/ODBC Access” on page 173 for more information on how to modify this file on your platform.

How to Configure ODBC.INI

Microsoft’s 16-bit ODBC Driver Manager and all non-Microsoft ODBC Driver Managers use the `odbc.ini` file to record information about the available

drivers and data sources. ODBC Driver Managers on UNIX platforms also uses the odbcinst.ini file. Although the necessary files are updated automatically by the tools on most platforms, users of ODBC on UNIX platforms will have to edit them manually. The file odbc.ini (and odbcinst.ini where required) are located:

UNIX Home directory of the user ID running the ODBC application (on UNIX the odbc.ini file name has a dot before it: .odbc.ini)

It is also possible to modify this file manually. Do not change any of the existing entries in the file. To edit this file manually perform the following steps:

Step 1. Use an ASCII editor to edit the odbc.ini file.

The following is an example odbc.ini file:

```
[ODBC Data Sources]
MS Access Databases=Access Data (*.mdb)

[MS Access Databases]
Driver=D:\WINDOWS\SYSTEM\simba.dll
FileType=RedISAM
SingleUser=False
UseSystemDB=False
```

The [ODBC Data Sources] section lists the name of each available data source and the description of the associated driver.

For each data source listed in the [ODBC Data Sources] section, there is a section that lists additional information about that data source. These are called the *Data Source Specification* sections.

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

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

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

- 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 SDK 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 SDK 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 see *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 Page at:

`http://www.software.ibm.com/data/db2/java`

Configuring the Environment

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

AIX: The Java Development Kit (JDK) Version 1.1.2 for AIX from IBM.

HP-UX:
The HP-UX Developer's Kit for Java Release 1.1.3 or later from Hewlett-Packard.

Linux: The Java Development Kit (JDK) 1.1.7, Version 1a or later for Linux from the Blackdown Organization.

OS/2: The Java Development Kit (JDK) Version 1.1.4 for OS/2 from IBM.

Silicon Graphics IRIX:
The Java Development Environment 3.1 (Sun JDK 1.1.5) and the Java Execution Environment 3.1 (Sun JRE 1.1.5) from Silicon Graphics, Inc.

Solaris:
The Java Development Kit (JDK) Version 1.1.4 for Solaris, and the Solaris Native Thread pack, from Sun Microsystems.

Windows 32-bit operating systems
The Java Development Kit (JDK) Version 1.1 for Win32 from Sun Microsystems.

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

```
http://www.software.ibm.com/data/db2/java
```

For all supported platforms, you must also install and configure 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:

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

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

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

```
db2 get dbm cfg
```

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

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

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

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

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

To run Java programs, the following environment variables are automatically updated during DB2 installation on 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, Silicon Graphics IRIX, and Solaris: LD_LIBRARY_PATH includes the directory sqllib/lib; on HP-UX: SHLIB_PATH includes the directory sqllib/lib
- On Solaris only: THREADS_FLAG is set to ″native″

On Windows and OS/2 platforms:

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

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

On UNIX platforms:

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

On Windows and OS/2 platforms:

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

Java Applications

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

```
java prog_name
```

where prog_name is the name of the program.

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

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 11. 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 173.

Configuration Keywords by Category

CLI/ODBC Settings General Configuration Keywords

- “DBALIAS” on page 202
- “PWD” on page 218
- “UID” on page 228

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 204
- “DISABLEMULTITHREAD” on page 205
- “EARLYCLOSE” on page 206

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 191
- “GRAPHIC” on page 208
- “LOBMAXCOLUMNSIZE” on page 212
- “LONGDATACOMPAT” on page 212

© Copyright IBM Corp. 1993, 1999

Enterprise Configuration Keywords

The **Enterprise** set of options are used to maximize the efficiency of connections to large databases.

- "CLISCHEMA" on page 192
- "CONNECTNODE" on page 193
- "CURRENTPACKAGESET" on page 195
- "CURRENTSCHEMA" on page 196
- "CURRENTSQLID" on page 197
- "DB2CONNECTVERSION" on page 198
- "DBNAME" on page 203
- "GRANTEELIST" on page 207
- "GRANTORLIST" on page 207
- "SCHEMALIST" on page 219
- "SYSSCHEMA" on page 221
- "TABLETYPE" on page 222

Environment Configuration Keywords

The **Environment** set of options are used to define the location of various files on the server and client machines.

- "CURRENTFUNCTIONPATH" on page 194
- "DEFAULTPROCLIBRARY" on page 204
- "TEMPDIR" on page 223

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 196
- "DB2DEGREE" on page 199
- "DB2ESTIMATE" on page 199
- "DB2EXPLAIN" on page 200
- "DB2OPTIMIZATION" on page 201
- "KEEPSTATEMENT" on page 211

- "OPTIMIZEFORNROWS" on page 215
- "OPTIMIZESQLCOLUMNS" on page 216
- "UNDERSCORE" on page 229

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 190
- "IGNOREWARNINGS" on page 209
- "IGNOREWARNLIST" on page 210
- "PATCH1" on page 216
- "PATCH2" on page 217
- "POPUPMESSAGE" on page 218
- "SQLSTATEFILTER" on page 220
- "TRACE" on page 224
- "TRACECOMM" on page 225
- "TRACEFILENAME" on page 225
- "TRACEFLUSH" on page 226
- "TRACEPATHNAME" on page 227
- "WARNINGLIST" on page 230

Transaction Configuration Keywords

The **Transaction** set of options are used to control and speed up SQL statements used in the application.

- "ASYNCENABLE" on page 190
- "CONNECTTYPE" on page 193
- "CURSORHOLD" on page 197
- "KEEPCONNECT" on page 210
- "MAXCONN" on page 213
- "MODE" on page 214
- "MULTICONNECT" on page 214

- "SYNCPOINT" on page 220
- "TXNISOLATION" on page 228

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.

ASYNCENABLE

Keyword Description:
Enable or disable the ability to execute queries asynchronously.

db2cli.ini Keyword Syntax:
ASYNCENABLE = **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.

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:

"SYSSCHEMA" on page 221

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 CLISCHEMA 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 SYSSCHEMA keyword, CLISCHEMA should be used instead (where applicable).

CLISCHEMA improves data access efficiency: The user-defined tables used with SYSSCHEMA 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 CLISCHEMA 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 220

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.

Only Applicable when:
connecting to a DB2 for MVS/ESA v4.1 or later database.

Usage Notes:

This option will issue the command ″SET CURRENT PACKAGESET schema″ after every connect to a DB2 for MVS/ESA v4.1 or later 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).
- 99999999999999 - 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 99999999999999.

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.

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 219, "TABLETYPE" on page 222

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"

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 207

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.

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 230, "IGNOREWARNLIST" on page 210

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 230, "IGNOREWARNINGS" on page 209

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*.

OPTIMIZESQLCOLUMNS

Keyword Description:
Optimize SQLColumns() call with explicit Schema and Table Name.

db2cli.ini Keyword Syntax:
OPTIMIZESQLCOLUMNS = **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 OPTIMIZESQLCOLUMNS 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 OPTIMIZESQLCOLUMNS 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 217

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 216

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 220

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

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.

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.

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 218

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' "
```

SYNCPOINT

Keyword Description:
How commits and rollbacks are coordinated among multiple database (DUOW) connections.

db2cli.ini Keyword Syntax:
SYNCPOINT = **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 193

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 = *sysschema*

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

DB2 Universal Database	DB2 for MVS/ESA	DB2 for VSE & VM	OS/400	DB2 Universal Database for AS/400
SYSINDEXES	SYSINDEXES	SYSINDEXES	SYSINDEXES	SYSINDEXES
SYSTABAUTH	SYSTABAUTH	SYSTABAUTH		SYSCST
SYSRELS	SYSRELS	SYSKEYCOLS		SYSKEYCST
SYSDATATYPES	SYSSYNONYMS	SYSSYNONYMS		SYSCSTCOL
SYSPROCEDURES	SYSKEYS	SYSKEYS		SYSKEYS
SYSPROCPARMS	SYSCOLAUTH	SYSCOLAUTH		SYSREFCST
	SYSFOREIGNKEYS			
	SYSPROCEDURES **1**			
	SYSDATABASE			

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 225, "TRACEFLUSH" on page 226, "TRACEPATHNAME" on page 227

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 option is turned on.

See Also:
"TRACE" on page 224, "TRACEFILENAME", "TRACEPATHNAME" on page 227, "TRACEFLUSH" on page 226

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 option 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 option is turned on.

See Also:

"TRACE" on page 224, "TRACEFLUSH", "TRACEPATHNAME" on page 227

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 option is turned on.

See Also:

"TRACE" on page 224, "TRACEFILENAME" on page 225, "TRACEPATHNAME" on page 227

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 224, "TRACEFILENAME" on page 225, "TRACEFLUSH" on page 226

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 210, “IGNOREWARNINGS” on page 209

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

© Copyright IBM Corp. 1993, 1999

Chapter 12. 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* manual for more information.

For instructions on entering DB2 commands, see "Entering Commands Using the Command Center" on page 477 or "Entering Commands Using the Command Line Processor" on page 478.

If you need to manually configure communications, go to the section that describes your communication protocol.

- TCP/IP - see "Chapter 13. Configuring TCP/IP Communications on the DB2 Connect Workstation" on page 235
- APPC - see "Chapter 14. Configuring APPC Communications on the DB2 Connect Workstation" on page 247.

© Copyright IBM Corp. 1993, 1999

Chapter 13. Configuring TCP/IP Communications on the DB2 Connect Workstation

This section assumes that TCP/IP is functional on the DB2 Connect and host systems.

You will only need to refer to the instructions in this section if you wish to configure your TCP/IP connection to a host database by hand. TCP/IP can normally be configured automatically using the Client Configuration Assistant (CCA). 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* manual.

To set up TCP/IP communications between your DB2 Connect workstation and a host database server, perform the following steps:

Step 1. Identify and record parameter values.

Step 2. Configure the DB2 Connect workstation:

a. Resolve the host's IP address.

b. Update the `services` file.

Step 3. Catalog the TCP/IP node.

Step 4. Catalog the database.

Step 5. Catalog the database as a Database Connection Service (DCS) database.

Step 6. Bind utilities and applications to the database server.

Step 7. Test the host connection.

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.

© Copyright IBM Corp. 1993, 1999

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 23. TCP/IP Values Required at the DB2 Connect Workstation

Parameter	Description	Sample Value	Your Value
Host Name • Hostname (*hostname*) or • IP address (*ip_address*)	Use the *hostname* or *ip_address* of the remote host. To resolve this parameter: • Contact your network administrator to obtain the *hostname*. • Contact your network administrator to obtain the *ip_address* or enter the **ping** *hostname* command.	nyx or 9.21.15.235	

Table 23. TCP/IP Values Required at the DB2 Connect Workstation (continued)

Parameter	Description	Sample Value	Your Value

Service Name
• Connection Service name (*svcename*) or
• Port number / Protocol (*port_number/tcp*)
	Values required in the `services` file.
	
	The Connection Service name is an arbitrary name that represents the Connection port number (*port_number*) on the client.
	
	The port number for the DB2 Connect workstation must be the same as the port number that the *svcename* parameter maps to in the `services` file at the host database server. (The *svcename* 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 23. TCP/IP Values Required at the DB2 Connect Workstation (continued)

Parameter	Description	Sample Value	Your Value
Target database name (*target_dbname*)	The database name as it is known on the host or AS/400 system. • 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 a DB2 for VM or DB2 for VSE system, use the `dbname`.	newyork	
Local database name (*local_dcsname*)	An arbitrary local nickname for use by DB2 Connect that represents the remote host or AS/400 database.	ny	
Node name (*node_name*)	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 DB2 Connect Workstation

The following steps configure this protocol 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 239.

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 9 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 24. Location of the Local Hosts and Services Files

Platform	Location
Windows 9x	windows directory
Windows NT	winnt\system32\drivers\etc directory
UNIX	/etc directory
OS/2	Specified by the *etc* environment variable. Enter the **set etc** command to determine the location of your local hosts or services files. **Note:** For DOS and WIN-OS2 sessions, you might need to update the hosts and services files located in the tcpip_product\dos\etc directory.
Windows 3.x	Typically in the tcpip_product\etc directory, but it depends on the products that you have installed. Refer to your TCP/IP documentation for more information.

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 "Step 3. Catalog the TCP/IP Node" on page 240.

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 238. See Table 9 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.

Step 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 *svcename* (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/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/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 [svcename|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:

```
uncatalog node node_name
```

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

Step 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* is the local name of the host or AS/400 database.
- *target_dbname* is 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
```

Step 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 25. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (*database_name*)	The local DCS database name (*local_dcsname*) of the *remote* database. You specified this when you cataloged the DCS database directory. For example, *ny*.	`ny`	
Database alias (*database_alias*)	An arbitrary local nickname for the remote database. If you do not provide one, the default is the same as the database name (*database_name*). This is the name that you use when connecting to the database from a client.	`localny`	
Node name (node_name)	The name of the node directory entry that describes where the database resides. Use the same value for node name (*node_name*) that you used to catalog the node in the previous step.	`db2node`	

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/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/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
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
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 in the command line processor as follows:

```
uncatalog database database_alias
```

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

Step 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.

To bind the utilities and applications to the host or AS/400 database server, enter the following commands:

```
db2 connect to dbalias user userid using password
db2 "bind path@ddcsmvs.lst blocking all sqlerror continue
     messages mvs.msg grant public"
db2 connect reset
```

For example:

```
db2 connect to NYC3 user myuserid using mypassword
db2 "bind /sqllib/myapps@ddcsmvs.lst blocking all sqlerror continue
     messages mvs.msg grant public"
db2 connect reset
```

The *userid* and *password* specified must have the authority to bind applications against the target database.

Further information about these commands can be found in the *DB2 Connect User's Guide*.

Step 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:

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 in the Command Center or command line processor:

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

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

You are 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 (*svcename*) parameter was updated correctly in the database manager configuration file.

__ 4. The database was created and cataloged properly.

__ 5. The database manager was stopped and started again (enter the **db2stop** and **db2start** commands on the server).

__ 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*) that 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 14. Configuring APPC Communications on the DB2 Connect Workstation

This section describes how to configure a DB2 Connect workstation to communicate with a host or AS/400 database server using the APPC communication protocol. The instructions that follow 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 configure your APPC connection to a host or AS/400 database by hand. APPC can often be configured automatically using the Client Configuration Assistant (CCA). 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* manual.

The following steps are required to set up a DB2 Connect workstation to use APPC communications with a host or AS/400 database server:

Step 1. Identify and record parameter values.

Step 2. Update the APPC profiles on the DB2 Connect workstation.

Step 3. Catalog the database.

Step 4. Catalog the APPC or APPN Node.

Step 5. Catalog the database as a Database Connection Service (DCS) database.

Step 6. Bind utilities and applications to the database server.

Step 7. Test the Host or AS/400 Connection.

Step 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 26 on page 248 for *each* host or AS/400 database to which you want to connect.

After you fill in the *Your Value* entries, 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.

© Copyright IBM Corp. 1993, 1999

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.

In the configuration instructions, the * symbol denotes entries that need to be changed but do not have a representation on the worksheet.

Table 26. 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	
5	Target database name (*target_dbname*)	**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	

Table 26. 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
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'07F6C4C2' or DB2DRDA **VM/VSE:** AXE for VSE, or the DB2 for VM db name for VM **AS/400:** X'07F6C4C2' or QCNTEDDM	
DB2 Directory Entries (at the DB2 Connect workstation)				
19	Node name		db2node	
19	Security		program	
20	Local database name (*local_dcsname*)		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 *RDB_NAME* for VM or VSE, or a relational database 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**), it is recommended that you use the defaults listed in the worksheet.
13. Leave the other items blank for now (**18** to **21**).

Step 2. Update the APPC Profiles on the DB2 Connect Workstation

Use the completed worksheet in Table 26 on page 248 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 NT" on page 274
- "Configuring IBM eNetwork Communications Server for Windows NT SNA API Client" on page 280
- "Configuring Microsoft SNA Server for Windows NT" on page 283
- "Configuring Microsoft SNA Client" on page 292
- "Configuring IBM eNetwork Communication Server for AIX" on page 295
- "Configuring Bull SNA for AIX" on page 308
- "Configuring SNAPlus2 for HP-UX" on page 311
- "Configuring SunLink SNA for Solaris" on page 324

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 Values* entries in the worksheet in Table 22 on page 135 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

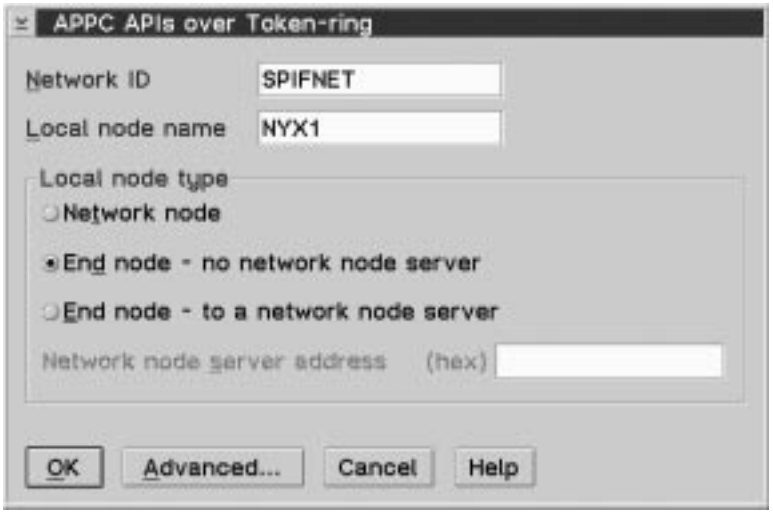
a. Select the **Commonly used definitions** radio button.

b. In the Communications Definitions box, select the protocol that you want to use.

These intructions use APPC APIs over Token-Ring.

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 (**9**) in the **Network ID** field.

e. Enter your local Control Point name (**10**) in the **Control point name** field.

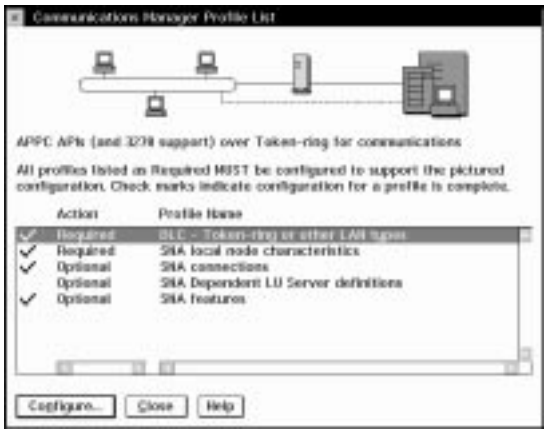
f. Click on the **End node** push button that your network administrator advises you to use.

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.

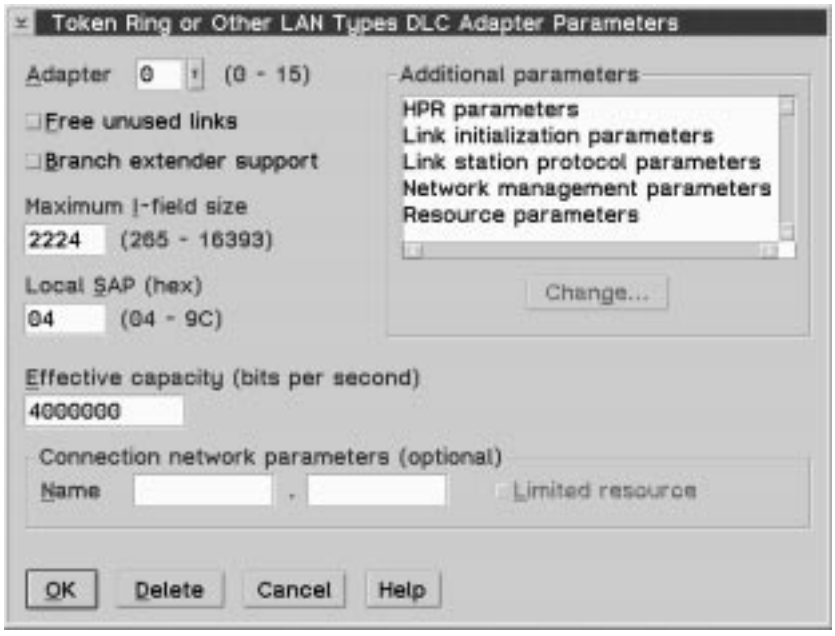
Subsequent steps begin from this window. You will return to this window when each of the following steps are completed.

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.

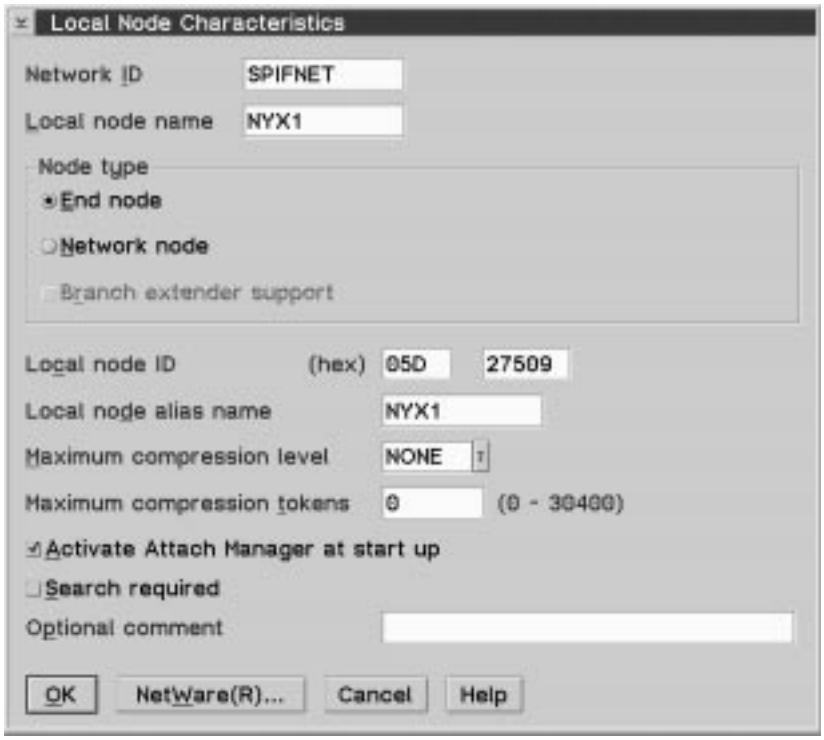


b. Enter your Network ID (**9**) in the **Network ID** field.

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.

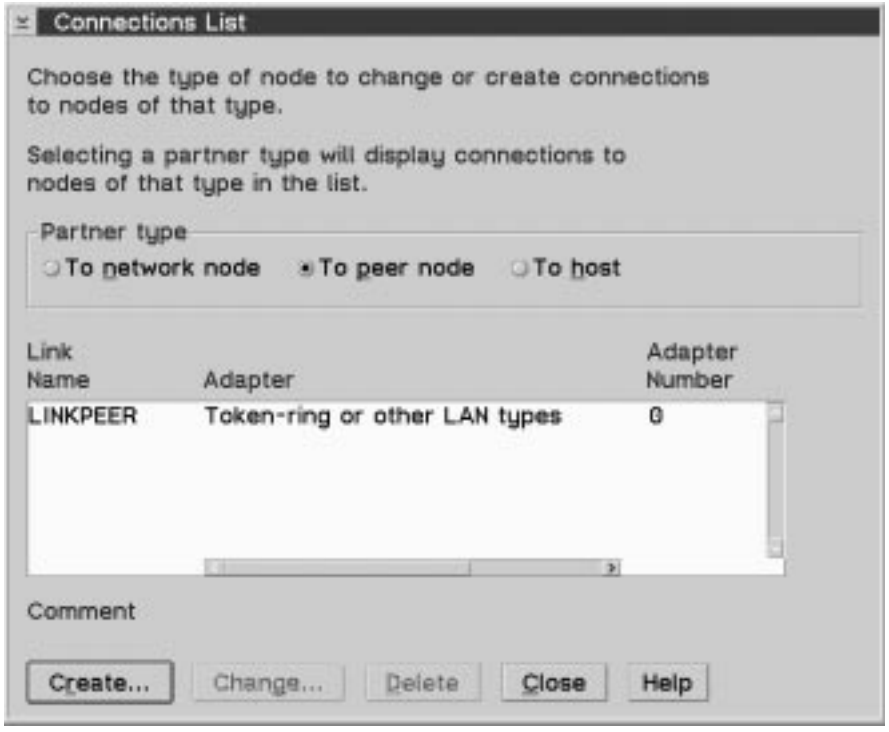
d. Enter your Node ID (**13** , **14**) in the **Local node ID (hex)** field.

The first part should be filled in for you already when you display the profile. You only need to complete the second part.

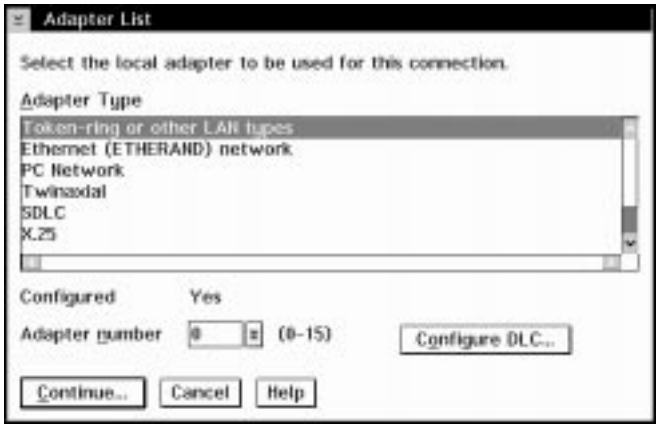
e. Click on **OK.**

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.

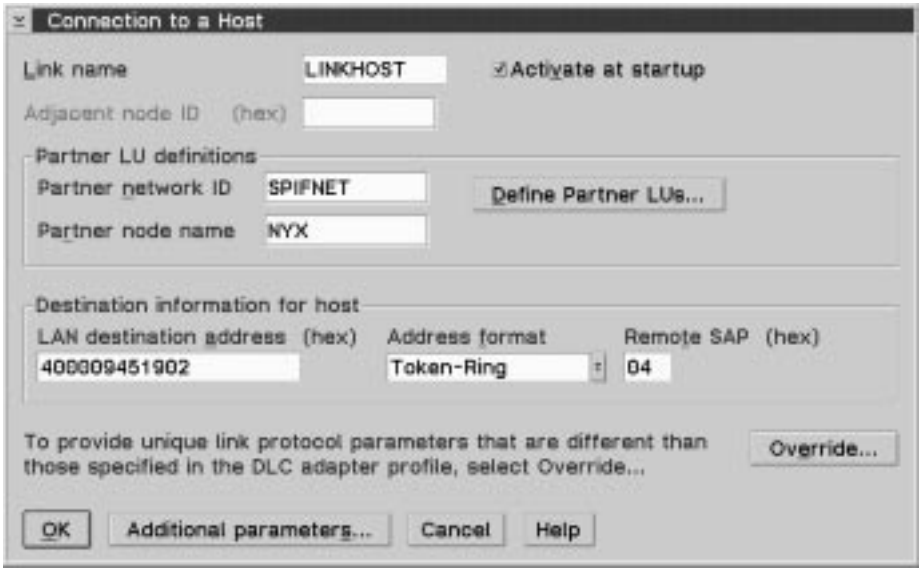


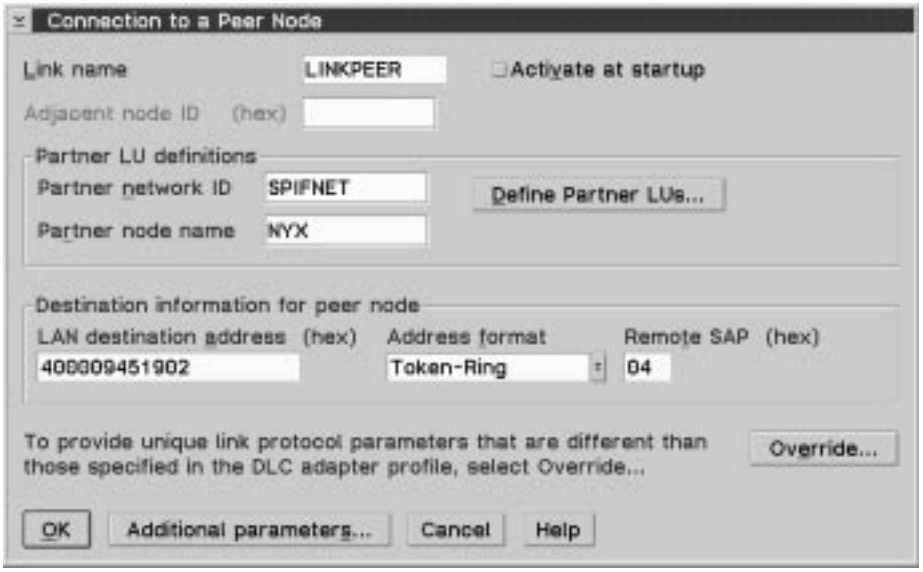
b. From the **Partner Type** window, select either the **To peer node** radio button (normally used for OS/400 connections), or the **To host** radio button (normally used for OS/390, MVS, VSE, and VM connections), and click on the **Create** push button. The Adapter List window opens.



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

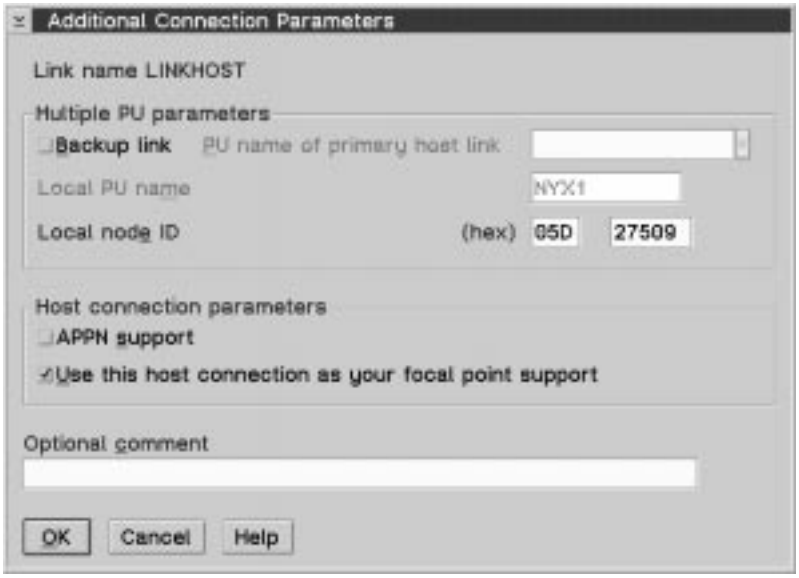
a. Enter the Network ID (**3**) of the remote system in the **Network ID** field.

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.

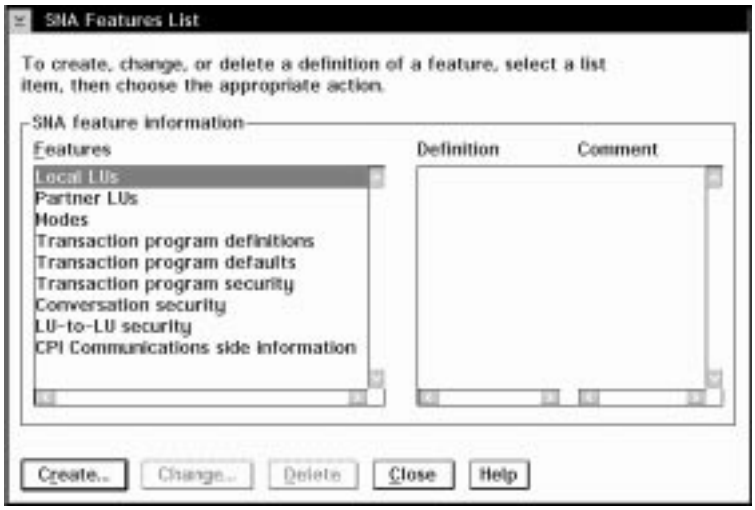
h. Click on **OK** to return to the Connections List window.

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 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.

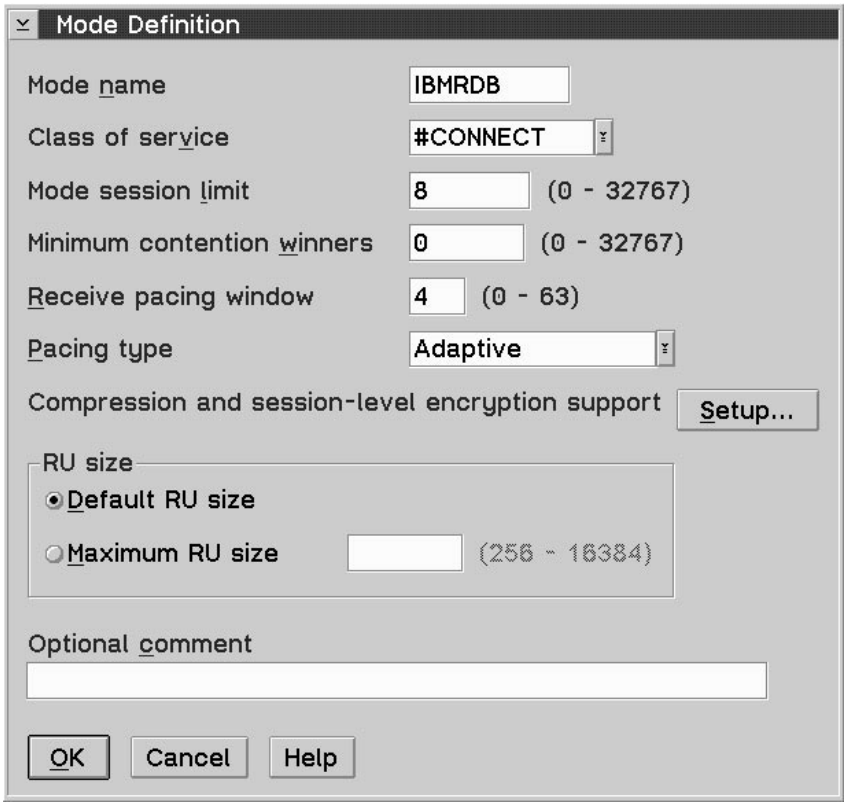
d. Select the **Independent LU** radio button in the **NAU address** box.

e. Click on **OK.**

f. To use this local LU when the 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.



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.

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 Comminucation 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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 and the sections that follow.

Configuring IBM Personal Communications for Windows NT and Windows 9x

This section describes how to configure IBM Personal Communications for Windows NT, 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:
   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.

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.

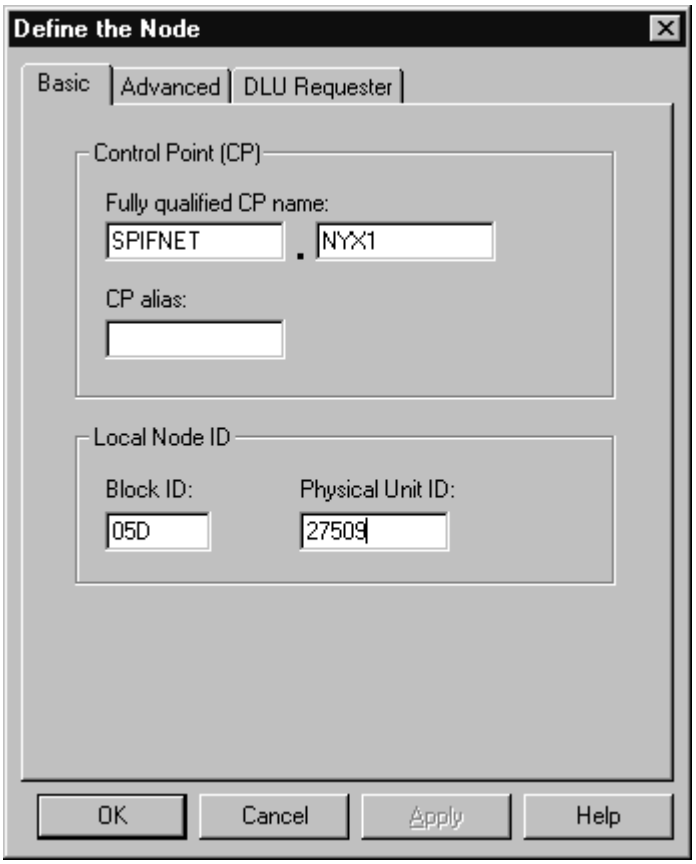
Step 2. Select **File->New** from the menu bar. The Define the Node window opens. Subsequent steps will begin from this window.

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.



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.

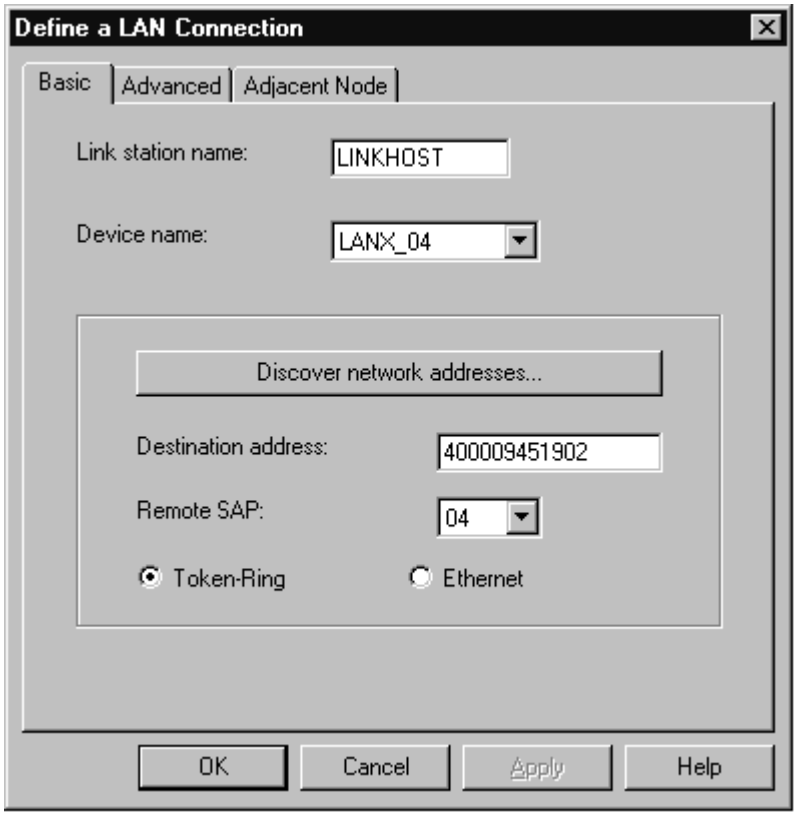
d. Click on **OK** to accept the default values.

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.

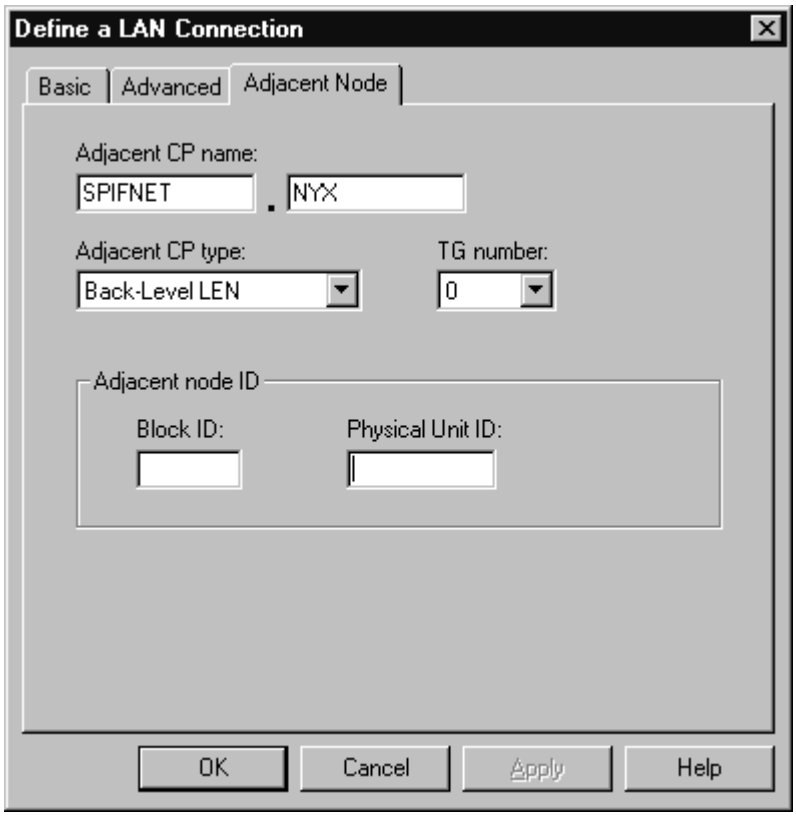


d. On the Basic tab panel:

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:

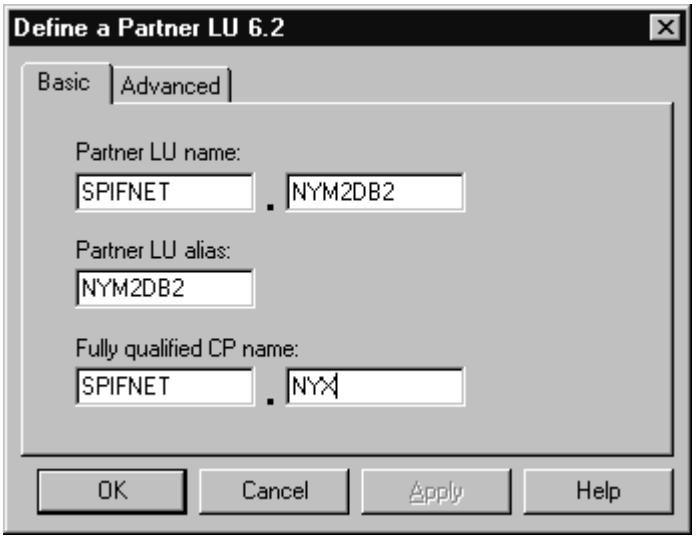


1) In the **Adjacent CP name** fields, type in the network ID **3** and the Control Point name **4** (`SPIFNET.NYX`).
2) In the **Adjacent CP type** field, select **Back-level LEN**.
3) Ensure that **TG number** is set to `0` (the default).
4) Click on **OK**.

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.



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`).

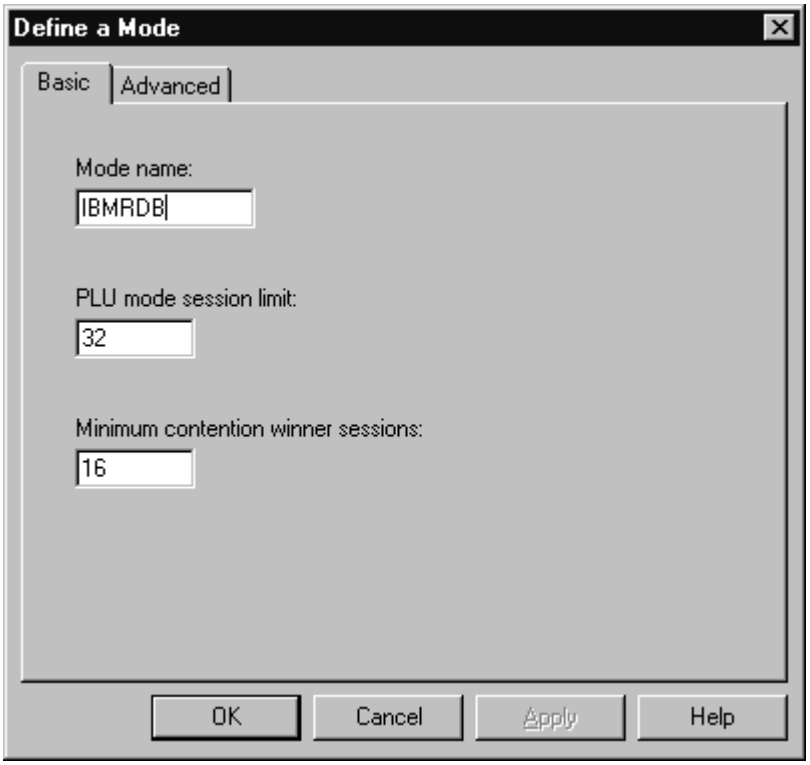
d. In the **Fully-qualified CP name** fields, type in the network ID **3** and the adjacent control point SSCP name **4** (`SPIFNET.NYX`).

   Accept the defaults in the Advanced tab panel.

e. Click on **OK**.

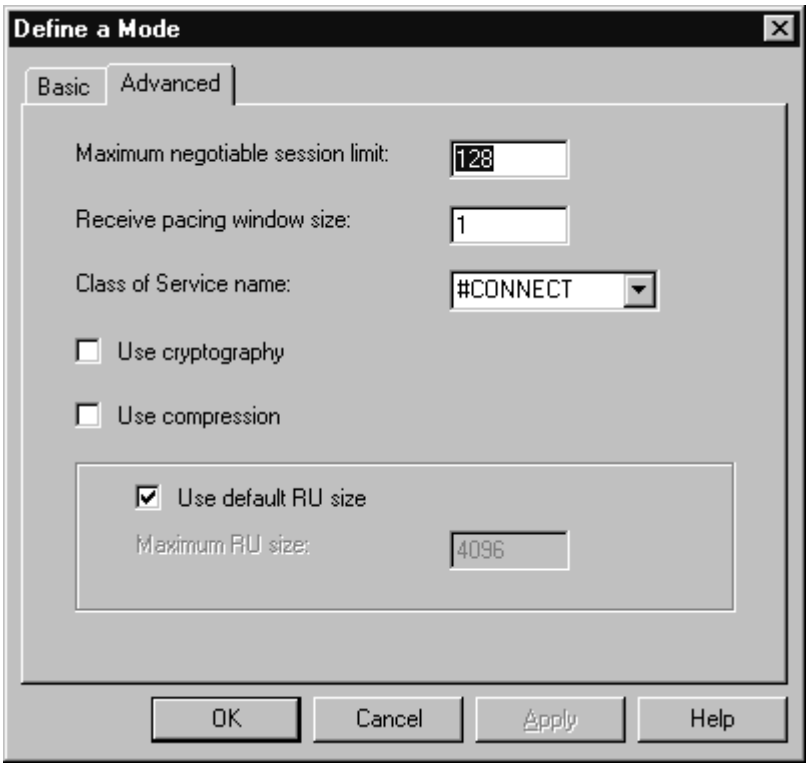
Step 5. Configure Modes

a. In the **Configuration options** box, select **Configure modes**, then click on the **New** button. The Define a mode window opens.



b. Enter your Mode name (**15**) in the **Mode name** field of the Basic tab.

c. Select the **Advanced** tab.



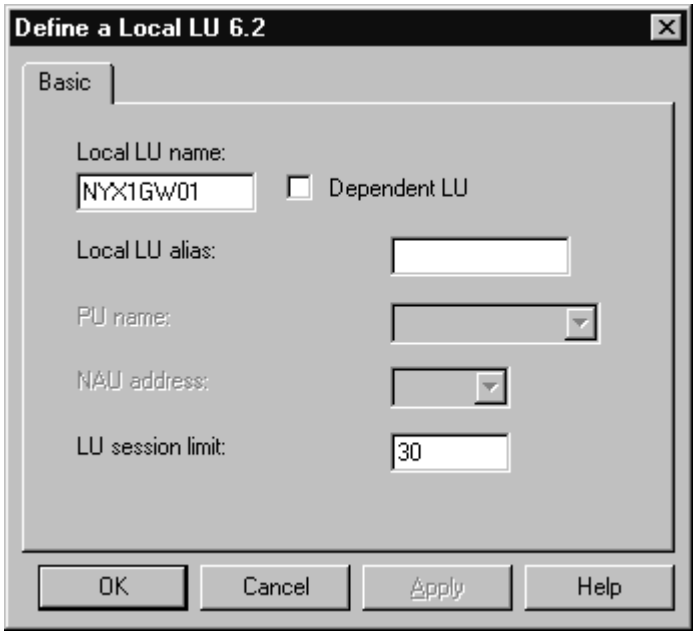
d. Select **#CONNECT** from the **Class of Service Name** field.

e. Click 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.



b. Enter your Local LU name (**11**) in the **Local LU name** field.

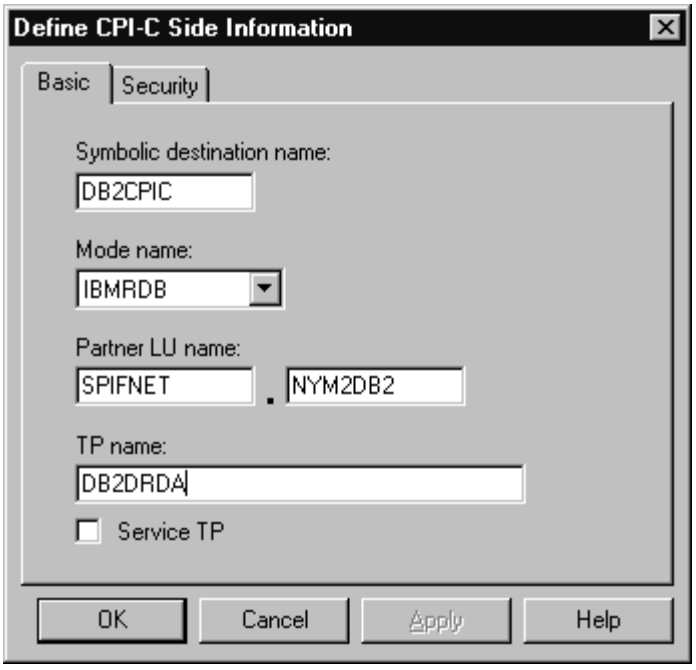
c. Type in a value for the **LU session limit**field. The default, 0, specifies the maximum allowed value. Accept the defaults for the other fields.

d. Click 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.

Accept the defaults for the other fields.

f. 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 **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 typing `set appcllu=`*`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 `appcllu` in the **Variable** field.

e. Type your local LU name (**4**) in the **Value** field.

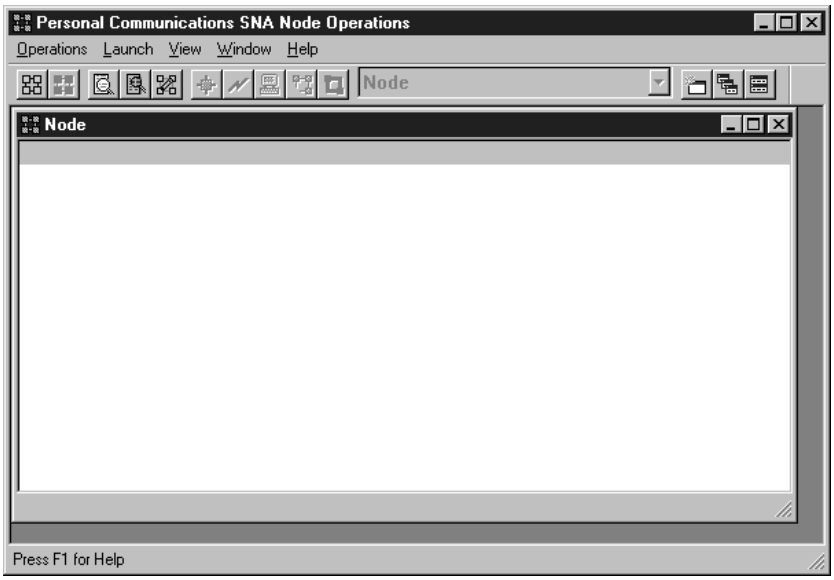
f. Click on the **Set** push button to accept the changes.

g. Click 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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 and the sections that follow.

Configuring IBM eNetwork Communications Server for Windows NT

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 was installed (this is an installation option for Communications Server)

__ 3. The APAR fixes JR11529 and J$11170 were 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 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, then select **Settings->Control Panel**.

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

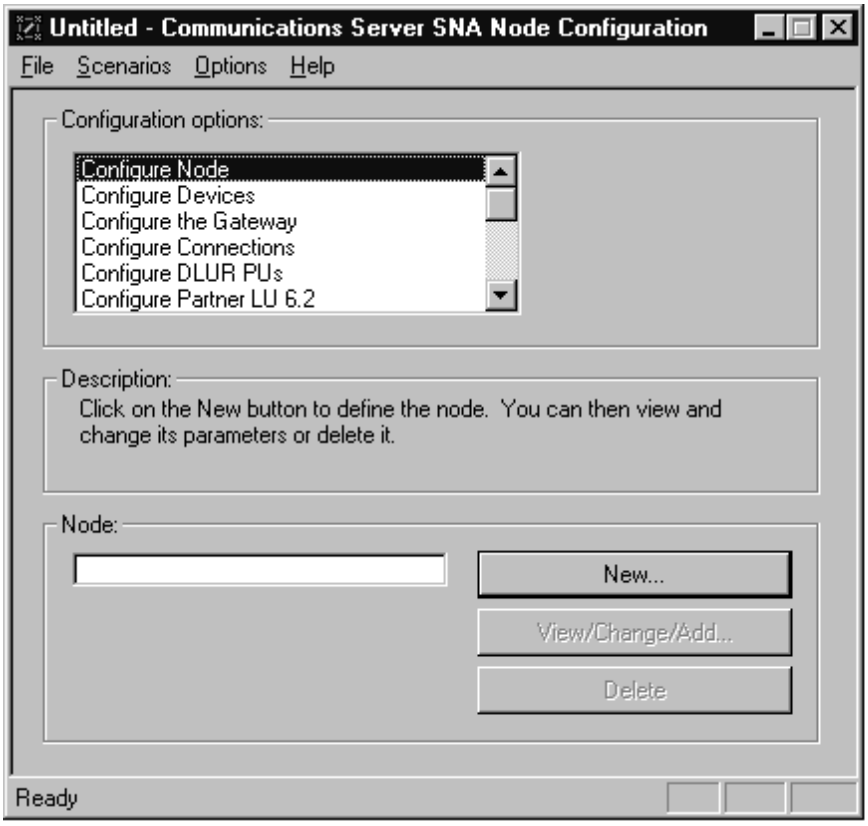
   c. On the Network window, click on the **Protocols** tab. **IBM LLC2 Protocol** must be one of the protocols listed. If it is not, you need to install this protocol from your IBM Communications Server for Windows NT software. Refer to documentation supplied with CS/NT for instructions.

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

Step 1. Click on the **Start** button 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 appears.

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 appears 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 NT 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 appears.

b. Click on the **SNA Clients** tab.

c. Ensure that there is a check mark in the **Enable SNA API Client Services** check box.

d. Click on **OK** to accept the default values.

Step 4. Configure Connections

a. In the **Configuration options** field, 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 appears.

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 (eg. **Back-level LEN**).

3) Ensure that **TG number** is set to `0` (the default).

4) Click on **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 appears.

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).

   Leave the other fields blank.

e. 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 appears.

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**.

   Accept the defaults for the other fields.

d. 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 appears.

b. In the **Local LU name** field, type in the name (**11**) from the worksheet (NYX1GW01).

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 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 appears.

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. 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, 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.

Accept the defaults for the other fields.

f. Click on **OK**.

Step 9. Save the Configuration

a. Select **File->Save as** from the menu bar. The Save As window appears.

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 10. Update the Environment

IBM Personal Communications uses an environment variable called **appcllu** 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 appcllu=`*`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 the **Start** push button and select **Settings->Control Panel**. Double-click on the **System** icon. When the **System Properties** window appears, select the **Environment** tab.

Step b. Type `appcllu` in the **Variable** field, and type 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 the **Start** push button, then 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 NT Service

To automatically start Communications Server when the machine is booted, 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 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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 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 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 accout. 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 on the **Groups** push button

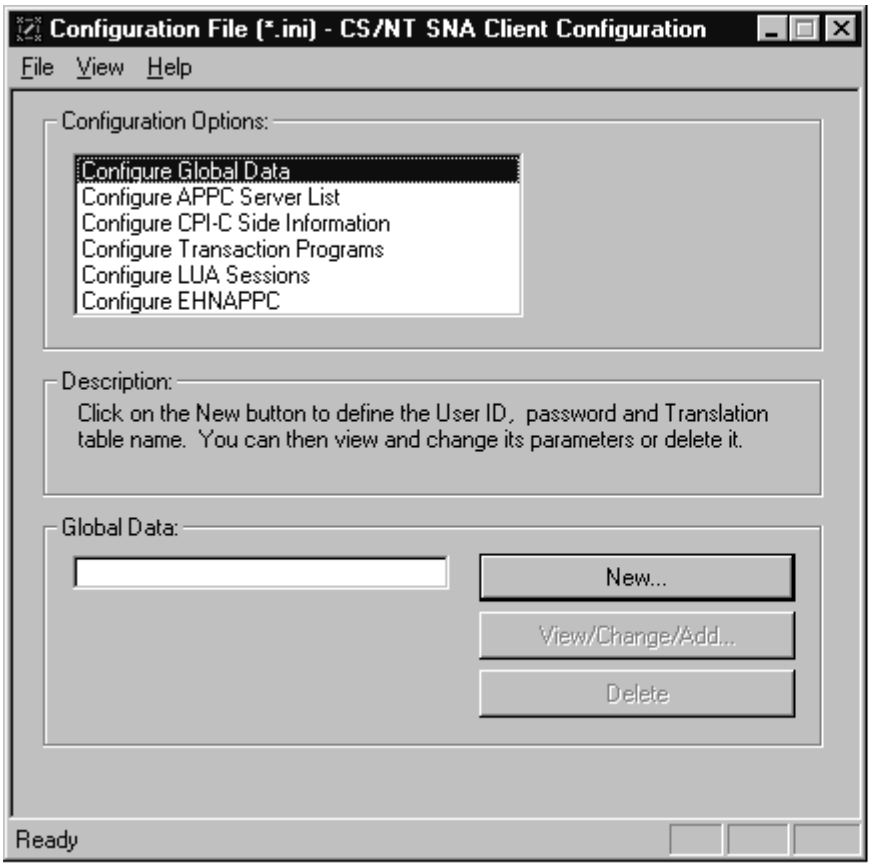
2) Select a group from the **Not member of** box and click on the **<- Add** push button. Repeat this step for each group that your user account must belong to.

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** push 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. 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 on **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 appears.

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 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, 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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 and the sections that follow.

Configuring Microsoft SNA Server for Windows NT

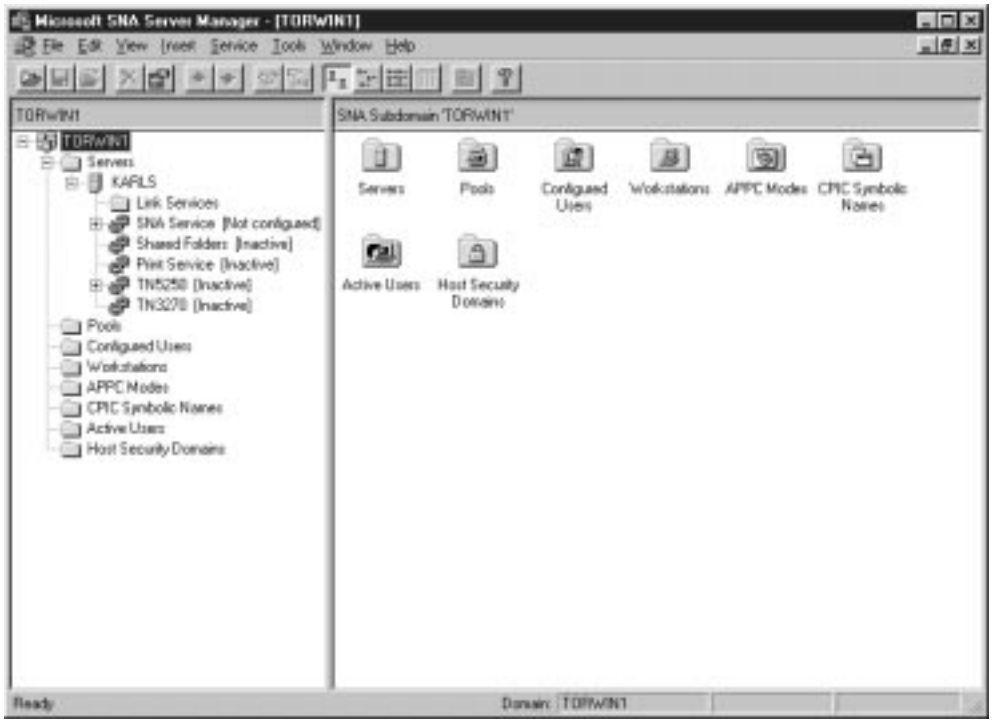
This section describes how to configure Microsoft SNA Server Version 4.0 for Windows NT on your DB2 Connect workstation to connect to host or AS/400 database servers using APPC. Although Microsoft SNA Server will run on Windows NT 4.0 Workstation, Windows NT 4.0 Server is recommended.

For instructions on how to configure Microsoft SNA Client for Windows NT, go to "Configuring Microsoft SNA Client" on page 292.

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. 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. Click on the [+] sign beside the **Servers** folder.

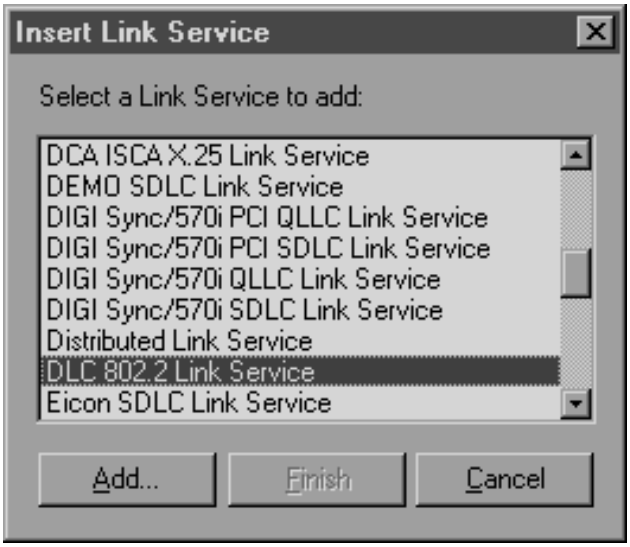
b. Right-click on **SNA Service** folder and select the **Properties** option. The Properties window opens.

c. Enter the correct **NETID** (**9**) and **Control Point Name** (**10**) in the corresponding fields.

d. Click on **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.



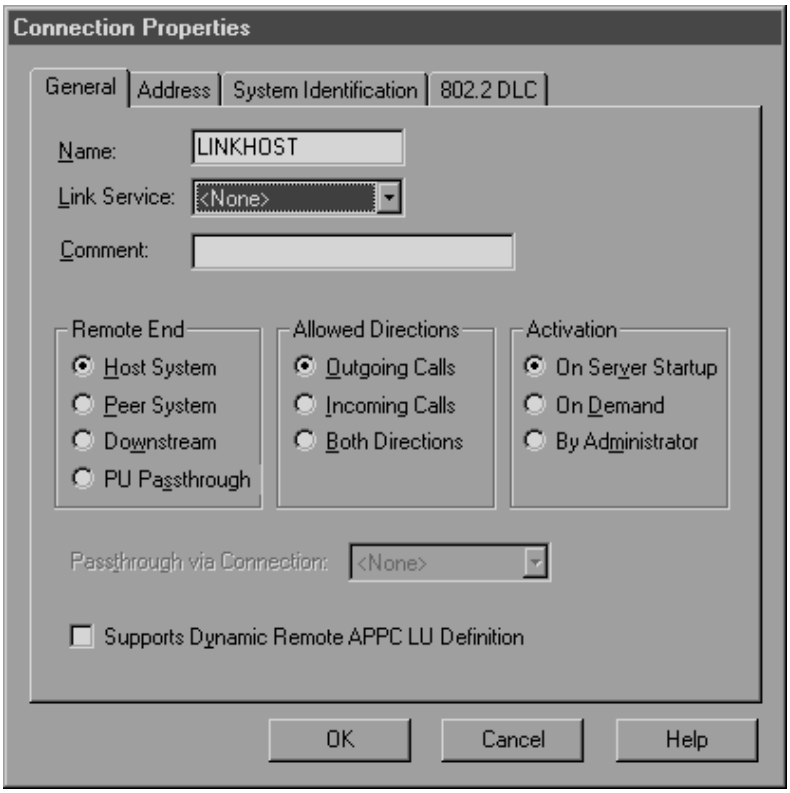
b. Select DLC 802.2 Link Service.

c. Click on the **Add** push button.

d. Click on the **Finish** push button.

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.



b. Enter a connection name (**7**) in the **Name** field.

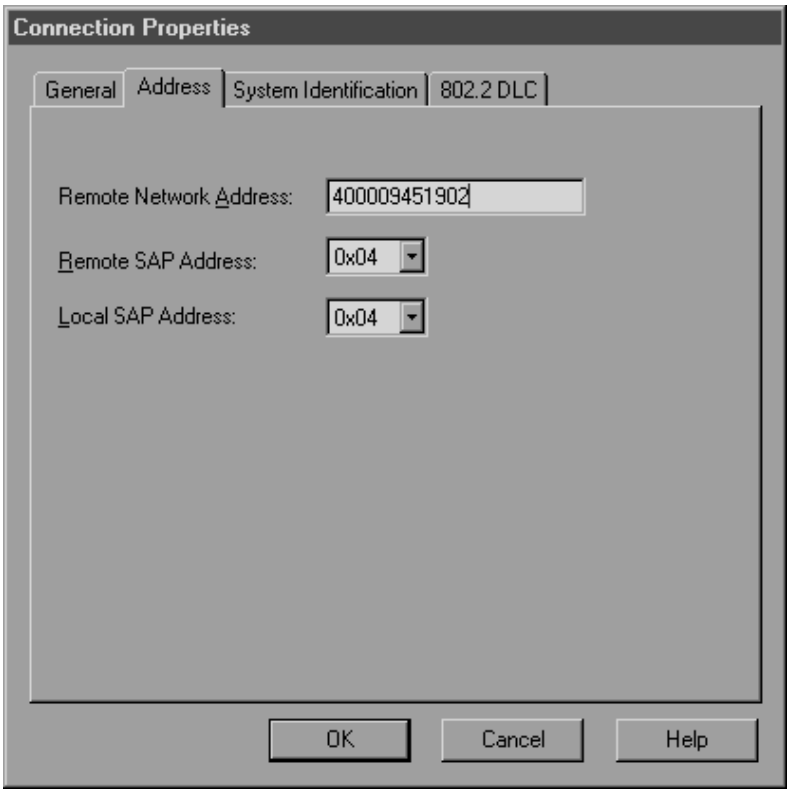
c. Click on the **Link Service** drop-down box and select the **SnaDlc1** option.

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.

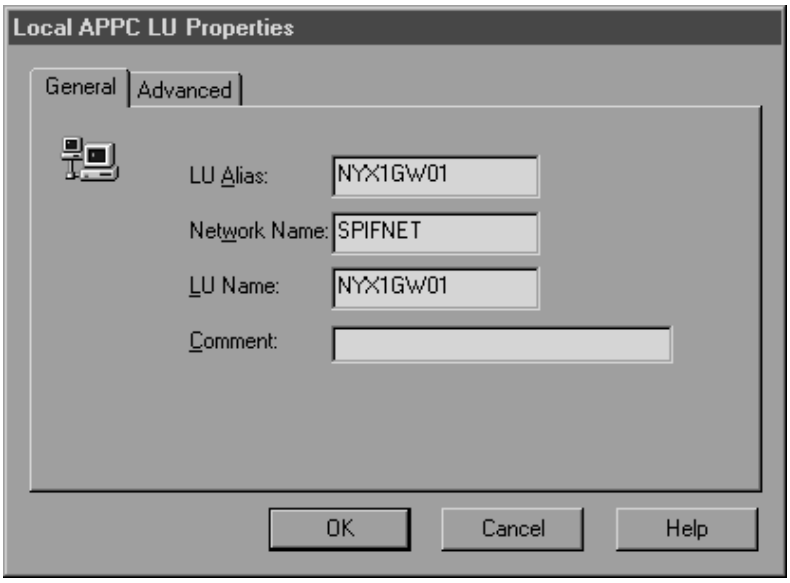
2) For the **Remote Node Name**, add the **NETID** (1) and the **Control Point Name** (4). Accept the other defaults

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 **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.

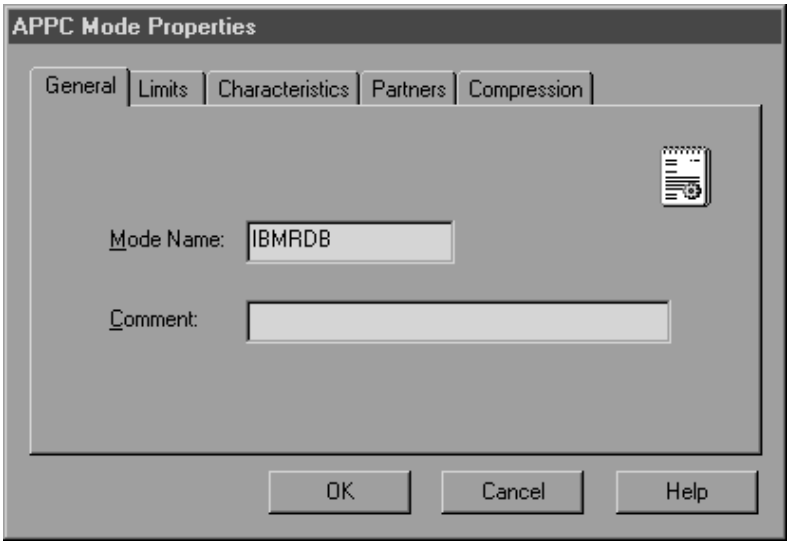
d. Enter the Network ID (**1**) in the **Network Name** field.

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 on **OK**.

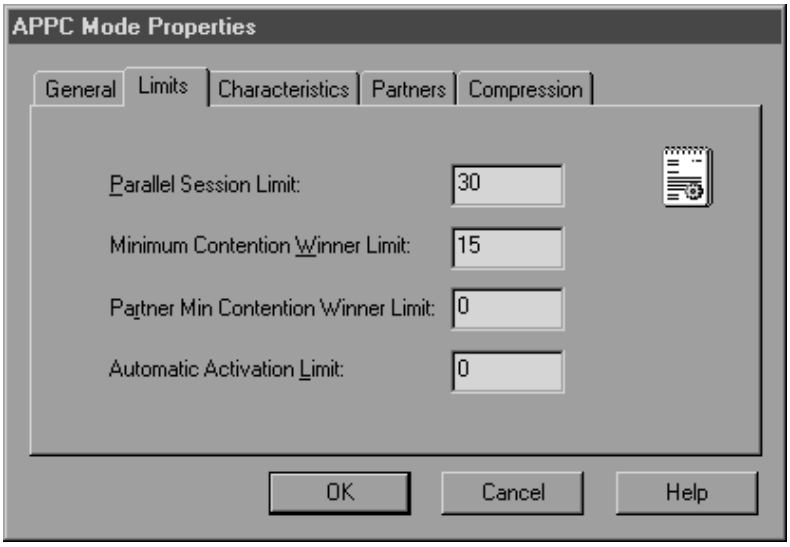
Step 7. Define a mode

a. Right-click on **APPC Modes** folder and select the **Insert->APPC->Mode Definition** option. The APPC Mode Properties window opens.



b. Enter the Mode Name **6** in the **Mode Name** field.

c. Select the **Limits** tab.

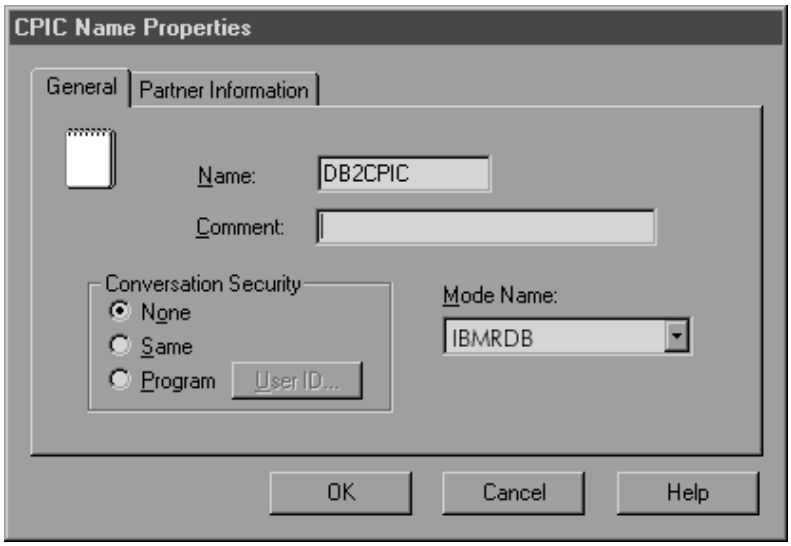


d. Enter appropriate numbers for the **Parallel Session Limit** and **Minimum Contention Winner Limit** fields. Your Host-Side or LAN administrator should be able to supply you with the numbers if you do not know the limits you should place here.

e. Accept the other defaults, and click on **OK**.

Step 8. Define the CPIC name properties

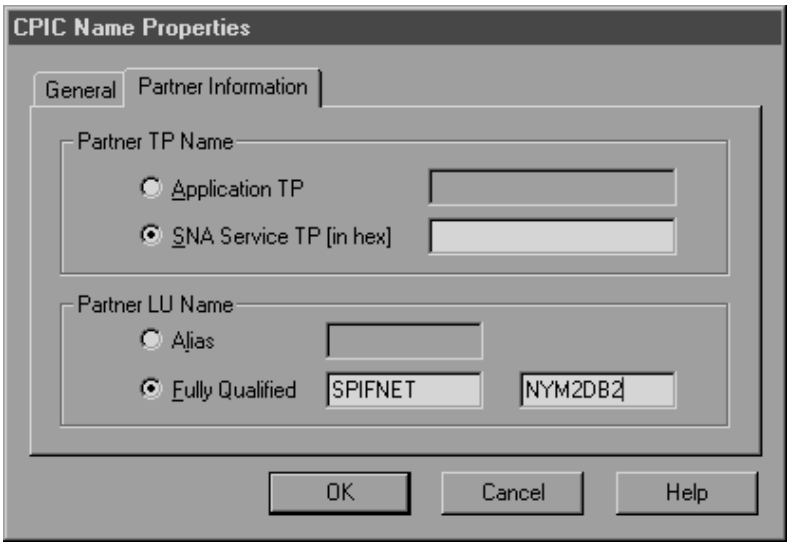
a. Right-click on **CPIC Symbolic Name** folder icon and select the **Insert->APPC->CPIC Symbolic Name** option. The CPIC Name Properties window opens.



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.



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 on **OK**.

i. 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.

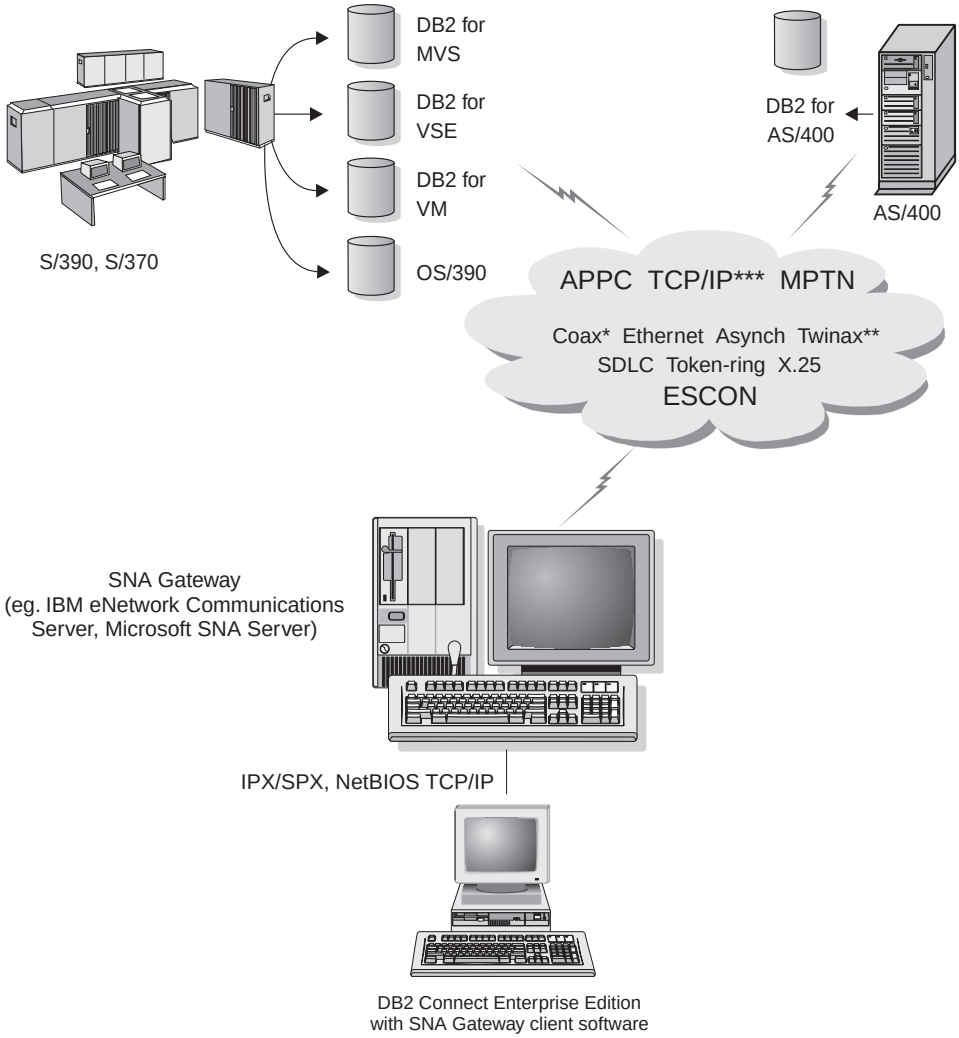
For OS/2 and Windows platforms, the easiest way to do this is to use the Client Configuration Assistant (CCA). For more information on using the CCA, see "Chapter 6. Configuring Client-to-Server Communications Using the Client Configuration Assistant" on page 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 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 NT" on page 283.

Figure 2 on page 293 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:

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 the host and AS/400 databases that you need to access, as described in "Configuring Microsoft SNA Server for Windows NT" on page 283.

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

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 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

Step 1. Install DB2 Connect.

Step 2. Click on the **Start** button and select **Programs->DB2 for Windows NT->Client Configuration Assistant**.

Step 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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 and the sections that follow.

Configuring IBM eNetwork Communication Server for AIX

This section describes how to configure IBM eNetwork Communication Server V5.0.2.5 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.

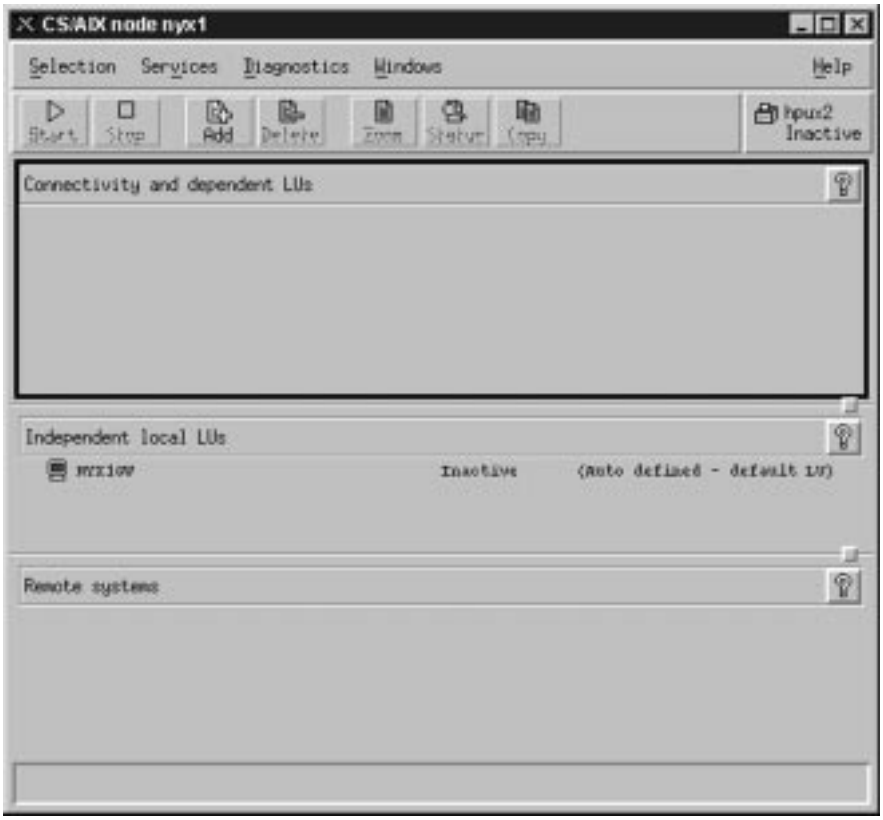
Before you begin, ensure that your workstation has IBM eNetwork Communication Server V5.0.2.5 for AIX (CS/AIX) installed. If you need further information in order to configure your SNA environment, refer to the online help provided with CS/AIX.

The following assumptions are made:

- The basic installation of the IBM eNetwork Communication Server V5 for AIX package has already been completed.
- DB2 Connect has been installed.
- The user is logged on as root.

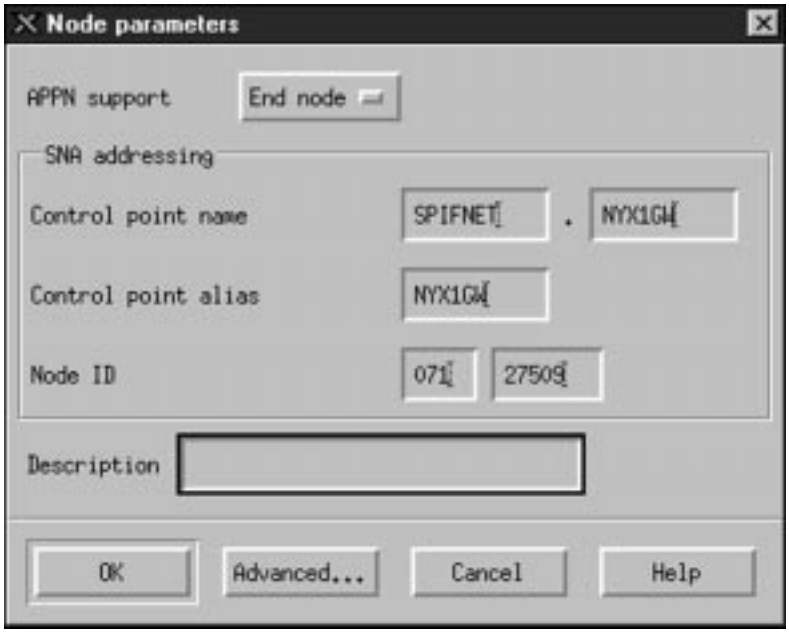
To configure CS/AIX for use by DB2 Connect, log on to the system as a user with `root` authority and use either the **/usr/bin/snaadmin** program or the **/usr/bin/X11/xsnaadmin** program. Information about these programs can be found in the system documentation. The following steps describe how to use 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 Local PU Name (**10**) in the **Control point alias** field.

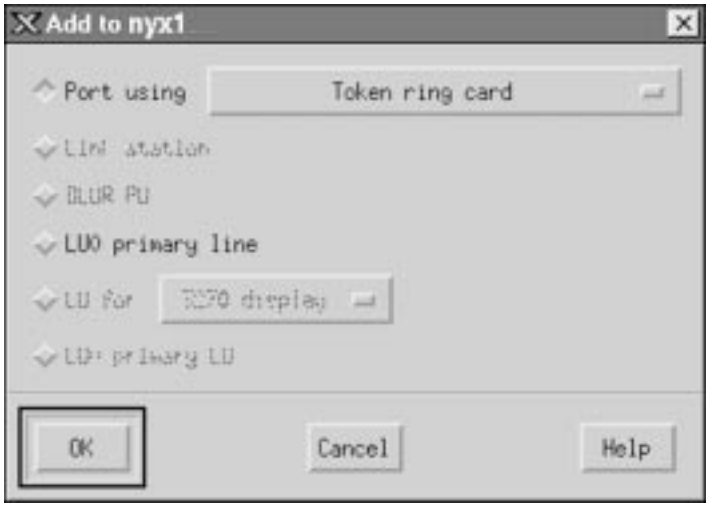
e. Enter your Node ID (**13** and **14**) in the **Node ID** fields.

f. Click on **OK**.

Step 3. Define a port

a. Select the **Connectivity and Dependent LUs** windows.

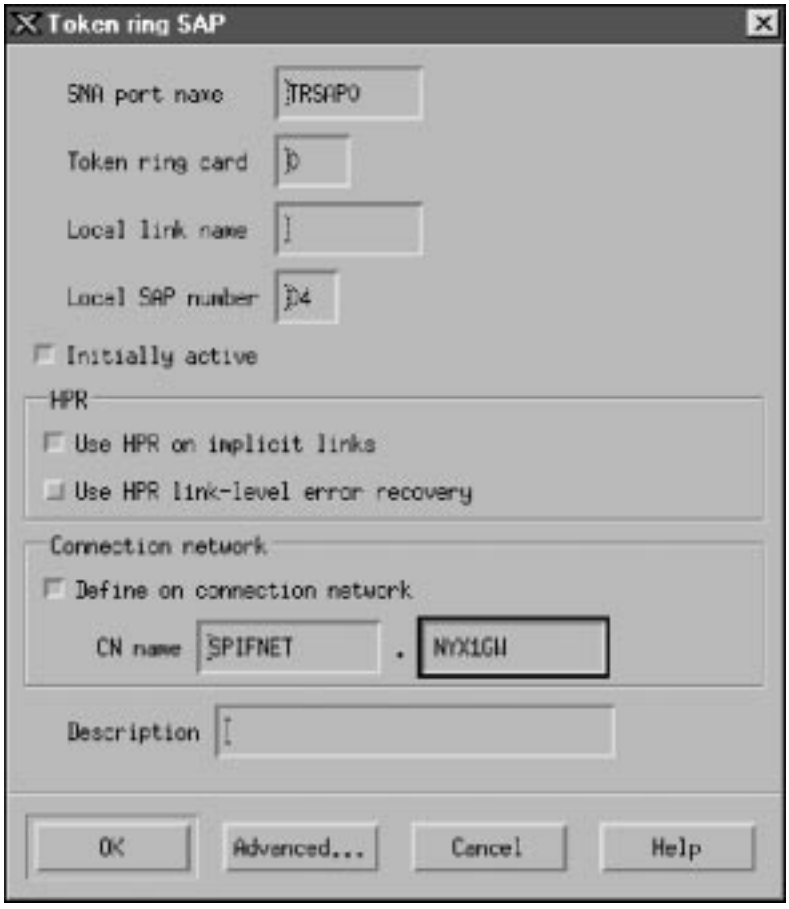
b. Click on the **Add** push button. The Add to Node window opens.



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 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. 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 on **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 on **OK**. The Token ring link station window opens.

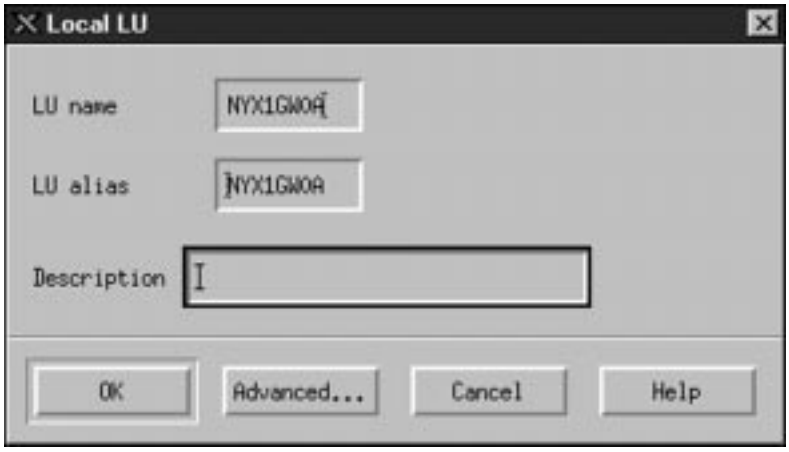
e. Enter a name for the link in the **Name** field.
f. Click on the **Activation** drop down box and select the **On demand** option.
g. Select the **Independent only** option in the **LU traffic** box.
h. In the **Independent LU traffic** box:
   1) Enter the Network ID (**3**) and the Partner LU Name (**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 for the host or AS/400 system in the **Mac address** field.

j. Click on **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 on the **Add** push button. The Local LU window opens.



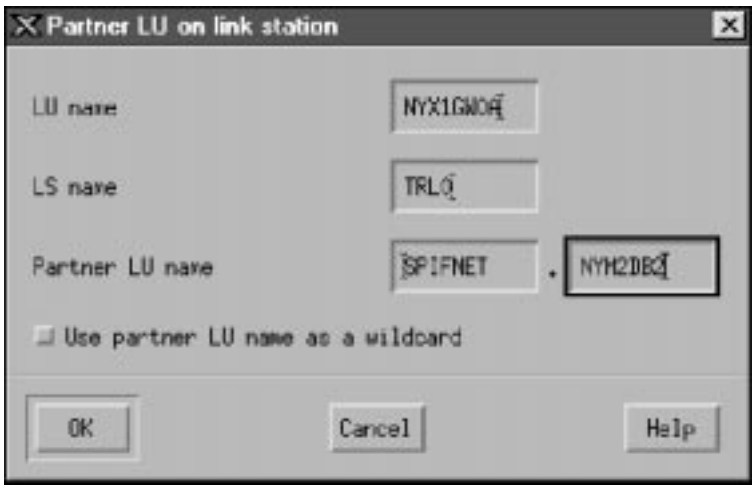
c. Enter your independent local LU Name (**11**) in the **LU name** field.

d. Enter the same name 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 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** fields.

e. Click on **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 Sytems** window.

b. Click on the **Add** push button. The Add to node window opens.

c. Select the **Define partner LU alias** radio button.

d. Click on **OK**. The Partner LU window opens.

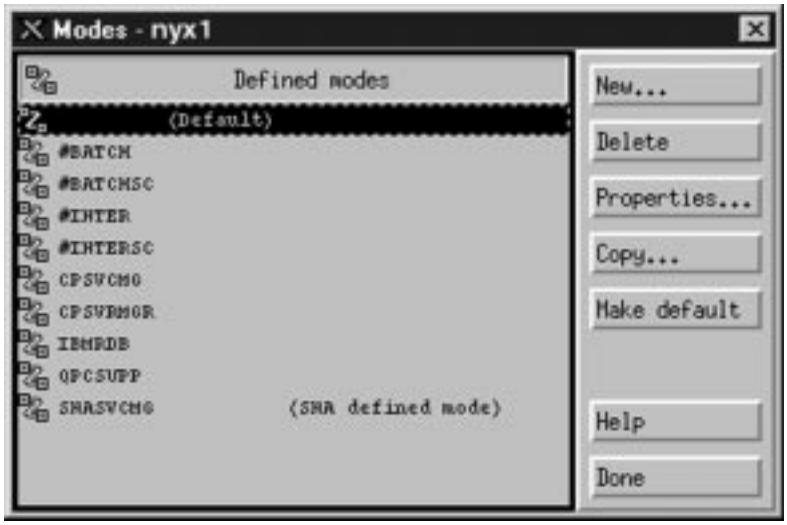
e. Enter an alias for the partner LU in the **Alias** field.

f. Enter the same value in the **Uninterpreted name** field.

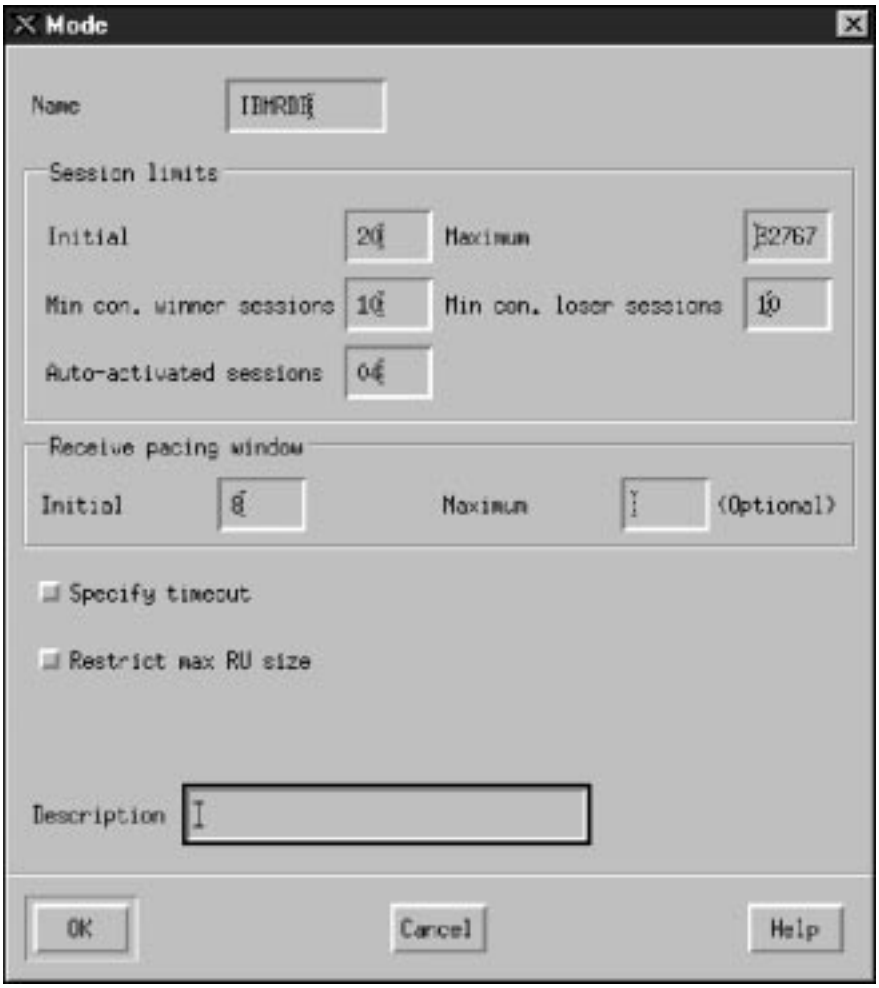
g. Click on **OK**.

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

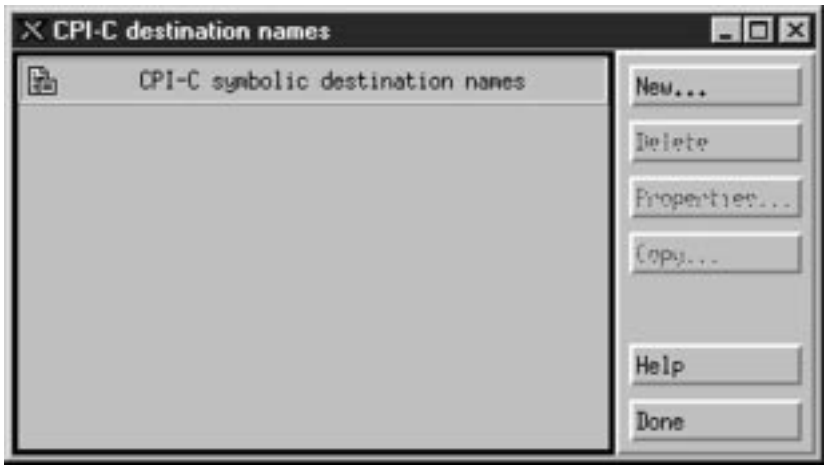
These values are suggested because they are known to work. You will need to tailor these values so that they are optimized for your particular application environment.

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

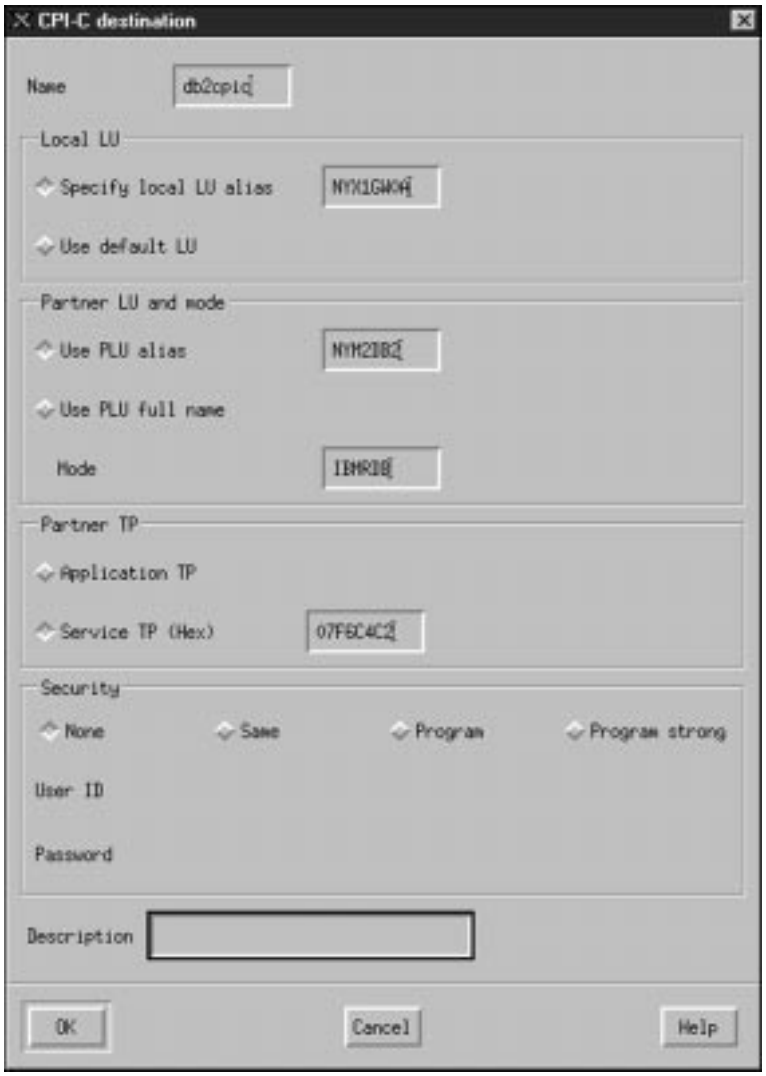
f. Click on **Done**.

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 host or AS/400 server database in the **Name** field. This example uses `DB2CPIC`.

d. 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 VSE, 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) 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 on **OK**. The new destination name appears in the Destination names window.

h. Click on **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 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 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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 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 V5.0.2.5 for AIX. See "Configuring IBM eNetwork Communication Server for AIX" on page 295 for more information.

To determine if Bull SNA is installed on your AIX 4.1.4 (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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 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.

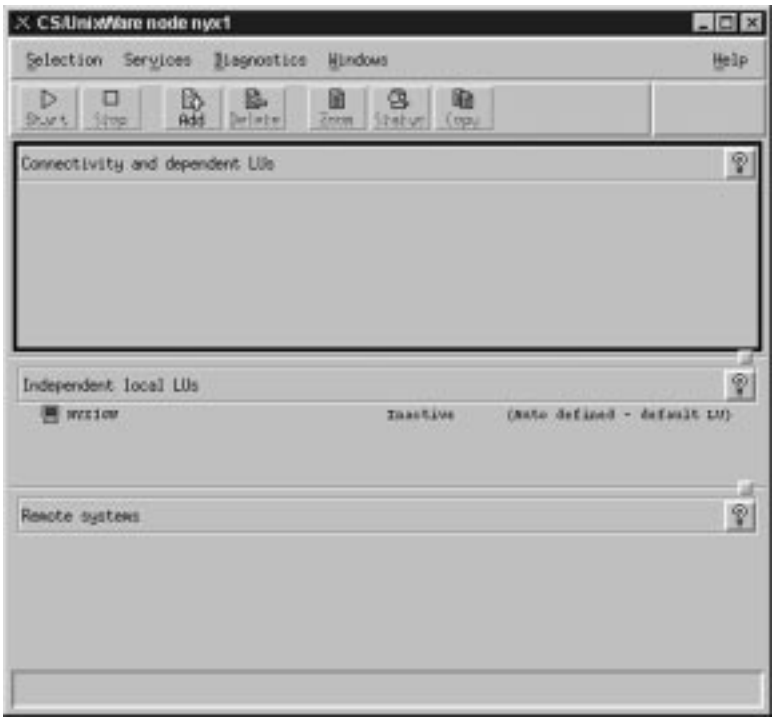
Before you begin, ensure that your workstation has HP-UX SNAplus2 installed. If you need more information in order to configure your SNA environment, refer to the online help provided with SNAplus2.

The following assumptions are made:

- The basic installation of the SNAplus2 for HP-UX package has already been completed.
- DB2 Connect has been installed.
- The user is logged on as root.

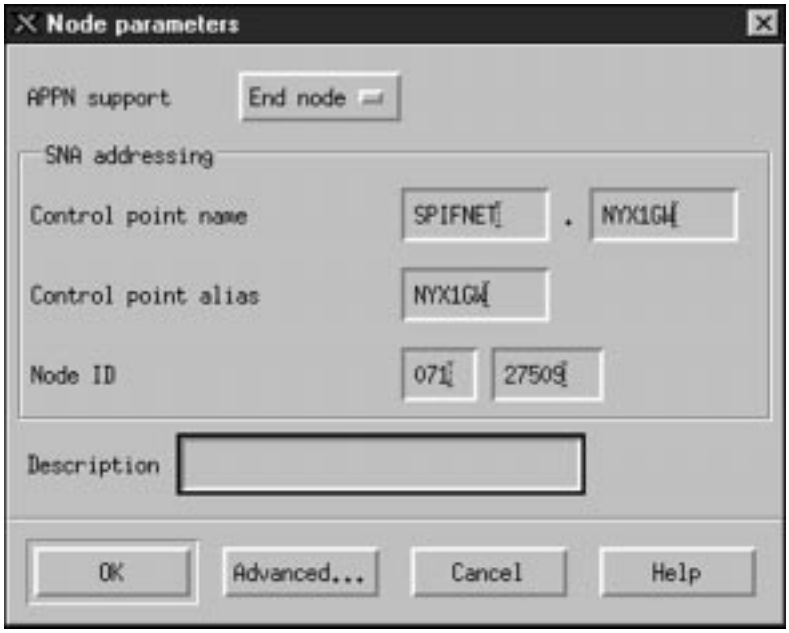
To configure SNAplus2 for DB2 Connect, log on to the system as a user with `root` authority and use either the **/opt/sna/bin/snapadmin** program or the **/opt/sna/bin/X11/xsnapadmin** program. Information about these programs can be found in the system documentation. The following steps describe how 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.



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 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.

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 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 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 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 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** 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.

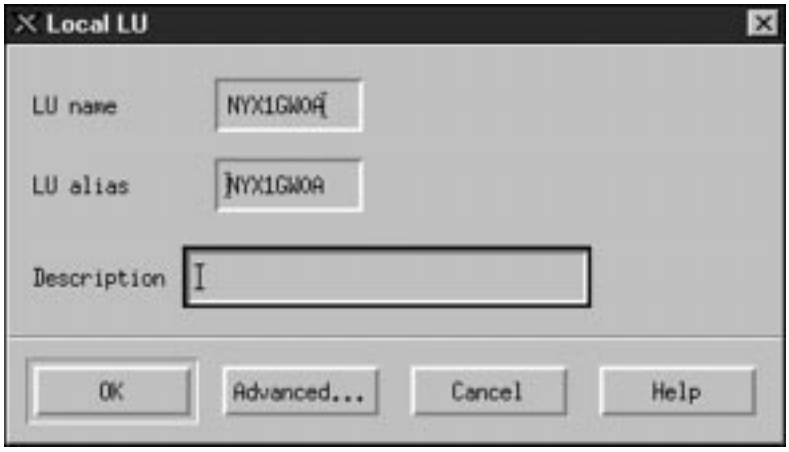
i. In the **Contact information** box, enter the SNA Destination Address (**8**) assigned for the host or AS/400 system in the **Mac address** field.

j. Click on **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 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 same name in the **LU alias** field (**12**).

e. Click on **OK**. The new LU appears in the **Independent local LUs** window.

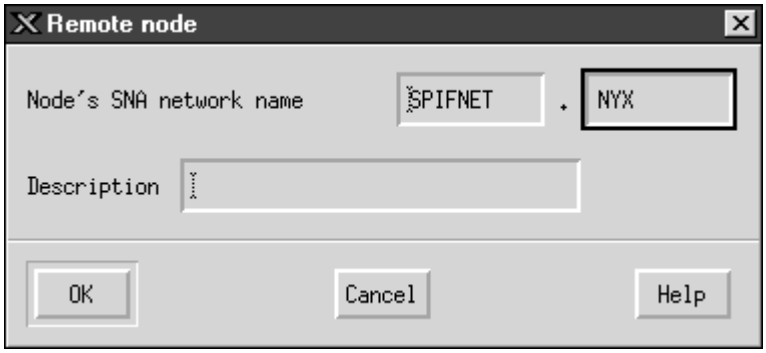
Step 6. Define a remote node

a. Select the **Remote Systems** window.

b. Click on the **Add** push button. The Add to Node window opens.

c. Select **Define remote node**.

d. Click on **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 on **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.

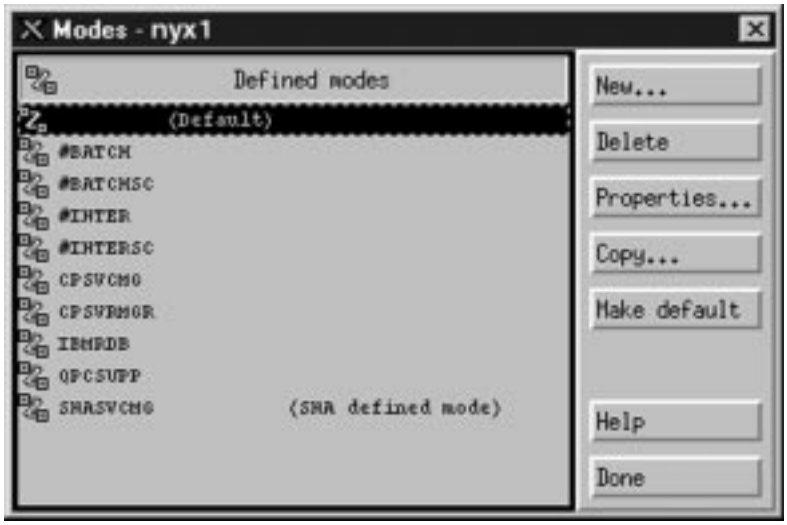
b. Enter the same Partner LU name (**2**) in the **Alias** and **Uninterpreted name** fields.

c. Select **Supports parallel sessions**.

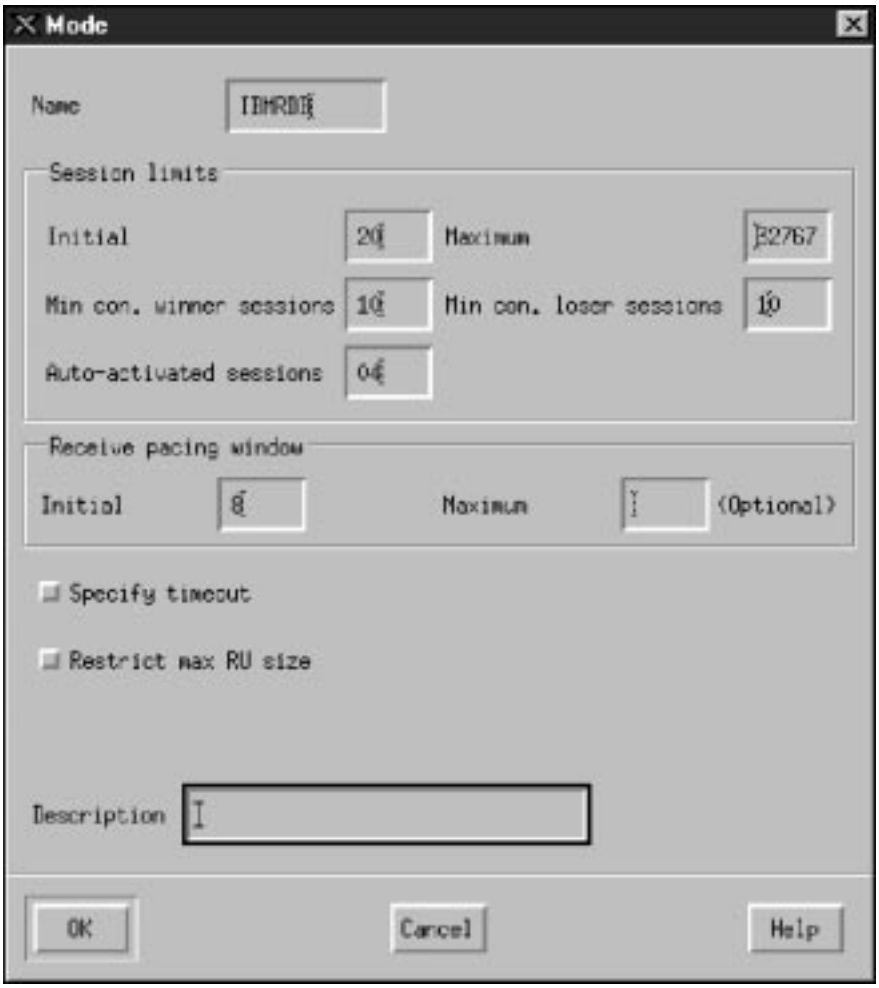
d. Click on **OK**.

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

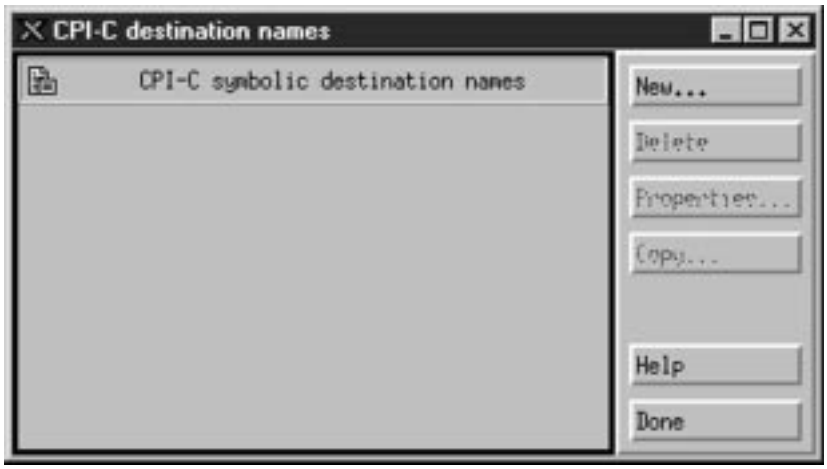
These values are suggested because they are known to work. You will need to tailor these values so that they are optimized for your particular application environment.

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

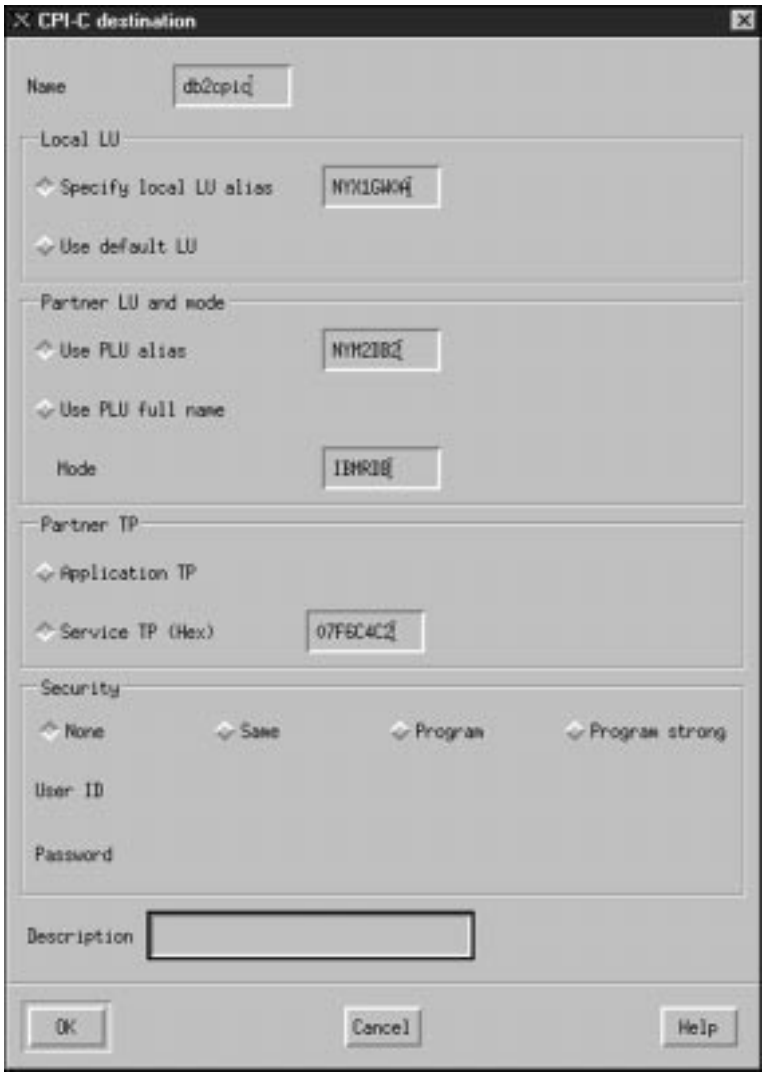
f. Click on **Done**.

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 host or AS/400 server database in the **Name** field. This example uses `DB2CPIC`.

d. 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 VSE, 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) 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 on **OK**. The new destination name appears in the Destination names window.

h. Click on **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 use either the **/opt/sna/bin/snaadmin** command, the **/opt/sna/bin/X11/xsnaadmin** command.

c. Start the subsystem node. Select the appropriate the 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 on the **Start** push button.

e. Start the session. Select the LU you defined previously in the **Independent Local LUs** pane, then click on the **Start** button. A session activation window opens. Select or enter the Partner LU and Mode desired.

f. Click on **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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" on page 327 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 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 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.
- DB2 Connect has been installed.
- The user is logged on as root.

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. "Create the CPIC Side File"

Step 2. "Create the SNA Server Configuration File" on page 325

Step 3. "Define the Environment Variables Required by SunLink SNA" on page 326

Step 4. "Start the SunLink SNA Subsystem" on page 326

Note: 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.

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. However, for remote DB2 clients to be able to access it on the DB2 Connect for Solaris system, the file should be in one of the following directories:

- INSTHOME/sqllib/adm or
- INSTHOME/sqllib/bin

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 Connect for Solaris system.

The example below shows the CPIC side file sections required for configuring the SunLink SNA to connect to host or AS/400 database servers.

```
# CPIC Side File information
#
PTNR_LU_NAME=NYM2DB2
MODE_NAME=IBMRDB
TP_NAME=X'07F6C4C2'
SECURITY=PROGRAM
```

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 host or AS/400 database servers. Other sections that are not shown are required for accepting inbound APPC 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=NYX1                       // Local name (8 char max)
        NQ_CP_NAME=SPIFNET.NYX1         // Network Qualified Name
        window opens

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

DLC     NAME=HOSTLINK                   // 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'400009451902       // sysB_mac_addr
        TERMID=x'07127509'              // XID negotiation
        window opens
```

```
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
        window opens

PTNR_LU NAME=NYM2DB2                    // Partner LU name(8 char max)
        LOC_LU_NAME=NYX1GW0A            // Associated Local LU
        NQ_LU_NAME=SPIFNET.NYM2DB2      // Network Qualified Name
        window opens

MODE    NAME=IBMRDB                     // Mode Name (8 char max)
        DLC_NAME=HOSTLINK               // Associated DLC
        PTNR_LU_NAME=NYM2DB2            // Associated Partner LU
        LCL_MAX_SESS_LMT=30             // Max Session Limit
        MIN_CW_SESS=15                  // Min Conwinners
        MIN_CL_SESS=15                  // Min Conlosers
        window opens
```

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.

For a DB2 Connect for Solaris server, export these on the DB2 Connect 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 full details.

Step 3. Ensure you have created the CPIC side file.

Step 4. Ensure you have created the SNA server configuration file.

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 31. For manual configuration instructions, and for UNIX platforms, see "Step 3. Catalog the APPC or APPN Node" and the sections that follow.

Step 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, Windows 9x, and Windows NT, 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/sqllib/db2profile    (for Bourne or Korn shell)
source INSTHOME/sqllib/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 must 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 26 on page 248:

```
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:

```
uncatalog node node_name
```

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

Step 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* is the local name of the host or AS/400 database.

- *target_dbname* is 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
```

Step 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 27. Worksheet: Parameter Values for Cataloging Databases

Parameter	Description	Sample Value	Your Value
Database name (*database_name*)	The local DCS database name (*local_dcsname*) of the *remote* database. You specified this when you catalogued the DCS database directory. For example, *ny*.	`ny`	
Database alias (*database_alias*)	An arbitrary local nickname for the remote database. If you do not provide one, the default is the same as the database name (*database_name*). This is the name that you use when connecting to the database from a client.	`localny`	

Table 27. Worksheet: Parameter Values for Cataloging Databases (continued)

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

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

Step 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.

To bind the utilities and applications to the host or AS/400 database server, enter the following commands:

```
db2 connect to dbalias user userid using password
db2 "bind path@ddcsmvs.lst blocking all sqlerror continue
     messages mvs.msg grant public"
db2 connect reset
```

For example:

```
db2 connect to NYC3 user myuserid using mypassword
db2 "bind /sqllib/myapps@ddcsmvs.lst blocking all sqlerror continue
     messages mvs.msg grant public"
db2 connect reset
```

The *userid* and *password* specified must have the authority to bind applications against the target database.

Further information about these commands can be found in the *DB2 Connect User's Guide*.

Step 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 *db_alias* value you defined in "Step 3. Catalog the APPC or APPN Node" on page 327 :

```
connect to db_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 tabname 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 15. Enabling Multisite Updates (Two-Phase Commit)

This chapter provides an overview of the multisite update function as it applies to scenarios that involve host and AS/400 database servers. It describes 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. A good example of a multisite update is 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 that implement debit operation on one account do not get committed unless updates required to process credit 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 update. This support is available for applications developed using regular SQL as well as applications that utilize Transaction Monitor products that implement X/Open XA interface specification. Examples of such Transaction Monitor products include IBM TxSeries (CICS and Encina), Message and Queuing Series, Component Broker Series, San Francisco Project as well as Microsoft Transaction Server (MTS), BEA Tuxedo, NCR TopEnd 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 SYCNCPOINT 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, e.g. `CICS SYNCPOINT`, `Encina Abort()`, `MTS SetAbort()`. When using native SQL multisite update, the normal `SQL COMMIT` and `ROLLBACK` must be used.

© Copyright IBM Corp. 1993, 1999

TP Monitor multisite update can coordinate a transaction that accesses both DB2 and non-DB2 resource managers such as Oracle, Informix, SQLServer, etc. 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. At the time of this writing, the following DB2 servers provided DUOW support that enabled them to participate in distributed transactions:

- DB2 Common Server V2
- DB2 Universal Database V5 and V6
- DB2 for MVS/ESA V3.1 and 4.1
- DB2 for OS/390 V5.1
- DB2 Universal Database for OS/390 V6.1
- DB2/400 V3.1 or later (using SNA only at this time)
- DB2 Server for VM and VSE V5.1 and 6.1

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, a DB2 for OS/390 database and a DB2/400 database all within a single transaction.

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. The need for SPM is dictated by the choice of protocol (SNA vs. TCP/IP) and use of a TP monitor. See the following table for a summary of scenarios that require use of the SPM. The table 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.

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

Host and AS/400 multisite update scenarios that require SPM.				
Yes	TCP/IP	Yes	• DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise-Extended Edition	• DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1
Yes	SNA	Yes	• DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise-Extended Edition* **Note:** *AIX, OS/2 and Windows NT platforms only.	• DB2 for MVS/ESA V3.1 and 4.1 • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 • DB2/400 V3.1 or later • DB2 Server for VM or VSE V5.1 or later

Host and AS/400 multisite update scenarios that require SPM.				
No	TCP/IP	No	• DB2 Connect Personal Edition • DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise-Extended Edition	• DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1
No	SNA	Yes	• DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise-Extended Edition* **Note:** *AIX, OS/2 and Windows NT platforms only	• DB2 for MVS/ESA V3.1 and 4.1 • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 • DB2/400 V3.1 or later • DB2 Server for VM and VSE V5.1 or later

Note:

For more information about DUOW refer to the *DB2 Connect User's Guide*.

For additional information about DB2 Connect two-phase commit support, as well as instructions for setting up for several popular TP monitors, please see the *Administration Guide* or access the DB2 Product and Service Technical Library on the World Wide Web:

1. Set your Web browser to the following URL:

   `http://www.software.ibm.com/data/db2/library/`
2. Select ″DB2 Universal Database″.
3. Search for ″Technotes″ using the search keywords ″DDCS″, ″SPM″, ″MTS″, ″CICS″, and ″ENCINA″.

Enabling Multisite Updates Using the Control Center

As of Version 6, you can use the Control Center to provide multisite update. The procedure is very simple, and is outlined below. For more information about the multisite update configuration process, including how to configure your system manually, refer to the on-line *Connectivity Supplement*.

Starting the Multisite Update Smartguide

From the Control Center, click on the [+] sign to expand the tree view. With the right mouse button, select the instance that you wish to configure. A popup menu opens. Select **Multisite Update->Configure** from this menu.

Smartguide Steps

The Smartguide provides a notebook-type interface. Each page of the notebook will prompt you for certain information about your configuration. The pages are shown below in the order in which you will encounter them.

Step 1. Specify a Transaction Processor (TP) monitor.

This field will show the defaults for the TP Monitor you have enabled. If you do not wish to use a TP monitor, select **Do Not Use a TP Monitor**.

Step 2. Specify the communications protocols you will use.

Step 3. Specify a Transaction Manager database.

This panel defaults to the first database you connect to (1ST_CONN). You can leave this default or select another catalogued database.

Step 4. Specify the types of database servers involved in the update, and also whether or not TCP/IP is to be used exclusively.

Step 5. Specify the Syncpoint Manager settings.

This page will only appear if the settings on the previous page indicate that you need to use DB2's Syncpoint Manager in a multisite update scenario.

Testing the Multisite Update Feature

Step 1. Select the instance with the right mouse button and choose the **Multisite Update->Test** menu option from the pop-up menu. The **Test Multi-Site Update** window opens.

Step 2. Select the databases you wish to test from the available databases in the left sub-window. You can use the arrow buttons in the middle to move selections to and from the **Selected Databases** sub-window.

Step 3. When you have finished your selection, press the **Test...** button at the bottom of the window. The **Multi-Site Update Test Result** window opens.

Step 4. The **Multi-Site Update Test Result** window shows which of the databases you selected succeeded or failed the update test. The window will show sql codes and error messages for those that failed.

Chapter 16. Enabling Multisite Updates with Host and AS/400 Database Servers and Clients

This section describes how to manually enable multisite updates with host and AS/400 database clients and servers.

It is strongly recommended that you use the Control Center Smartguide to configure multisite update. Please see "Chapter 15. Enabling Multisite Updates (Two-Phase Commit)" on page 333 or refer to the *Administration Guide* for information on Multisite Update (Distributed Unit of Work, Two Phase commit).

In addition, host and AS/400 database clients who wish to access DB2 Universal Database servers using multisite upate require the DB2 Universal Database server (or an upstream DB2 Universal Database Server which routes the connection to the destination server) to use the DB2 Syncpoint Manager.

For the most recent information about the DB2 Syncpoint Manager, access the DB2 Product and Service Technical Library on the World Wide Web:

1. Set your Web browser to the following URL:

   `http://www.software.ibm.com/data/db2/library/`
2. Select "DB2 Universal Database".
3. Search for "Technotes" using the search keywords "DDCS" and "SPM".

Planning to Use the SPM

This section describes important planning considerations that you should take into account before attempting to use multisite updates with host and AS/400 database servers.

Software Prerequisites

The following minimum software levels are required to support the SPM. Refer to the `Readme` file for any additional PTFs that may be required.

- AIX SNA connectivity:
  - IBM eNetwork Communications Server for AIX V5.0.2.4
- OS/2 SNA connectivity:

© Copyright IBM Corp. 1993, 1999

  - IBM Communications Server for OS/2 Warp Version 5 or higher must be installed on the system where DB2 Connect Enterprise Edition is installed or on the DB2 Universal Database server that has SPM support installed.

- Windows NT SNA Connectivity:
  - IBM Communications Server for Windows NT Version 5.01 or higher must be installed on the system where DB2 Connect Enterprise Edition is installed or on the DB2 Universal Database server that has SPM support installed.
- DB2 for MVS/ESA or DB2 for OS/390:

```
All DB2/MVS V3.1 systems
  PTFs UN73393 and UN76673  (fixes APARs PN67179 and PN70102)

All DB2/MVS V4.1 systems
  PTF UN76674 (fixes APARs PN70102 and PQ03146)

All DB2 for OS/390 systems
  Apply ESO tape 9802 (February, 1998)
  Exclude the following PTFs from ESO tape 9802:
    UQ12940
    UQ12897
    UQ13903
  Install the following additional PTFs:
    UQ15503
    UQ15036
  Install the fix for APAR PQ15977

OS/390 TCP/IP
  PTF PQ14253 (V3R3, V3R4)
  PTF PQ14383 (V3R3)
```

- With DB2 for VSE & VM V3:

```
The following APAR fixes are required:
  PQ01680
  PQ03829
  VM60922
  VM61072
  VM61194
```

Database Manager Configuration Parameters

The following database manager configuration parameters are required in order to use the DBS Syncpoint Manager (SPM), on workstations where DB2 Connect Enterprise Edition is installed:

SPM_LOG_PATH

This parameter specifies the directory where the SPM logs are written. By default, the logs are written to the sqllib directory, which in a

high-volume transaction environment can cause an I/O bottleneck. Use this parameter to have the SPM logs placed on a faster disk than the current sqllib directory.

SPM_LOG_FILE_SZ
Determines the size of the SPM log file, in 4K pages. For more information about SPM logging, see "SPM Logging" on page 350. The default value is 256.

SPM_MAX_RESYNC
Determines the number of simultaneous resynchronization operations with the host or AS/400 database servers.

SPM_NAME
Value used by this DB2 instance to uniquely identify itself for multisite updates with host or AS/400 database servers. The default value for this parameter is a variant of your TCP/IP hostname.

If SNA connectivity is being used, this parameter represents the local LU alias used for communications. It must not be the control point LU. This parameter must be set in order for SPM to start when the database manager is started.

If TCP/IP connectivity is being used with the host system, ensure this name is unique in your network. The suggested value is the TCP/IP hostname of the machine on which the DB2 instance is installed. The generated default value is normally sufficiently unique.

See "Configuring Communications Server for OS/2" on page 346 for additional information about the use of this parameter with OS/2. See "Configuring Communications Server for Windows NT" on page 348 for additional information about the use of this parameter with Windows NT. See "Configuring Communications Server for AIX" on page 348 for additional information about the use of this parameter with AIX.

For more details about these parameters, see the *Administration Guide*.

Other Requirements

In addition to having the correct software prerequisites installed, additional planning is required in order to assign LU names for use by the SPM and its partners if you are using SNA connectivity. See "Setting up SNA Communications for the DB2 Syncpoint Manager (SPM)" on page 346 for further information about what actions may be required in your installation.

Configuring the SPM

This section describes the steps required to set up the SPM on a DB2 Connect or DB2 Universal Database server, and to configure remote clients and remote servers. It also lists considerations that apply with respect to the location of the transaction manager database, which may reside on the DB2 Connect server (the recommended location), or a DB2 Universal Database server.

Applications Which Access Host or AS/400 Database Servers

Applications Which Use DB2 Universal Database to Coordinate Multisite Updates

In this situation, only SNA connectivity requires the DB2 Syncpoint Manager. TCP/IP connectivity does not require the SPM for multisite updates. To setup the DB2 Syncpoint manager, complete the following steps:

Step 1. Ensure your host or AS/400 database servers are enabled for multisite update. On DB2 for MVS and DB2 for OS/390 the VTAM APPL statement for your DB2 location must have SYNCLVL=SYNCPT. For other host or AS/400 platforms, please see the online *Connectivity Supplement*.

Step 2. Install the DB2 Universal Database product which provides the DB2 SPM. This could be either DB2 Connect Enterprise Edition, DB2 Universal Database Enterprise Edition, or DB2 Universal Database Enterprise Edition - Extended.

a. Install DB2 Connect Enterprise Edition or DB2 Universal Database Enterprise Edition or Enterprise Edition - Extended to provide multisite update support with host or AS/400 database servers.

b. Create a database instance on the same system. For example, you can use the default instance DB2, or use the following command to create a new instance:

```
db2icrt myinstance
```

c. Supply licensing information as required.

d. Create a TM database. Any DB2 Universal Database Version 5 or greater can be used for this purpose. The TM database should be a local database on the DB2 Connect workstation. For example, to create the TM database, enter the following command in the command line processor:

```
create Database TMB alias TMB
```

Note: If you have only DB2 Connect Enterprise Edition installed, you are licensed to create one database that can be used as the TM database.

e. If the TM database is local, update the *TM_DATABASE* Database Manager Configuration parameter with the name of the TM database. For example:

```
update database manager configuration using tm_database TMB
```

f. Configure SNA communications as required. See "Setting up SNA Communications for the DB2 Syncpoint Manager (SPM)" on page 346.

The configuration will be easier if the SPM_NAME value is the same as the LU name.

g. Determine the value to be specified for the SPM_NAME database manager configuration parameter, and values for the SPM_LOG_FILE_SZ and SPM_MAX_RESYNC database manager configuration parameters if the defaults are not appropriate for your situation. The default value for the SPM_NAME is a variant of the TCP/IP hostname. See "Database Manager Configuration Parameters" on page 340 for further details.

h. If the default value is not acceptable, update SPM_NAME on the DB2 Connect or DB2 Universal Database system. For example, you can use the following command:

```
update database manager configuration using spm_name SPMNAME
```

i. Ensure that the registry value db2comm includes the value APPC.

j. Stop and restart the database manager on the DB2 Connect workstation/DB2 Universal Database server to enable the SPM.

Step 3. The database administrator must also perform the following steps at each system where a DB2 client will use DB2 Syncpoint services when accessing a host or AS/400 database server.

a. Update the *TM_DATABASE* Database Manager Configuration parameter with the name of the TM database. For example:

```
update database manager configuration using tm_database TMB
```

b. Configure communications as required for the DB2 client to connect to the server..

c. Stop and restart the database manager on the application requester.

Applications Which Use an XA Compliant TP Monitor to Coordinate the Multisite Update:

Examples of XA compliant TP monitors include:

- IBM TX Series CICS
- IBM TX Series Encina Monitor
- BEA TopEnd
- Microsoft Transaction Server

- BEA Tuxedo

In this situation, SNA and TCP/IP connectivity requires the DB2 Syncpoint Manager. Both SNA and TCP/IP connectivity can be simultaneously supported by the DB2 Syncpoint Manager.

1. On the workstation which will be used:
   a. Install DB2 Connect Enterprise Edition or DB2 Universal Database Enterprise Edition or Enterprise Edition - Extended to provide multisite update support with host or AS/400 database servers.
   b. Create a database instance on the same system. For example, you can use the default instance DB2, or use the following command to create a new instance:

      ```
      db2icrt myinstance
      ```

   c. Supply licensing information as required.
   d. If SNA connectivity
      1) Ensure that the registry value db2comm includes the value APPC.
      2) Configure SNA communications as required. See "Setting up SNA Communications for the DB2 Syncpoint Manager (SPM)" on page 346.

         The configuration will be easier if the SPM_NAME value is the same as the LU name.

   e. If TCP/IP connectivity

      You must ensure the DB2 TCP/IP Listener is activated. Do the following
      1) Ensure that the registry value db2comm includes the value TCPIP.
      2) Ensure the Database Manager Configuration parameter SVCENAME is set to the proper TCP/IP port.

   f. Determine the value to be specified for the *SPM_NAME* database manager configuration parameter, and optionally values for the *SPM_LOG_FILE_SZ* and *SPM_MAX_RESYNC* database manager configuration parameters if the defaults are not appropriate for your situation. Ensure that the *SPM_NAME* value is unique within your network. The *SPM_NAME* value will be set to a variant of the TCP/IP hostname when the instance is created.

      If only SNA connectivity is being used, the *SPM_NAME* value must be the local LU alias used for communications with the host system. If only TCP/IP connectivity is being used, then the suggested value for *SPM_NAME* is the TCP/IP hostname of the machine on which the DB2 instance is installed. If both SNA and TCP/IP are used, the *SPM_NAME* value must be the local LU alias of the local LU used for communications with the host system.

See “Database Manager Configuration Parameters” on page 340for further details.

g. If the default value is not acceptable, update SPM_NAME on the DB2 Connect workstation/DB2 Universal Database server. For example, you can use the following command:

```
update database manager configuration using spm_name SPMNAME
```

h. Stop and restart the database manager on the DB2 Connect workstation/DB2 Universal Database server to activate the SPM.

Host or AS/400 Applications Which Access a DB2 Universal Database Server in a Multisite Update

In this situation, SNA connectivity is the only type supported. The DB2 Syncpoint Manager is required to permit multisite update. DB2 Universal Database does not support multisite update from host or AS/400 database clients using TCP/IP connectivity.

The database server which is being accessed from the host or AS/400 database client does not have to be local to the workstation which has the DB2 Syncpoint Manager. The host or AS/400 database client could connect to a DB2 UDB server using the DB2 Syncpoint Manager workstation as an interim server. This allows you to isolate the DB2 Syncpoint Manager workstation in a secure environment while the actual DB2 UDB Servers are remote in your organisation. This also permits UDB servers which do not have SNA Syncpoint Support to participate in a multisite update originating from a host or AS/400 database client.

The steps are as follwos:

1. On the workstation which has the DB2 Syncpoint Manager
   a. Install DB2 Universal Database Enterprise Edition or Enterprise Edition - Extended to provide multisite update support with host or AS/400 database clients.
   b. Create a database instance on the same system. For example, you can use the default instance DB2, or use the following command to create a new instance:

      ```
      db2icrt myinstance
      ```

   c. Supply licensing information as required.
   d. Ensure that the registry value db2comm includes the value APPC.
   e. Configure SNA communications as required. See “Setting up SNA Communications for the DB2 Syncpoint Manager (SPM)” on page 346.

      The configuration will be easier if the SPM_NAME value is the same as the LU name, and the SPM uses the same LU as the DB2 Connect workstation.

f. Determine the value to be specified for the SPM_NAME database manager configuration parameter, and values for the SPM_LOG_FILE_SZ and SPM_MAX_RESYNC database manager configuration parameters if the defaults are not appropriate for your situation. The default for SPM_NAME will be a variant of the TCP/IP hostname. See "Database Manager Configuration Parameters" on page 340 for further details.

g. If the default value is not acceptable, update SPM_NAME on the DB2 Universal Database server. For example, you can use the following command:

```
update database manager configuration using spm_name SPMNAME
```

h. Configure communications as required for this DB2 workstation to connect to remote DB2 UDB servers, if any.

i. Configure communications as required for remote DB2 UDB Servers to connect to this DB2 Syncpoint Manager workstation.

j. Stop and restart the database manager on the DB2 Universal Database server to start the SPM for the first time.

You should be able to connect to the remote DB2 UDB servers from this DB2 Syncpoint Manager workstation.

Setting up SNA Communications for the DB2 Syncpoint Manager (SPM)

Multisite updates using SNA connectivity is provided on OS/2, AIX, and Windows NT only.

The supported SNA stacks are:

- IBM eNetwork Communications Server for OS/2 Version 5
- IBM eNetwork Communications Server for AIX Version 5.0.2.4
- IBM eNetwork Communications Server for Windows NT Version 5.01.

Configuring Communications Server for OS/2

The following example assumes:

- IBM Communications Server for OS/2 V5 is installed
- SPMNAME is the local LU name
- SPM_NAME is set to SPMNAME.

These requirements must be met:

1. The SPM_NAME Database Manager Configuration parameter must specify the local LU to be used by SPM. The SPM_NAME cannot represent the Control Point LU. There is no need to configure the LU 6.2 profile. DB2 will dynamically create or modify the necessary profiles. Ensure the LU represented by the SPM_NAME parameter is only used by DB2.

SPM_NAME can be set using the following command:

```
update database manager configuration using spm_name SPMNAME
```

2. If your DB2 Universal Database SPM workstation will be used as a server accepting DB2 Universal Database client connections through APPC, you will need to ensure that you configure the database manager configuration parameter *tpname* with an appropriate value. For example:

   ```
   db2 update dbm cfg using tpname NYSERVER
   ```

Note: If all of the following apply to your environment:

- You are using the SPM workstation to connect to host or AS/400 database servers.
- You are using the SPM workstation to accept connections from host or AS/400 database clients.
- You have DB2 Universal Database clients accessing the SPM workstation through APPC.

then the DB2 Universal Database clients and the host or AS/400 database clients must:

- Use the same transaction program name to connect to the SPM workstation.
- Use the SPM_NAME LU as their partner logical unit.

Communications Server Operational Considerations

Communications Server for OS/2 requires that syncpoint LUs be registered by the SPM before they can be used. Thus DB2 must be started explicitly beforehand so that the SPM can register the syncpoint LU. Otherwise you may encounter the following message:

```
SQL0859N Access to the Transaction Manager Database failed with
SQLCODE "-1032".  SQLSTATE=08502.
```

To execute a one-phase commit successfully when DB2 is not started, the default LU should specify a non-SPM LU. You can use the APPCLLU environment variable to specify an LU name that is not the same as the value in SPM_NAME. Otherwise you may encounter the following message:

```
SQL30081N  A communication error has been detected. Communication protocol
being used: "APPC". Communication API being used: "CPI-C". Location where
the error was detected: "". Communication function detecting the error:
"xcstp". Protocol specific error codes: "20", "*", "*".  SQLSTATE=08001
```

Configuring Communications Server for AIX

The following example assumes:

- IBM Communications Server for AIX V5.0.2.4 is installed
- SPMNAME is the local LU name
- SPM_NAME is set to SPMNAME.

These requirements must be met:

1. The SPM_NAME Database Manager Configuration parameter must specify the local LU to be used by SPM. The SPM_NAME cannot represent the Control Point LU. There is no need to configure the LU 6.2 profile. DB2 will dynamically create or modify the necessary profiles. Ensure the LU represented by the SPM_NAME parameter is only used by DB2.

   SPM_NAME can be set using the following command:

   ```
   update database manager configuration using spm_name SPMNAME
   ```

2. If your DB2 Universal Database SPM workstation will be used as a server accepting DB2 Universal Database client connections through APPC, you will need to ensure that you configure the database manager configuration parameter *tpname* with an appropriate value. For example:

   ```
   db2 update dbm cfg using tpname NYSERVER
   ```

Note: If all of the following apply to your environment:

- You are using the SPM workstation to connect to host or AS/400 database servers.
- You are using the SPM workstation to accept connections from host or AS/400 database clients.
- You have DB2 Universal Database clients accessing the SPM workstation through APPC.

then the DB2 Universal Database clients and the host or AS/400 database clients must:

- Use the same transaction program name to connect to the SPM workstation.
- Use the SPM_NAME LU as their partner logical unit.

Configuring Communications Server for Windows NT

The following example assumes:

- IBM Communications Server for Windows NT V5.01 is installed
- SPMNAME is the local LU name
- SPM_NAME is set to SPMNAME.

These requirements must be met:

1. The SPM_NAME Database Manager Configuration parameter must specify the local LU to be used by SPM. The SPM_NAME cannot represent the Control Point LU. There is no need to configure the LU 6.2 profile. DB2 will dynamically create or modify the necessary profiles. Ensure the LU represented by the SPM_NAME parameter is only used by DB2.

   SPM_NAME can be set using the following command:

   ```
   update database manager configuration using spm_name SPMNAME
   ```

2. If your DB2 Universal Database SPM workstation will be used as a server accepting DB2 Universal Database client connections through APPC, you will need to ensure that you configure the database manager configuration parameter *tpname* with an appropriate value. For example:

   ```
   db2 update dbm cfg using tpname NYSERVER
   ```

Note: If all of the following apply to your environment:

- You are using the SPM workstation to connect to host or AS/400 database servers.
- You are using the SPM workstation to accept connections from host or AS/400 database clients.
- You have DB2 Universal Database clients accessing the SPM workstation through APPC.

then the DB2 Universal Database clients and the host or AS/400 database clients must:

- Use the same transaction program name to connect to the SPM workstation.
- Use the SPM_NAME LU as their partner logical unit.

Note: It is very important that CS/NT be started before DB2. DB2 will wait for CS/NT to start before completing its own startup procedures. Therefore, if CS/NT is not started before DB2, DB2 will appear to ″hang″ upon starting. DB2 is actually waiting for CS/NT to start. It is strongly recommended that you register CS/NT as a Windows NT service, as outlined in the next section, to avoid this problem. CS/NT will then be started automatically before DB2 starts.

Registering CS/NT as a Windows NT Service

After installing IBM Communications Server for Windows NT (CS/NT) you should register it as a Windows NT Service. This will automatically start CS/NT when the machine is booted.

To register CS/NT as an Windows NT Service execute the following command. Either:

```
csstart  -a
```

To autostart CS/NT with the default configuration file, or:

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

Where **c:\ibmcs\private\your.acg** is the name of the non-default CS/NT configuration file you wish to be used.

The next time your machine is rebooted CS/NT will be started automatically with the requested configuration file.

SPM Migration Considerations

When migrating from DB2 or DDCS Version 2 to DB2 Connect or DB2 UDB Version 6, first ensure that there are no indoubt transactions. You can verify this by entering the following command in the command line processor:

```
list drda indoubt transactions
```

Note that, before issuing `LIST DRDA INDOUBT TRANSACTIONS`, the application must be connected to the Syncpoint Manager (SPM) instance - use the SPM_NAME database manager configuration parameter as the dbalias on the CONNECT statement. If any transactions are listed, ensure that automatic resolution occurs, or - if automatic resolution is not possible - you can force the resolution by issuing the following command:

```
list drda indoubt transactions with prompting
```

SPM Logging

The SPM maintains a log in which it records information about connections, and UOW status. This log is contained in the *spmlog* directory which is a sub-directory of *sqllib*. This directory is created the first time that the SPM is started.

The SPM log consists of the following files and directories:

- `SPMLOG` - this is the log directory, under `sqllib`.
- `SPMLOG.LCF` - this is the log control file.
- `SPMLOGSD` - this is a sub-directory under `spmlog` which contains the primary and secondary log files:
  - `SPM00000.LOG` - the first primary log file.
  - `SPM00001.LOG` - the second primary log file.

– SPM00002.LOG - the third primary log file.
– SPM00003.LOG - the first secondary log file.
– SPM00004.LOG - the second secondary log file.

SPM Log File Size

The SPM_LOG_FILE_SIZE database manager configuration parameter defines the number of 4K pages of each primary and secondary log file, and it can have the following values:

- Minimum - the minimum value is 4.
- Maximum - the maximum value is 1000 (4M bytes).
- Default - the default is 256 (1M byte).

The log normally only uses the primary extents which are allocated when the log is created. Secondary extents are allocated on demand when log space is constrained. Secondary extents are usually deleted during normal shutdown of the logger.

The SPM log size should be large enough so that performance is not adversely affected but small enough so that space is not wasted. The size that is required depends on the number of transactions using protected conversations and how often commit/rollback is issued. The SPM takes a checkpoint every time DB2 is started and stopped and when the log is 50% full. If the log is too small, the checkpoint process may be invoked often enough to impact performance. The log is probably too small if secondary log files are allocated. If the log is too large, then space will be wasted.

You should use the default log size to begin with.

Changing the SPM Log File Size

The size of the SPM log can be changed when the database manager is stopped and there are no in doubt transactions. The following procedure can be used:

1. Use the LIST DRDA INDOUBT TRANSACTIONS command to determine whether there are indoubt transactions for which the SPM is responsible. Note that, before issuing LIST DRDA INDOUBT TRANSACTIONS, the application must be connected to the Syncpoint Manager (SPM) instance - use the SPM_NAME database manager configuration parameter as the dbalias on the CONNECT statement. DB2 clients must be at Version 2.1.2 or higher to use the WITH PROMPTING capability of LIST INDOUBT.
2. If there are some indoubt transactions, stop the database manager:

```
db2stop
```

3. Update the Database Manager Configuration with a new SPM_LOG_FILE_SIZE value. For example, to double the log size from the default of 256 * 4K pages:

   ```
   db2 update dbm configuration using spm_log_file_size 512
   ```

4. Go to the sqllib directory and delete the current SPM log. For example, on AIX:

   ```
   rm -fr spmlog
   ```

5. Start the database manager:

   ```
   db2start
   ```

   A new SPM log of the specified size will be created during DB2 start-up.

Performance Characteristics

Distributed two-phase commit requires more request/response message exchanges than single-phase commit, which requires only a single message exchange. The overhead of protected conversations, even for read only applications, depends on the SNA LU 6.2 Syncpoint options supported by the server.

Memory Utilization

To support the LU 6.2 Syncpoint function, most of the control blocks used by the SPM are in global memory. The amount of global memory required is dependent on the number of protected conversations active at any one time. The amount of memory required can also be affected by the number of resynchronization connections that are pending.

SPM Protocol Violation Records

Refer to the *Message Reference*.

Part 5. Configuring DB2 UDB as a DRDA Application Server

© Copyright IBM Corp. 1993, 1999

Chapter 17. 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 Extended Enterprise Edition
Your host or AS/400 application can use the parallel processing power of DB2 Universal Database for CPU intensive queries.

Access Distributed Data
Your host or AS/400 application can access distributed data stored in departmental DB2 Universal Database servers.

Supported Clients

The following database products can access DB2 Universal Database servers:

- DB2 for MVS/ESA Version 3.1 (or higher)

  For information on setting up the connection from DB2 for MVS/ESA to a DB2 Universal Database server, see the instructions in "Configuration Steps for DB2 Universal Database Server" on page 356.

- 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 356.

- DB2 for AS/400 Version 5 (or higher)

  For information on setting up the connection from DB2 for AS/400 to a DB2 Universal Database server, refer to the *DB2 Connectivity Supplement*.

- DB2 for VM & VSE (SQL/DS) Version 3.2 and Version 4 (or higher)

  For information on setting up the connection from DB2 for VM & VSE to a DB2 Universal Database server, refer to the *DB2 Connectivity Supplement*.

© Copyright IBM Corp. 1993, 1999

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
Solaris	TCP/IP, APPC
OS/2	TCP/IP, APPC, APPC Multisite Update
Windows NT	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 5.0.2.4
- IBM eNetwork Communications Server for Windows NT Version 5.01
- IBM eNetwork Communications Server for OS/2 Version 5

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 9. Using the Command Line Processor to Configure Server Communications” on page 119. If you want to use Multisite Update, see “Chapter 15. Enabling Multisite Updates (Two-Phase Commit)” on page 333.)
5. Stop and restart the DB2 Universal Database instance to which you wish to connect. Enter the following commands:

```
db2stop
db2start
```

6. Test the host-server DRDA connection by logging onto TSO and using DB2I/SPUFI.

Using the DB2 Universal Database Server from Host or AS/400 Clients

Connections from host or AS/400 database clients are treated like any other connection to the DB2 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 28 identifies which functions are implemented in the AS on the DB2 Universal Database server. The subsequent table lists the supported bind options.

Table 28. 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 multipe row result sets	O	Yes

Note: * Certain required functions are not supported.

Bind Options Supported by the DB2 DRDA Application Server

Table 29. 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	**Null**	Yes	VERSION			VERSION
	Any other value	No				
Bind Existence Checking	**Object existence optional**	No	VALIDATE (**RUN**)[b]	**NOEXIST**	GENLVL(**10**, 11-40)	VALIDATE **RUN**
	Object existence required	Yes	VALIDATE (BIND)[b]	EXIST	GENLVL(00-09)	VALIDATE BIND
Package Replacement Option	**Replacement allowed**	Yes	ACTION (**REPLACE**)	**REPLACE**	REPLACE(***YES**)	ACTION **REPLACE**
	Replacement not allowed	No	ACTION(ADD)	NEW	REPLACE(*NO)	ACTION ADD
Package Authorization Option	**Keep authorizations**	Yes		**KEEP**		RETAIN **YES**
	Revoke authorizations	No		REVOKE		RETAIN NO
Statement String Delimiter (Note 2)	Apostrophe	Yes	**APOSTSQL**	**SQLAPOST**	OPTION([...] *APOSTSQL) (Note 3)	STRDEL APOSTROPHE
	Double quote	No	QUOTESQL	SQLQUOTE	OPTION([...] *QUOTESQL) (Note 4)	STRDEL QUOTE
Statement Decimal Delimiter (Note 5)	Period	Yes	**PERIOD**	**PERIOD**	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)	**ISO**	Yes	DATE(ISO) (Note 8)	DATE(**ISO**)	DATFMT(*ISO) (Note 8)	DATETIME **ISO** (Note 9)
	USA	Yes	DATE(USA)	DATE(USA)	DATFMT(*USA)	DATETIME USA
	EUR	Yes	DATE(EUR)	DATE(EUR)	DATFMT(*EUR)	DATETIME EUR
	JIS	Yes	DATE(JIS)	DATE(JIS)	DATFMT(*JIS)	DATETIME JIS
Time Format (Note 7)	**ISO**	Yes	TIME(ISO) (Note 8)	TIME(**ISO**)	TIMFMT(*ISO) (Note 8)	DATETIME **ISO** (Note 9)
	USA	Yes	TIME(USA)	TIME(USA)	TIMFMT(*USA)	DATETIME USA
	EUR	Yes	TIME(EUR)	TIME(EUR)	TIMFMT(*EUR)	DATETIME EUR
	JIS	Yes	TIME(JIS)	TIME(JIS)	TIMFMT(*JIS)	DATETIME JIS

Table 29. 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
Package Isolation Level (Note 10)	Repeatable read	Yes	ISOLATION(**RR**)[b]	ISOLATION(**RR**)		ISOLATION RR
	Read Stability (All)	Yes		ISOLATION(RS)	COMMIT(*ALL)	ISOLATION RS
	Cursor stability	Yes	ISOLATION(CS)[b]	ISOLATION(CS)	COMMIT(*CS)	ISOLATION **CS**
	Uncommitted Read (Change)	Yes		ISOLATION(UR)	COMMIT(***CHG**)	ISOLATION UR
	No commit	No (Note 11)			COMMIT(*NONE)	ISOLATION NC
Bind Creation Control	**No errors allowed**	Yes	SQLERROR (**NOPACKAGE**)[b]	**NOCHECK**	OPTION([...] *GEN) GENLVL(00-09, **10**, 11-20)	SQLERROR **NOPACKAGE**
	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	**No SQL statements**	Yes	EXPLAIN(**NO**)[b]	EXPLAIN(**NO**)		EXPLAIN **NO**
	All explainable SQL statements	No	EXPLAIN(YES)[b]	EXPLAIN(YES)		EXPLAIN YES
Package Owner Identifier	<***Authorization ID***>	Yes	OWNER[b]	OWNER		OWNER
	Any other value	No				
RDB Release Option	**Release at commit**	Yes	RELEASE (**COMMIT**)[b]	RELEASE (**COMMIT**)		RELEASE **COMMIT**
	Release at conversation deallocation	No	RELEASE (DEALLOCATE)[b]	RELEASE (DEALLOCATE)		RELEASE DEALLOCATE
Default RDB Collection ID	<***Authorization ID***>	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	**Fixed row**	Yes	CURRENTDATA (**YES**)[b]	SBLOCK	ALWBLK(***READ**)	BLOCKING **UNAMBIG**
	Limited block	Yes	CURRENTDATA (NO)[b]	BLOCK	ALWBLK (*ALLREAD)	BLOCKING ALL
	Forced fixed row	Yes		**NOBLOCK**	ALWBLK(*NONE)	BLOCKING NO
Package Default Char. Subtype						
	Use system default	Yes				CHARSUB **DEFAULT**

Table 29. 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	**Value specified when DB2 database was created**	Yes		CCSIDSBCS() CCSIDGRAPHIC() CCSIDMIXED()		CCSIDS 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	**Null**	Yes	REPLVER[b]			REPLVER
	Any other value	No				
Generic Bind Option	**Null**	No				GENERIC
	Any other value	No				
Package Authorization Rule	**Requester**	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	**1**	No				DEGREE 1
	n	No				DEGREE n
	ANY	No				DEGREE ANY

Table 29. 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/sqllib/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 `sqllib`, then no special setup is required for RXSQL; refer to the *SQL/DS Procedures Language Interface Installation* manual for details.

Security and Auditability

Under APPC, DB2 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. Thin Client and Thin Connect Architecture

© Copyright IBM Corp. 1993, 1999

Chapter 18. Distributed Thin Client Installation

You can install a DB2 client for Windows 9x or Windows NT on a code server, and have Thin Client workstations access the code across a LAN connection. These 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.

The benefits of a Thin Client installation are:

- Reduced disk space required for each Thin Client; you can save from 16–112 MB per client workstation.
- Easier software changes and updates; you only have one copy of the DB2 product to change and update.

The drawback of a Thin Client installation is a potential loss of system performance, since 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.

Installing and Configuring Thin Client Workstations to use a Code Server

This section describes how to set up and configure a Thin Client to run on your network.

Before You Begin

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 account to perform the installation.

Windows 9x
Any valid Windows 9x user.

Windows NT
Any user account that has the *"Log on as a service"* advanced user right on the machine where the account is defined. For more information on user rights, refer to the Windows NT online help.

© Copyright IBM Corp. 1993, 1999

Performing the Installation

Step 1. Install the DB2 Client on the Code Server

To install the DB2 client on a code server, perform the following steps:

Step 1. Log on to the system with a user account that meets the requirements for installing DB2 Connect Personal Edition. For more information, see "Before You Begin" on page 369.

Step 2. Shut down any programs that are running 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 auto-start, see the tip that follows.

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

a. Click on **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 36 on page 542 lists the code for each available language.

c. Click on **OK**.

Step 4. The Welcome window opens. Click on the **Next** push button.

Step 5. The Select Products window opens. Select the **DB2 client** check box and click on the **Next** push button.

Step 6. The Select Installation Type window opens. Click on the **Custom** graphic button.

Step 7. The Select Components window opens. Select the components that you want to install. You must select the **Thin Client Code Server** component to enable a Thin Client workstation to access the DB2 client on the code server.

Step 8. Respond to the setup program's remaining prompts. Online help can walk you through the remaining steps of the installation process. You can invoke online help by clicking on the **Help** push button, or by pressing the **F1** key, at any time.

Step 2. Create a Windows 9x Code Base on a Windows NT Code Server

This step describes how to set up a Windows NT code server to service Windows 9x Thin Client workstations.

If you are not planning on running a Windows 9x Thin Client workstation from a Windows NT code server, you can skip this step and proceed to "Step 3. Create a Network Share on the Code Server".

To have a Windows NT code server service a Thin Client workstation on a Windows 9x machine, perform the following steps:

Step 1. Create a directory on the Windows NT code server that will be used to service Windows 9x Thin Client workstations by entering the following command:

```
md d:\sqllib9x
```

where: *d:* represents a local hard drive.

Step 2. Copy the DB2 client directory (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 where the DB2 client was installed
- *d:* represents the drive where the `sqllib9x` directory was created in Step 1.

Step 3. Enter the **c:\sqllib\bin\db2thn9x.bat** command (where *c:* is the drive where you installed the code server) to enable this machine to service Windows 9x Thin Clients.

You now have two code bases on your Windows NT code server. If you are installing a Thin Client workstation on a Windows NT machine, use the Windows NT code base (for example, `c:\sqllib`). If you are installing a Thin Client workstation on a Windows 9x machine, use the Windows 9x code base (for example, `d:\sqllib9x`).

If you are logged onto a Windows 9x Thin Client workstation, that is running code off of a Windows NT 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 code server.

Step 3. Create a Network Share on the Code Server

The code server must be installed in a directory that can be accessed in *READ* mode by all Thin Client workstations.

To make the code server available to all Thin Client workstations in *READ* mode, perform the following steps:

Step 1. Click on **Start** and select **Programs->Windows Explorer**.

Step 2. Select the directory where you installed the DB2 product. For example, `c:\sqllib`.

Step 3. Select **File->Properties** from the menu bar.

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.

Step 7. Specify *READ* access for all users as follows:

On Windows 9x, you do not need to specify type of access when you set up a share. By default, everyone is granted read access. If you are setting up a code server on a Windows 9x workstation skip this step and go to "Step 4. Create a Response File for a Thin Client Installation".

a. Click the **Permissions** button. The **Access Through Share Permissions** window opens.

b. In the **Name** box, select **Everyone**.

c. Click on the **Type of Access** drop down box and select **Read**.

d. Click on **OK** until all windows are closed.

Step 4. Create a Response File for a Thin Client Installation

During a regular installation, you provide the information needed to install a DB2 client and configure its environment. During a distributed installation, this information is provided in the form of keywords and values in a response file. For a DB2 client installation, there is a ready-to-use sample response file, `db2thin.rsp`, with default settings for the most common installation type.

The `c:\sqllib\thnsetup` directory on the code server, where *c:* represents the drive where you installed the code server, contains all the files needed to install a Thin Client workstation, including a sample response file (called `db2thin.rsp`). If you need to customize a response file, perform the following steps:

Step 1. Open the `db2thin.rsp` sample response file, located in the `c:\sqllib\thnsetup` directory on the code server (where *c:* represents the drive where you installed the DB2 client).

The response file contains:

- Keywords unique to installation
- Registry value/environment variable settings
- Database manager configuration parameter settings.

Step 2. Customize the sample response file for the installation type that you want to perform. To activate an item in a response file:

a. Remove the asterisk (*) to the left of the variable.

b. Erase the current setting to the right of the equal sign (=).

c. Enter the new setting to the right of the equal sign (=); possible settings are listed to the right of the current setting.

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
⋮
```

The **TYPE** keyword specifies the type of install that you want.

- Install type 0 specifies a compact install. This option provides you with access to the client code, but none of the other components are accessible from the Thin Client workstation.
- Install type 1 specifies a typical install, and is the default setting. With this option, all components defined on the code server are accessible from the Thin Client workstation.
- Install type 2 specifies a custom install. This option allows you to specify specific components to install. Set the **TYPE** keyword to 2 (Custom Install) and activate the **COMP** keyword for each component you want to be accessible from the client workstation.

Notes:

a. Currently, the DB2 Client configuration files must be stored locally in a writable directory on the Thin Client workstation. By default, the c:\cfg directory will be created (as specified by the *DB2INSTPROF* keyword in the response file) to store the DB2 client configuration files.

b. If you want to allow any user to add a database to the system, you must set the *CATALOG_NOAUTH* database manager configuration parameter to *YES* in the response file or by updating the database manager configuration file. For information on how to update this database manager configuration parameter, refer to the *Administration Guide*.

c. You will need to reboot the target workstation after completing the installation. To have this done automatically, set the *REBOOT* keyword to *Yes*.

Step 3. Exit the file. If you have made any changes, save the file under a new file name (using the file extension `.rsp`) to preserve the original sample response file.

Step 5. Make the Setup Program Accessible to the Target Workstation

This step gives a Thin Client workstation access to the DB2 client code installed on the code server.

From the Thin Client workstation, enter the **net use** command to attach the shared directory that you created in "Step 3. Create a Network Share on the Code Server" on page 380 as follows:

```
net use x: \\computer_name\directory_sharename /USER:domain\username
```

where:

- *x:* represents the drive used to connect to the remote shared directory
- *computer_name* represents the computer name of the code server
- *directory_sharename* represents the shared directory on the code server.
- *domain* represents the domain where the user account is defined.
- *username* represents a user that has access to this machine.

For example, to assign a shared directory called `sqllib` on a machine called *myserver*, to the *x:* drive, enter the following command:

```
net use x: \\myserver\sqllib
```

If you are planning on having a DB2 for Windows NT code server service a Windows 9x Thin Client workstation, make sure that you enter the share name for the directory that you created for Windows 9x Thin Client workstations in "Step 2. Create a Windows 9x Code Base on a Windows NT Code Server" on page 371. For example, `sqllib9x`.

Step 6. Install the Thin Client Code on a Target Workstation

This step installs the Thin Client code on the target workstation, and sets up the required links to the code server. To install a Thin Client workstation, perform the following steps:

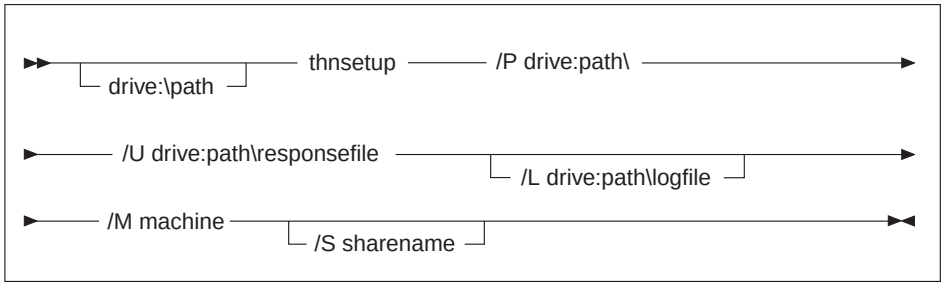
Step 1. At the workstation where the Thin Client code will be installed, log on with a user account that you created to perform the installation in "Before You Begin" on page 369.

Step 2. Run the setup program:

a. Click **Start**, and select the **Run** option.

b. In the **Open** field, enter the path to the **thnsetup** command to install a Thin Connect workstation.

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



/P Specifies the path where the DB2 client is installed on the code server. You can specify this parameter as a local directory (if you have entered the **net use** command to attach the shared directory) or as a share name (for example, `\\myserver\sqllib`). If you specify a sharename, the setup program will implicitly perform the **net use** command for you. This parameter is required.

/U Specifies the fully qualified response file name. This parameter is required.

/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 log file name (`db2.log`) is used. This file is 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 required.

/S Specifies the share name of the code server drive or directory where DB2 Thin Client code server is installed. If you specify this parameter, you will not have to perform a **net use** command to the code server each time you reboot your machine.

If this share requires that a user account be entered, specify the share name as: `sharename,userid,password`.

This parameter is optional.

For example, to install a Thin Connect workstation where the shared `sqllib` directory, on a code server called *myserver*, is mapped as drive *x:*, enter the following command:

```
x:\thnsetup\thnsetup /P x: /U x:\thnsetup\db2thin.rsp /M myserver
```

Step 3. Check the messages in the log file for installation errors when the installation finishes.

Step 4. After the installation, the workstation must be rebooted before you use the Thin Client workstation.

Chapter 19. DB2 Thin Connect Workstations in a Windows 32-bit Operating System Environment

You can install DB2 Connect Personal Edition Version 6 for Windows 9x or Windows NT on a code server, and have Thin Connect workstations access the code across a LAN connection. These Thin Connect workstations function like any other DB2 Connect Personal Edition workstations. In the setup, the main difference is that the DB2 Connect Personal Edition code is installed on a code server, and not individually on each workstation. Any Thin Connect workstation needs only a minimal configuration to set its parameters and establish links to a code server.

The benefits of a Thin Connect workstation are:

- Reduced disk space required for each Thin Connect workstation; you can save from 16–112 MB per workstation.
- Easier software changes and updates; you only have one copy of the DB2 product to change and update.

The drawback of a Thin Connect installation is a potential loss of system performance, since DB2 programs are loaded from a code server instead of the workstation. However, any loss will depend on network load, network speed, and code server speed and load.

Installing and Configuring Thin Connect Workstations to use a Code Server

This section describes how to set up and configure a Thin Connect workstation to run on your network.

Before You Begin

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 account to perform the installation.

Windows 9x
Any valid Windows 9x user.

Windows NT
Any user account that does not belong to the *Guests* group on the machine where the account is defined.

© Copyright IBM Corp. 1993, 1999

To set up a code server and install one or more Thin Connect workstations to access it, perform the following steps:

- "Step 1. Install DB2 Connect Personal Edition on the Code Server".
- "Step 2. Create a Windows 9x Code Base on a Windows NT Code Server" on page 379.
- "Step 3. Create a Network Share on the Code Server" on page 380.
- "Step 4. Create a Response File for a Thin Connect Installation" on page 381.
- "Step 5. Make the Thin Connect Workstation Setup Program Accessible to the Target Workstation" on page 383.
- "Step 6. Install the Thin Connect Code on a Target Workstation" on page 383.

Performing the Installation

Step 1. Install DB2 Connect Personal Edition on the Code Server

To install DB2 Connect Personal Edition on a code server, perform the following steps:

Step 1. Log on to the system with a user account that meets the requirements for installing DB2 Connect Personal Edition. For more information, see "Before You Begin" on page 377.

Step 2. Shut down any programs that are running 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 auto-start, see the tip that follows.

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

a. Click on **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 36 on page 542 lists the code for each available language.

c. Click on **OK**.

Step 4. Insert the CD-ROM into the drive. The auto-run feature will automatically start the setup program.

Step 5. The Welcome window opens. Click on the **Next** push button.

Step 6. The Select Products window opens. Select the **DB2 Connect Personal Edition** check box and click on the **Next** push button.

Step 7. The Select Installation Type window opens. Click on the **Custom** graphic button.

Step 8. The Select Components window opens. Select the components that you want to install.

You must select the **Thin Connect Code Server** component to enable a Thin Connect workstation to access DB2 Connect Personal Edition on the code server.

Step 9. Respond to the setup program's remaining prompts. Online help can walk you through the remaining steps of the installation process. You can invoke online help by clicking on the **Help** push button, or by pressing the **F1** key, at any time.

Step 2. Create a Windows 9x Code Base on a Windows NT Code Server

This step describes how to set up a Windows NT code server to service Windows 9x Thin Connect workstations.

If you are not planning on running a Windows 9x Thin Connect workstation from a Windows NT code server, you can skip this step and proceed to "Step 3. Create a Network Share on the Code Server" on page 380.

To have a Windows NT code server service a Thin Connect workstation on a Windows 9x machine, perform the following steps:

Step 1. Create a directory on the Windows NT code server that will be used to service Windows 9x Thin Connect workstations by entering the following command:

```
md d:\sqllib9x
```

where: *d:* represents a local hard drive.

Step 2. Copy the DB2 client directory (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 where the Thin Connect product was installed
- *d:* represents the drive where the `sqllib9x` directory was created in Step 1.

Step 3. Enter the **c:\sqllib\bin\db2thn9x.bat** command (where *c:* is the drive where you installed the code server) to enable this machine to service Windows 9x Thin Connect workstations.

You now have two code bases on your Windows NT code server. If you are installing a Thin Connect workstation on a Windows NT machine, use the Windows NT code base (for example, `c:\sqllib`). If you are installing a Thin Connect workstation on a Windows 9x machine, use the Windows 9x code base (for example, `d:\sqllib9x`).

If you are logged onto a Windows 9x Thin Connect workstation, that is running code off of a Windows NT 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 code server.

Step 3. Create a Network Share on the Code Server

The code server must be installed in a directory that can be accessed in *READ* mode by all Thin Connect workstations.

To make the code server available to all Thin Connect workstations in *READ* mode, perform the following steps:

Step 1. Click on **Start** and select **Programs->Windows Explorer**.

Step 2. Select the directory where you installed the DB2 product. For example, `c:\sqllib`.

Step 3. Select **File->Properties** from the menu bar.

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.

Step 7. Specify *READ* access for all users as follows:

On Windows 9x, you not need to specify type of access when you set up a share. By default, everyone is granted read access. If you are setting up a code server on a Windows 9x workstation skip this step and go to "Step 4. Create a Response File for a Thin Connect Installation".

a. Click the **Permissions** button. The **Access Through Share Permissions** window opens.
b. In the **Name** box, select **Everyone**.
c. Click on the **Type of Access** drop down box and select **Read**.
d. Click on **OK** until all windows are closed.

Step 4. Create a Response File for a Thin Connect Installation

During a regular installation, you provide the information needed to install DB2 Connect Personal Edition and configure its environment. During a distributed installation, this information is provided in the form of keywords and values in a response file. DB2 Connect Personal Edition includes a ready-to-use sample response file, `db2thin.rsp`, with default settings for the most common installation type.

The `c:\sqllib\thnsetup` directory on the code server, where *c:* represents the drive where you installed the code server, contains all the files needed to install a Thin Connect workstation, including a sample response file (called `db2thin.rsp`). If you need to customize a response file, perform the following steps:

Step 1. Open the `db2thin.rsp` sample response file, located in the `c:\sqllib\thnsetup` directory on the code server (where *c:* represents the drive where you installed DB2 Connect Personal Edition).

The response file contains:

- Keywords unique to installation
- Registry value/environment variable settings
- Database manager configuration parameter settings.

Step 2. Customize the sample response file for the installation type that you want to perform. To activate an item in a response file:

a. Remove the asterisk (*) to the left of the variable.
b. Erase the current setting to the right of the equal sign (=).
c. Enter the new setting to the right of the equal sign (=); possible settings are listed to the right of the current setting.

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
⋮
```

The **TYPE** keyword specifies the type of install that you want.

- Install type `0` specifies a compact install. This option provides you with access to the Connect code, but none of the other components are accessible from the Thin Connect workstation.
- Install type `1` specifies a typical install, and is the default setting. With this option, all components defined on the code server are accessible from the Thin Connect workstation.
- Install type `2` specifies a custom install. This option allows you to specify specific components to install. Set the **TYPE** keyword to 2 (Custom Install) and activate the **COMP** keyword for each component you want to be accessible from the client workstation.

Notes:

a. Currently, the DB2 Connect configuration files must be stored locally in a writable directory on the Thin Connect workstation. By default, the `c:\cfg` directory will be created (as specified by the *DB2INSTPROF* keyword in the response file) to store the DB2 Connect configuration files.

b. If you want to allow any user to add a database to the system, you must set the *CATALOG_NOAUTH* database manager configuration parameter to *YES* in the response file or by updating the database manager configuration file. For information on how to update this database manager configuration parameter, refer to the *Administration Guide*.

c. You will need to reboot the target workstation after completing the installation. To have this done automatically, set the *REBOOT* keyword to *Yes*.

Step 3. Exit the file. If you have made any changes, save the file under a new file name (using the file extension `.rsp`) to preserve the original sample response file.

Step 5. Make the Thin Connect Workstation Setup Program Accessible to the Target Workstation

This step gives a Thin Connect workstation access to the DB2 Connect Personal Edition code installed on the code server.

From the Thin Connect workstation, enter the **net use** command to attach the shared directory that you created in "Step 3. Create a Network Share on the Code Server" on page 380 as follows:

```
net use x: \\computer_name\directory_sharename /USER:domain\username
```

where:

- *x:* represents the drive used to connect to the remote shared directory
- *computer_name* represents the computer name of the code server
- *directory_sharename* represents the shared directory on the code server.
- *domain* represents the domain where the user account is defined.
- *username* represents a user that has access to this machine.

For example, to assign a shared directory called `sqllib` on a machine called *myserver*, to the *x:* drive, enter the following command:

```
net use x: \\myserver\sqllib
```

If you are planning on having a DB2 for Windows NT code server service a Windows 9x Thin Connect workstation, make sure that you enter the share name for the directory that you created for Windows 9x Thin Connect workstations in "Step 2. Create a Windows 9x Code Base on a Windows NT Code Server" on page 379. For example, `sqllib9x`.

Step 6. Install the Thin Connect Code on a Target Workstation

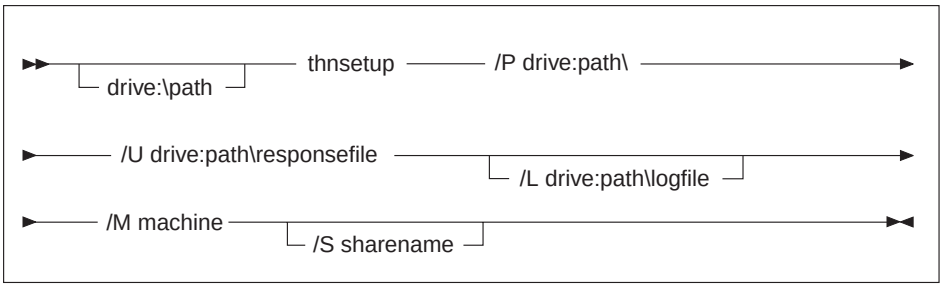
This step installs the Thin Connect code on the target workstation, and sets up the required links to the code server. To install a Thin Connect workstation, perform the following steps:

Step 1. At the workstation where the Thin Client code will be installed, logon with a user account that you created to perform the installation in "Before You Begin" on page 377.

Step 2. Run the setup program:

a. Click **Start**, and select the **Run** option.
b. In the **Open** field, enter the path to the **thnsetup** command to install a Thin Connect workstation.

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



/P Specifies the path where DB2 Connect Personal Edition is installed on the code server. You can specify this parameter as a local directory (if you have entered the **net use** command to attach the shared directory) or as a share name (for example, `\\myserver\sqllib`). If you specify a sharename, the setup program will implicitly perform the **net use** command for you. This parameter is required.

/U Specifies the fully qualified response file name. This parameter is required.

/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 log file name (`db2.log`) is used. This file is 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 required.

/S Specifies the share name of the code server drive or directory where DB2 Thin Connect code server is installed. If you specify this parameter, you will not have to perform a **net use** command to the code server each time you reboot your machine.

If this share requires that a user account be entered, specify the share name as: `sharename,userid,password`.

This parameter is optional.

For example, to install a Thin Connect workstation where the shared `sqllib` directory, on a code server called *myserver*, is mapped as drive *x:*, enter the following command:

```
x:\thnsetup\thnsetup /P x: /U x:\thnsetup\db2thin.rsp /M myserver
```

Step 3. Check the messages in the log file for installation errors when the installation finishes.

Step 4. After the installation, the workstation must be rebooted before you use the Thin Connect workstation.

For information on setting up your DB2 Connect workstation, refer to the appropriate *DB2 Connect Quick Beginnings* manual.

Part 7. Other DB2 for UNIX Installation Methods

© Copyright IBM Corp. 1993, 1999

Chapter 20. Other Methods to Install AIX Server

This section describes the other methods you can use to install DB2 Universal Database products. Some of the advanced methods to install DB2 are detailed in this section. The DB2 Installer program can perform all of the steps described in this section.

We recommend that you use the DB2 Installer program when installing DB2, but if you prefer not to use the DB2 Installer program, you can use any one of the methods described in this chapter. See "Chapter 2. Installing DB2 Clients" on page 13 for information on how to install the DB2 clients.

Installation Steps

To install DB2, perform the following steps:

Step 1. Install DB2.

Step 2. Create or assign groups and user IDs.

Step 3. Create a DB2 instance.

Step 4. Create the Administration Server.

Step 5. Install the license key.

Step 6. Create links for DB2 files.

You can install DB2 using either of the following methods:

Using SMIT

The System Management Interface Tool (SMIT) allows you to install a subset of products and filesets on a single machine. If you want to install only a selected set of DB2 filesets or components, see "Install Products or Filesets Using SMIT" on page 390.

Installing software bundles using SMIT

A software bundle contains a list of filesets that are suited for a particular use. Installation, although easier with software bundles, only occurs on a single machine.

If you prefer to use the **installp** command, you must first mount the DB2 CD-ROM. You also need to mount the CD-ROM if you want to view the *Installation Notes* or print the postscript manuals prior to installation.

© Copyright IBM Corp. 1993, 1999

Step 1. Install DB2

Your product CD-ROM contains several products. Select the products you are licensed to install. *Your Proof of Entitlement* and *License Information* booklet identify the products for which you are licensed.

Go to the installation method that you would like to use to install your product.

- "Install Products or Filesets Using SMIT".
- "Install Software Bundles Using SMIT" on page 391.

Install Products or Filesets Using SMIT

To install a DB2 product or filesets using the System Management Interface Tool (SMIT), perform the following steps:

Step 1. Log in as user with `root` authority.

Step 2. Insert and mount the appropriate CD-ROM.

Step 3. Enter the **smit install_latest** command. The Software Installation and Maintenance menu opens.

Step 4. Specify the `INPUT device/directory` for DB2 and press Enter.

For example, if `/cdrom` is the CD-ROM mount directory, enter the following as the INPUT device:

```
/cdrom/db2
```

If you are installing a DB2 client or a DB2 Software Developer's Kit, enter the following as the INPUT device:

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

Step 5. Press **F4** to display a list of software to install.

Step 6. Press **F7** to mark one or more filesets for installation.

Step 7. Press **Enter** to install DB2.

Step 8. Press **F10** to exit when the command displays **OK**.

After the installation is complete, DB2 will be installed in the `/usr/lpp/db2_06_01` directory on that machine; now you must repeat the above steps on all machines.

Now that you have finished installing DB2, go to "Install the DB2 Product Library (Optional)" on page 392 and "Install the DB2 Product Messages (Optional)" on page 393 to install the optional product documentation and messages. If you do not want to install product documentation or messages, proceed to the next step.

Install Software Bundles Using SMIT

The following procedure shows how to install a DB2 product using the *Software Bundles* feature. A software bundle is a collection of filesets. When you install a software bundle, every fileset in that bundle is installed.

Step 1. Log in as a user with `root` authority.

Step 2. Insert and mount the appropriate CD-ROM.

Step 3. To proceed directly to the automatic bundle processing application, enter the **smit easy_install_bundle** command.

Step 4. Specify the installation device or directory for the installation media by pressing **F4** to display a list.

Step 5. Select the CD-ROM drive you mounted, then press Enter.

Step 6. If you did not install the DB2 Media-Defined software bundle on this system, use the following procedure to install it:

a. In the **Bundle to Install** menu, select the **Media-Defined** option, then press Enter.

b. In the **Install Bundle Contents** menu, press Enter to continue with the installation. Messages may appear to indicate the status of the installation. Press **Enter** to continue.

c. Press **F3** to return to the **Install Bundle Contents** menu.

d. Press **F4** to re-display the list of software bundles.

Step 7. From the displayed menu, select one of the following software bundles:

- DB2V6-UDB-Enterprise
- DB2V6-UDB-Ext-Enterprise
- DB2V6-UDB-Data-Links-Mgr
- DB2V6-Query-Patroller
- DB2V6-Connect-Enterprise
- DB2V6-SDK
- DB2V6-Admin-Client
- DB2V6-Run-Time-Client

Every required and recommended fileset for a product is installed with its bundle.

Step 8. Press Enter to start the installation.

Messages will appear to indicate the status of the installation. Press **Enter** to continue, or **F3** if you want to cancel.

Step 9. Repeat the steps in this section if you want to install software bundles on other machines.

Now that you have finished installing DB2, go to "Install the DB2 Product Library (Optional)" and "Install the DB2 Product Messages (Optional)" on page 393 to install the optional product documentation and messages. If you do not want to install product documentation or messages, proceed to the next step.

Install the DB2 Product Library (Optional)

When you install DB2 using SMIT, the DB2 Product Library (HTML) is not automatically installed.

To install the DB2 Product Library, perform the following steps:

Step 1. Log in as user with `root` authority.

Step 2. Insert and mount the appropriate CD-ROM.

Step 3. Enter the **smit install_latest** command. The Software Installation and Maintenance menu opens.

Step 4. Enter the directory for the **INPUT device/directory for software**. For example, if `/cdrom` represents the CD-ROM mount directory, enter the following as the INPUT device:

```
/cdrom/db2
```

Step 5. Press **F4** to display a list of software to install.

Step 6. Select the fileset for the DB2 Product Library (HTML) that you want to install. A separate fileset exists for every language in which DB2 documentation is translated.

Note: Not every manual is translated into multiple languages. The *DB2 Product Library (HTML) - English* fileset contains a complete set of books in English. When you select any non-English HTML fileset without selecting English, the English HTML fileset is also installed.

The DB2 Product Library (HTML) fileset is listed as follows:

`DB2 Product Library (HTML) - language`

Step 7. Press **Enter** to start the installation.

Step 8. Press **F10** to exit SMIT when the command completes.

Step 9. Run the following command to uncompress and untar the HTML files:

```
/usr/lpp/db2_06_01/doc/db2insthtml locale
```

where *locale* represents the desired locale. For example, `en_US` is the locale for English documentation.

Note: If you are installing a non-English locale, the English locale is installed as well. You should run **db2insthtml** against the English locale first.

Step 10. From your Web browser, open the following file URL to view DB2 online manuals:

```
file:/usr/lpp/db2_06_01/doc/html/index.htm
```

To recover some disk space, you may want to remove the compressed tar files. It is recommended that you do not just remove the compressed tar HTML files; instead, you should remove the fileset for the DB2 Product Library.

After the fileset is removed, only compressed tar files will be removed. You will still be able to view the online DB2 books.

Install the DB2 Product Messages (Optional)

When you install DB2 using SMIT, DB2 product messages for locales, other than en_US (English - ISO 88591), are not automatically installed.

To install DB2 product messages in other locales, perform the following steps:

1. Log in as a user with `root` authority.
2. Insert and mount the appropriate CD-ROM.
3. Enter the **smit install** command.
4. Select the **Install and Update Software** option, then select the **Install and Update from all Available Software** option.
5. Enter the directory for the **INPUT device/directory for software**. For example, if `/cdrom` represents the CD-ROM mount directory, enter the following as the INPUT device:

   ```
   /cdrom/db2
   ```

6. Press **F4** to display a list of software to install.
7. List the software to be installed using the **SOFTWARE to install** option.
8. Select the messages options for the locales you want to install.

Step 2. Create or Assign a Group and User ID

If you want to use an existing user or group ID, you do not need to create new ones at this time. Instead, you can proceed to the next step.

If you are using Network Information Services (NIS), NIS+, or any other tools to manage system configuration files, you must perform this step on the master server. After you have completed the following, ensure that the changes have taken effect on all DB2 servers.

You must perform this step using the *root* user ID.

1. Create a group that will be the primary group of the username for the instance owner. Any user that belongs to the SYSADM group will have system administrator authority for a given instance.

   Next, create a username that will be the instance owner. This username will be the name of the instance. Make this user's primary group the SYSADM group you created. In our examples, the instance username is `db2inst1` and the SYSADM group is `dbadmin1`.

Dedicate the instance owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

2. Use the **mkgroup** command to create groups, and the **mkuser** command to create users. For example, to create a user called `db2inst1` with user ID *1004*, which will use `dbadmin1` with group ID *999* as its primary group and use `/home/db2inst1` as its home directory, type:

   ```
   mkgroup id=999 dbadmin1
   mkuser id=1004 pgrp=dbadmin1 groups=dbadmin1 home=/home/db2inst1 db2inst1
   passwd db2inst1
   ```

3. Create a group and user ID for fenced user defined functions and stored procedures.

   For example:

   ```
   mkgroup id=991 db2fadm1
   mkuser id=1001 pgrp=db2fadm1 groups=db2fadm1 home=/home/db2fenc1 db2fenc1
   passwd db2fenc1
   ```

   Fenced user defined functions (UDFs) and stored procedures will execute under this user and group. The group must be the primary group of the user. The user for fenced UDFs and stored procedures is specified as a parameter of the instance creation script. The group for fenced UDFs and stored procedures is implicitly set to the primary group of this specified user (`db2fenc1`).

For security reasons, we recommend that you do not use the instance name as the FencedID. However, if you are not planning to use fenced user defined functions or stored procedures, you can set the FencedID to the instance name instead of creating another user for the FencedID.

Notes:

1. You have to repeat Steps 1 and 2 above to create a user name and group name for the Administration Server. You must use different user IDs for the Administration Server and a DB2 instance (for example, `db2as` and `db2inst1`). For security reasons, you should not use the primary group of a DB2 instance as the primary group for the user ID for the Administration Server. It is recommended that you create a different group ID (for example, `db2asgrp`).

2. In addition to the rules imposed by the operating system for login names and groups, you must also adhere to the rules described in "Appendix G. Naming Rules" on page 545.

Step 3. Create a DB2 Instance

A DB2 instance is an environment where you store data and run applications. Use the **db2icrt** command to create an instance. You must have *root* authority to enter this command. For more information about database instances, refer *Administration Guide*.

Step 4. Create the Administration Server

The Administration Server provides services to support client tools that automate the configuration of connections to DB2 databases. The Administration Server also supports client tools that administer DB2 from your server system or a remote client using the Command Center. Use the **dasicrt** command to create an Administration Server. For more information about Administration Servers, refer to the *Administration Guide*.

Step 5. Install the License Key

The following steps describe how to upgrade your DB2 license.

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

1. Log in as a user with root authority.
2. The DB2 product license key can be found in the `nodelock` file:
   ```
   /var/ifor/nodelock
   ```
3. Update your DB2 product license with the following command:
   ```
   /usr/lpp/db2_06_01/cfg/db2licm -a filename
   ```

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

   db2udbee.lic
   DB2 Universal Database Enterprise Edition

   db2conee.lic
   DB2 Connect Enterprise Edition

db2conpe.lic
DB2 Connect Personal Edition (Linux only)

db2udbwe.lic
DB2 Universal Database Workgroup Edition (Linux only)

db2udbpe.lic
DB2 Universal Database Personal Edition (Linux only)

db2dlm.lic
DB2 Universal Database Data Links Manager

db2qp.lic
DB2 Universal Database Query Patroller

db2udbeee.lic
DB2 Universal Database Enterprise - Extended Edition

For example, if the CD-ROM is mounted in the /cdrom directory and the name of the license file is db2dlm.lic, the command should be as follows:

```
/usr/lpp/db2_06_01/cfg/db2licm -a /cdrom/db2/license/db2dlm.lic
```

Step 6. Create Links for DB2 Files (Optional)

You can create links for the DB2 files to the /usr/lib directory, and for the include files to the /usr/include directory for a particular version and release level of the product.

You may want to create these links if you are developing or running applications and want to avoid having to specify the full path to the product libraries and include files.

Log in as a user with root authority and invoke the **db2ln** command as follows:

```
/usr/lpp/db2_06_01/cfg/db2ln
```

If you do not issue the **db2ln** command using the **dsh** command, **db2ln** only sets up the links on the machine where the command is issued.

If there are existing links to the /usr/lib and /usr/include directories from previous versions of DB2, they will automatically be removed by entering the **db2ln** command to create links for this version of DB2. If you want to re-establish the links to the libraries or previous versions, then you must execute the **db2rmln** command from Version 5 before you execute the **db2ln** command from the previous versions of DB2.

Links can be established for only one version of DB2 on a given system.

Now that you have installed DB2 Enterprise Edition, see “Chapter 9. Using the Command Line Processor to Configure Server Communications” on page 119 for information on how to set up your server to accept in-bound client connections.

Chapter 21. Other Methods to Install HP-UX Servers

This section describes the other methods you can use to install DB2 Universal Database products. Some of the advanced methods to install DB2 are detailed in this section. The DB2 Installer program can perform all of the steps described in this section.

We recommend that you use the DB2 Installer program when installing DB2, but if you prefer not to use the DB2 Installer program, you can use any one of the methods described in this chapter. See “Chapter 2. Installing DB2 Clients” on page 13 for information on how to install the DB2 clients.

Installation Steps

To install DB2 for HP-UX products, perform the following steps:

1. Install DB2.
2. Update the kernel configuration parameters and reboot the system.
3. Create or assign groups and user IDs.
4. Create a DB2 instance.
5. Create an Administration Server.
6. Install the license key.
7. Create links for DB2 files.

Step 1. Install DB2

Use the **swinstall** program to install DB2 for HP-UX. To install, perform the following steps:

1. Log in as a user with root authority.
2. Insert the DB2 CD-ROM into the CD-ROM drive and mount it. For example:

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

   where `/cdrom` represents the CD-ROM mount directory.
3. Run the **swinstall** program using the following command:

   ```
   swinstall -x autoselect_dependencies=true
   ```

© Copyright IBM Corp. 1993, 1999

This command opens the Software Selection window and the Specify Source window.

4. If necessary, change the **Source Host Name** in the Specify Source window.
5. Enter the following as the value for the **Source Depot Path** field:
   - For HP-UX Version 10.20:

     ```
     /cdrom/db2/hpux10
     ```

   - For HP-UX Version 11:

     ```
     /cdrom/db2/hpux11
     ```

   where `/cdrom` represents the CD-ROM mount directory.
6. To return to the Software Selection window, select the **OK** button.
7. The Software Selection window contains a list of available software to install. Highlight one or more of the following products to install:

   DB2V6CAE
   DB2 Universal Database Run-Time Client for HP-UX

   DB2V6CONN
   DB2 Connect Enterprise Edition for HP-UX

   DB2V6ENTP
   DB2 Universal Database Enterprise Edition for HP-UX

   DB2V6WGRP
   DB2 Universal Database Workgroup Edition for HP-UX

   DB2V6WCC
   DB2 Universal Database Control Center and Help (all locales)

   DB2V6SDK
   DB2 Software Developer's Kit

   DB2V6HTML
   DB2 Universal Database Product Library in HTML (all locales)

   DB2V6MSG
   DB2 Universal Database Product Messages (all locales)

   Notes:

   a. This CD-ROM contains several DB2 products. Select the products you are licensed to install. Your **Proof of Entitlement** and **License Information** booklet identify the products for which you are licensed.
   b. Do not select products DB2V6MSG, DB2V6HTML, and DB2V6WCC. These products are collections of DB2 product messages, HTML documentation in several languages, and the DB2 Control Center and online help. The procedures to install these products are described later in this section.

8. Select **Mark for Install** from the **Actions** menu to choose the product to be installed.
9. Select **OK** when the following message appears:

   ```
   In addition to the software you just marked, other software was
   automatically marked to resolve dependencies.  This message will
   not appear again.
   ```

10. Select **Install (analysis)** from the **Actions** menu to begin product installation and to open the Install Analysis window.
11. Select **OK** in the Install Analysis window when the **Status** field displays a Ready message.
12. Select the **Yes** button in the Confirmation window windows to confirm that you want to install the HP-UX software products.

    View the Install window to read processing data while the software is being installed, until the **Status** field indicates Ready and the Note window opens. The **swinstall** program loads the fileset, and runs the control scripts for the fileset.
13. Select **Exit** from the **File** menu to exit out of **swinstall**.

The HP-UX operating system provides detailed help for **swinstall**. For help, type:

```
man swinstall
```

Install the DB2 Product Library (Optional)

To view DB2 product documents online, HTML versions of these documents are provided for installation. These documents are translated into several languages. However, not every manual in the DB2 product library is translated into multiple languages. The English version of every manual is available in HTML format. When you select a non-English HTML fileset without selecting the English version, the English HTML fileset is also installed.

The DB2V6HTML product includes filesets for all the DB2 product documentation. A separate fileset exists for each language. For example, the fileset name for the DB2 product library in English is DB2V6HTML.en_US. For a complete list of filesets in the DB2V6HTML product, see “Chapter 23. Contents of the DB2 Products for UNIX platforms” on page 419.

To install the DB2 Product Library (HTML) filesets, do the following:

1. Log in as a user with root authority.
2. Insert and mount the DB2 product CD-ROM as described in “Step 1. Install DB2” on page 399.
3. Run the **swinstall** command as follows:

- For HP-UX Version 10:

  ```
  swinstall -s /cdrom/db2/hpux10
  ```

- For HP-UX Version 11:

  ```
  swinstall -s /cdrom/db2/hpux11
  ```

4. From the list of products displayed, highlight the **DB2V6HTML** product and press Enter.

   This will display a list of all the filesets in the **DB2V6HTML** product.

5. Highlight one or more filesets that you want to install and select **Mark for Install** from the Actions menu.

   For example, to install the DB2 product Library in English, highlight the following fileset:

   ```
   en_US     DB2 Product Library (HTML) - English
   ```

6. Follow the remaining installation procedure, starting at Step 9 on page 401 in "Step 1. Install DB2" on page 399.

   This will install the **compressed-tar** files for the DB2 documents in the /opt/IBMdb2/V6.1/doc/<lang>/html directory, where <lang> is the language/locale identifier.

7. Run the following command to uncompress and un-tar the HTML files:

   ```
   /opt/IBMdb2/V6.1/doc/db2insthtml lang
   ```

   where *lang* represents the language/locale identified. For example, en_US for English. For valid language/locale identifiers, see the *Administration Guide*.

   Note: If you are installing a non-English locale, the English locale is installed as well. You should run **db2insthtml** against the English locale first.

8. From your Web browser, open the following URL to view DB2 online manuals:

   ```
   file:/opt/IBMdb2/V6.1/doc/en_US/html/index.htm
   ```

To recover some disk space, you may want to remove the compressed tar files. It is recommended that you do not just remove the compressed tar HTML files; instead, you should remove the fileset for the DB2 Product Library.

After the fileset is removed, only compressed tar files will be removed. You will still be able to view the online DB2 books.

Install DB2 Product Messages (Optional)

While DB2 Product Message documents are translated into several languages, the English language DB2 Product Messages are always installed. To install the DB2 Product Messages in other locales, you have to install one or more filesets in the DB2V6MSG product.

The DB2V6MSG product includes a separate fileset for every locale in which DB2 Product Messages are available. There is no fileset for the English (en_US.iso88591 or en_US.roman8) locale since these messages are installed with the base product. For example, to install the DB2 Product messages in the `fr_FR.iso88591` locale, you need to install the fileset `DB2V6MSG.fr_FR`. For a complete list of filesets in the DB2V6MSG product, see "Chapter 23. Contents of the DB2 Products for UNIX platforms" on page 419.

To install DB2 Product Message filesets, do the following:

1. Log in as a user with root authority.
2. Insert and mount the DB2 product CD-ROM as described in "Installation Steps" on page 399.
3. Run the **swinstall** command as follows:
   - For HP-UX Version 10:
     ```
     swinstall -s /cdrom/db2/hpux10
     ```
   - For HP-UX Version 11:
     ```
     swinstall -s /cdrom/db2/hpux11
     ```

   where `/cdrom` is the CD-ROM mount directory.
4. From the list of products displayed, highlight the **DB2V6MSG** product and press Enter.

   This will display a list of all the filesets in the **DB2V6MSG** product.
5. Highlight one or more filesets that you want to install and select **Mark for Install** from the Actions menu.
6. Follow the remaining installation procedure, starting at Step 9 on page 401 in "Step 1. Install DB2" on page 399.

Install DB2 Control Center and Help (Optional)

The Control Center is the main DB2 graphical tool for administering your database. From the Control Center, you get 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.

The DB2V6WCC product includes a separate fileset for every locale in which the DB2 Control Center is available. For a complete list of filesets in the DB2V6WCC product, see “Chapter 23. Contents of the DB2 Products for UNIX platforms” on page 419.

To install DB2 Control Center filesets, do the following:

1. Log in as a user with root authority.
2. Insert and mount the DB2 product CD-ROM as described in “Installation Steps” on page 399.
3. Run the **swinstall** command as follows:
   - For HP-UX Version 10:

     ```
     swinstall -s /cdrom/db2/hpux10
     ```

   - For HP-UX Version 11:

     ```
     swinstall -s /cdrom/db2/hpux11
     ```

   where `/cdrom` is the CD-ROM mount directory.
4. From the list of products displayed, highlight the **DB2V6WCC** product and press **Enter**.

   This will display a list of all the filesets in the **DB2V6WCC** product.
5. Highlight one or more filesets that you want to install and select **Mark for Install** from the Actions menu.
6. Follow the remaining installation procedure, starting at Step 9 on page 401 in “Step 1. Install DB2” on page 399.

Step 2. Update the Kernel Configuration Parameters

To run DB2 for HP-UX, you may have to update some kernel configuration parameters; the following values are recommended:

Table 30. HP-UX Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Physical Memory		
	64MB - 128MB	**128MB - 256MB**	**256MB+**
maxuprc	256	384	512
maxfiles	256	256	256
nproc	512	768	1024
nflocks	2048	4096	8192
ninode	512	1024	2048
nfile	(4 * ninode)	(4 * ninode)	(4 * ninode)
msgseg	8192	16 384	32 768
msgmnb	65 535 (1)	65 535 (1)	65 535 (1)
msgmax	65 535 (1)	65 535 (1)	65 535 (1)
msgtql	256	512	1024
msgmap	130	258	258
msgmni	128	256	256
msgssz	16	16	16
semmni	128	256	512
semmap	130	258	514
semmns	256	512	1024
semmnu	256	512	1024
shmmax	67 108 864	134 217 728 (2)	268 435 456 (2)
shmseg	16	16	16
shmmni	300	300	300

Notes:

1. Parameters *msgmnb* and *msgmax* must be set to at least 65535.
2. Parameter *shmmax* should be set to 134217728 or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set *shmmax* to 184 968 806 (196*0.9*1024*1024).
3. To maintain the interdependency among kernel parameters, change parameters in the same sequence in which they appear in the preceding table.

Step 3. Create or Assign Groups and User IDs

If you want to use an existing user or group ID, you do not need to create new ones at this time. Instead, you can proceed to the next step.

You must be a user with root authority to perform the following:

1. Create a group that will be the primary group of the username for the instance owner. Any user that belongs to the SYSADM group will have system administrator authority for a given instance.

   Next, create a username that will be the instance owner. This username will be the name of the instance. Make this user's primary group the SYSADM group you created. In our examples, the instance username is `db2inst1` and the SYSADM group is `dbadmin1`.

Dedicate the instance owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

2. Use **SAM** or **groupadd** to create groups, and **SAM** or **useradd** to create users. For example:

   ```
   groupadd dbadmin1
   useradd -g dbadmin1 -d /home/instl -m db2inst1
   passwd db2inst1
   ```

3. Create a group and user for fenced user defined functions and stored procedures.

   For example:

   ```
   groupadd db2fadm1
   useradd -g db2fadm1 -d /home/db2fenc1 -m db2fenc1
   passwd db2fenc1
   ```

   Fenced user defined functions (UDFs) and stored procedures will execute under this user and group. The group must be the primary group of the user. The user for fenced UDFs and stored procedures is specified as a parameter of the instance creation script. The group for fenced UDFs and stored procedures is implicitly set to the primary group of this specified user (`db2fenc1`).

For security reasons, we recommend that you do not use the instance name as the FencedID. However, if you are not planning to use fenced user defined functions or stored procedures, you can set the FencedID to the instance name instead of creating another user for the FencedID.

Notes:

1. You have to repeat Steps 1 and 2 above to create a user name and group name for the Administration Server. You must use different user IDs for the Administration Server and a DB2 instance (for example, `db2as` and

db2inst1). For security reasons, you should not use the primary group of a DB2 instance as the primary group for the user ID for the Administration Server. It is recommended that you create a different group ID (for example, db2asgrp).

2. In addition to the rules imposed by the operating system for login names and groups, you must also adhere to the rules described in "Appendix G. Naming Rules" on page 545.

Step 4. Create a DB2 Instance

A DB2 instance is an environment where you store data and run applications. Use the **db2icrt** command to create an instance. You must have *root* authority to enter this command. For more information about database instances, refer *Administration Guide*.

Step 5. Create an Administration Server

The Administration Server provides services to support client tools that automate the configuration of connections to DB2 databases. The Administration Server also supports client tools that administer DB2 from your server system or a remote client using the Command Center. Use the **dasicrt** command to create an Administration Server. For more information about Administration Servers, refer to the *Administration Guide*.

Step 6. Install the License Key

The following steps describe how to upgrade the license:

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

1. Log in as a user with root authority.
2. The DB2 product license key can be found in the nodelock file:

   ```
   /var/opt/ifor/nodelock
   ```

3. Update your DB2 product license with the following command:

   ```
   /opt/IBMdb2/V6.1/cfg/db2licm -a filename
   ```

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

   db2udbee.lic
   DB2 Universal Database Enterprise Edition

db2conee.lic
DB2 Connect Enterprise Edition

For example, if the CD-ROM is mounted in the /cdrom directory and the name of the license file is db2conee.lic for the DB2 Universal Database Extended Enterprise Edition for UNIX. The command should be as follows:

```
/opt/IBMdb2/V6.1/cfg/db2licm/cdrom/db2/license/db2conee.lic
```

Step 7. Create Links for DB2 Files (Optional)

You can create links for the DB2 files to the /usr/lib directory, and for the include files to the /usr/include directory for a particular version and release level of the product.

You may want to create these links if you are developing or running applications and want to avoid having to specify the full path to the product libraries and include files.

Log in as root and invoke the **db2ln** command as follows:

```
/opt/IBMdb2/V6.1/db2ln
```

If there are existing links to the /usr/lib and /usr/include directories from previous versions of DB2, they will automatically be removed by entering the **db2ln** command to create links for this version of DB2. If you want to re-establish the links to the libraries or previous versions, then you must execute the **db2rmln** command from Version 5 before you execute the **db2ln** command from the previous versions of DB2.

Links can be established for only one version of DB2 on a given system.

Now that you have installed DB2 Enterprise Edition, see "Chapter 9. Using the Command Line Processor to Configure Server Communications" on page 119 for information on how to set up your server to accept in-bound client connections.

Chapter 22. Other Methods to Install DB2 for Solaris

This section describes the other methods you can use to install DB2 Universal Database products. Some of the advanced methods to install DB2 are detailed in this section. The DB2 Installer program can perform all of the steps described in this section.

We recommend that you use the DB2 Installer program when installing DB2, but if you prefer not to use the DB2 Installer program, you can use any one of the methods described in this chapter. See “Chapter 2. Installing DB2 Clients” on page 13 for information on how to install the DB2 clients.

Installation Steps

To install DB2 for Solaris, perform the following steps:

1. Install DB2.
2. Update the kernel configuration parameters.
3. Create or assign groups and user IDs.
4. Create a DB2 instance.
5. Create an Administration Server.
6. Install the license key.
7. Create links for DB2 files.

Step 1. Install DB2

Use the **db2_install** command, which is available on the DB2 CD-ROM, to install DB2. If your system does not have a local CD-ROM drive, you can mount the CD-ROM on a different system and NFS mount on the current system.

To install DB2 from the DB2 CD-ROM, perform the following steps:

1. Log in as a user with root authority.
2. Insert the DB2 CD-ROM into the CD-ROM drive.
3. If the Volume Manager is not running on your system, enter the following commands to mount the CD-ROM:

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

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

© Copyright IBM Corp. 1993, 1999

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. Run the **db2_install** command as follows:

   ```
   /cdrom/unnamed_cdrom/db2_install
   ```

   The **db2_install** command prompts for one or more of the following to be installed, and for the base directory where the product files are to be installed. The products are listed by keyword and product description.

   DB2.EENT
   DB2 Universal Database Enterprise - Extended Edition for Solaris

   DB2.ENTP
   DB2 Universal Database Enterprise Edition for Solaris

   DB2.CENT
   DB2 Connect Enterprise Edition for Solaris

   DB2.CAE
   DB2 Universal Database Administration Client

   DB2.RCAE
   DB2 Universal Database Run-Time Client

   DB2.SDK
   DB2 Software Developer's Kit

   DB2.DQP
   DB2 Universal Database Query Patroller

   NETQ IBM NetQuestion HTML Search System

   When installing DB2 Connect for Solaris, the **db2_install** command prompts you for the product to be installed and for the base directory where the product files are to be installed. Select DB2.CENT to install DB2 Connect Enterprise Edition.

   The command displays the following prompt: `Specify one or more of the keywords separated by spaces.`

5. Type the keyword of the products to be installed when prompted.
6. Type the name of the base directory when prompted. The default base directory is `/opt`.

If the default base directory is used, all files will be installed in the /opt/IBMdb2/V6.1 directory.

7. Enter Yes to start the DB2 product installation.

Installing the DB2 Product Library (Optional)

To view DB2 product documents online, HTML versions of these documents are provided for installation. These documents are translated into several languages. However, not every manual in the DB2 product library is translated into multiple languages. The English version of every manual is available in HTML format. When you select a non-English HTML fileset without selecting the English version, the English HTML fileset is also installed.

A separate package exists for every language in which DB2 documents are translated. Packages for DB2 documentation are given names such as db2ht%L50, where %L represents two letters for the language and locale identifier.

For example, the package name for the DB2 Product Library in English is db2hten50. For a complete list of packages for the DB2 Product Library, see "Chapter 23. Contents of the DB2 Products for UNIX platforms" on page 419.

To install the DB2 Product Library (HTML) package, do the following:

1. Log in as a user with root authority and mount the DB2 product CD-ROM as described in "Step 1. Install DB2" on page 409.
2. Run the **pkgadd** command as follows:

   ```
   pkgadd -d cdrom-dir/db2 pkgname
   ```

   where *cdrom-dir* is the CD-ROM mount directory and *pkgname* is the package name. For example, to install DB2 documentation in English, you can enter the following command:

   ```
   pkgadd -d /cdrom/unnamed_cdrom/db2 db2hten61
   ```

   This command installs the compressed-tar files for the DB2 documents in the /opt/IBMdb2/V6.1/doc/*lang*/html directory, where *lang* is the language and locale identifier.
3. Run the following command to uncompress and untar the HTML files:

   ```
   /opt/IBMdb2/V6.1/doc/db2insthtml lang
   ```

   where *lang* is the language and locale identifier.

For example, *lang*=en_US for English.

If you are installing a non-English locale, the English locale is installed as well. You should perform this command in the English locale first.

4. From your Web browser, open the following URL to view the DB2 manuals:

   ```
   file:/opt/IBMdb2/V6.1/doc/en_US/html/index.htm
   ```

To recover some disk space, you may want to remove the compressed tar files. It is recommended that you do not just remove the compressed tar HTML files; instead, you should remove the fileset for the DB2 Product Library.

After the fileset is removed, only compressed tar files will be removed. You will still be able to view the online DB2 books.

Installing the DB2 Product Messages (Optional)

A separate package exists for every locale in which DB2 Product Messages are translated. Packages for DB2 Product Messages are given names such as db2ms%L50, where %L represents two letters for the language name. For example, the package name for the DB2 Product Messages in the de (German) locale is db2msde50. For a complete list of DB2 Product Messages, see "Chapter 23. Contents of the DB2 Products for UNIX platforms" on page 419.

To install DB2 Product Message packages, do the following:

1. Log in as a user with root authority and mount the DB2 CD-ROM as described in "Step 1. Install DB2" on page 409.
2. Run the **pkgadd** command as follows:

   ```
   pkgadd -d cdrom-dir/db2 pkgname
   ```

   where *cdrom-dir* is the CD-ROM mount directory and *pkgname* is the package name.

   For example, to install the DB2 Product Messages in the de (German) locale, enter the following command:

   ```
   pkgadd -d /cdrom/unnamed_cdrom/db2 db2msde61
   ```

Step 2. Update the Kernel Configuration Parameters

To run DB2 for Solaris, you may have to update some kernel configuration parameters. The following minimum values are recommended:

Table 31. Solaris Kernel Configuration Parameters (Recommended Values)

Kernel Parameter	Physical Memory			
	64MB - 128MB	128MB - 256MB	256MB - 512MB	512MB+
msgsys:msginfo_msgmax	65 535(1)	65535(1)	65 535(1)	65 535(1)
msgsys:msginfo_msgmnb	65 535(1)	65 535(1)	65 535(1)	65 535(1)
msgsys:msginfo_msgmap	130	258	258	258
msgsys:msginfo_msgmni	128	256	256	256
msgsys:msginfo_msgssz	16	16	16	16
msgsys:msginfo_msgtql	256	512	1024	1024
msgsys:msginfo_msgseg	8 192	16 384	32 768	32 768
shmsys:shminfo_shmmax	67 108 864	134 217 728(2)	268 435 456(2)	536 870 912(2)
shmsys:shminfo_shmseg	16	16	16	16
shmsys:shminfo_shmmni	300	300	300	300
semsys:seminfo_semmni	128	256	512	1024
semsys:seminfo_semmap	130	258	514	1026
semsys:seminfo_semmns	256	512	1024	2048
semsys:seminfo_semmnu	256	512	1024	2048

Notes:

1. The *msgsys:msginfo_msgmnb* and *msgsys:msginfo_msgmax* parameters must be set to 65535 or larger.
2. The *shmsys:shminfo_shmmax* parameters should be set to the suggested value in the above table, or 90% of the physical memory (in bytes), whichever is higher. For example, if you have 196 MB of physical memory in your system, set the *shmsys:shminfo_shmmax* parameter to 184968806 (196*0.9*1024*1024).

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

```
set parameter_name = value
```

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

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

Sample files for updating the kernel configuration parameters are provided in the `/opt/IBMdb2/V6.1/cfg` directory. The names for these files are as follows:

kernel.param.64MB
for systems with 64MB–124MB of physical memory

kernel.param.128MB
for systems with 128MB–256MB of physical memory

kernel.param.256MB
for systems with 256MB–512MB of physical memory

kernel.param.512MB
for systems with 512MB–1GB of physical memory

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

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

Step 3. Create or Assign Groups and User IDs

If you want to use an existing user or group ID, you do not need to create new ones at this time. Instead, you can proceed to the next step.

If you are using Network Information Services (NIS), NIS+, or any other tools to manage system configuration files, you must perform this step on the master server. After you have completed the following, ensure that the changes have taken effect on all DB2 servers.

You must perform the following as a user with `root` authority.

1. Create a group that will be the primary group of the username for the instance owner. Any user that belongs to the SYSADM group will have system administrator authority for a given instance.

   Next, create a username that will be the instance owner. This username will be the name of the instance. Make this user's primary group the SYSADM group you created. In our examples, the instance username is `db2inst1` and the SYSADM group is `dbadmin1`.

Dedicate the instance owner user ID to that instance's use only. This allows for easier error recovery if a system error occurs.

2. Use **admintool** or **groupadd** to create groups, and **admintool** or **useradd** to create users. For example:

```
groupadd -g 999 dbadmin1
useradd -g dbadmin1 -u 1004 -d /export/home/db2inst1 -m db2inst1
passwd db2inst1
```

3. Similarly, create a group and user for fenced user defined functions and stored procedures.

   For example:

   ```
   groupadd -g 991 db2fadm1
   useradd -g db2fadm1 -u 1001 -d /export/home/db2fenc1 -m db2fenc1
   passwd db2fenc1
   ```

   Fenced user defined functions (UDFs) and stored procedures will execute under this user and group. The group must be the primary group of the user. The user for fenced UDFs and stored procedures is specified as a parameter of the instance creation script. The group for fenced UDFs and stored procedures is implicitly set to the primary group of this specified user (`db2fenc1`).

For security reasons, we recommend that you do not use the instance name as the FencedID. However, if you are not planning to use fenced user defined functions or stored procedures, you can set the FencedID to the instance name instead of creating another user for the FencedID.

Notes:

1. You have to repeat Steps 1 and 2 above to create a user name and group name for the Administration Server. You must use different user IDs for the Administration Server and a DB2 instance (for example, `db2as` and `db2inst1`). For security reasons, you should not use the primary group of a DB2 instance as the primary group for the user ID for the Administration Server. It is recommended that you create a different group ID (for example, `db2asgrp`).
2. In addition to the rules imposed by the operating system for login names and groups, you must also adhere to the rules described in "Appendix G. Naming Rules" on page 545.

Step 4. Create a DB2 Instance

A DB2 instance is an environment where you store data and run applications. Use the **db2icrt** command to create an instance. You must have *root* authority to enter this command. For more information about database instances, refer *Administration Guide*.

Step 5. Create an Administration Server

The Administration Server provides services to support client tools that automate the configuration of connections to DB2 databases. The Administration Server also supports client tools that administer DB2 from your server system or a remote client using the Command Center. Use the

dasicrt command to create an Administration Server. For more information about Administration Servers, refer to the *Administration Guide*.

Step 6. Install the License Key

The following steps describe how to upgrade the license:

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

1. Log in as a user with root authority.
2. The DB2 product license key can be found in the `nodelock` file:

   ```
   /var/netls/nodelock
   ```

3. Update your DB2 product license with the following command:

   ```
   /opt/IBMdb2/V6.1/cfg/db2licm -a filename
   ```

   where `filename` is the full pathname and filename for the license file that corresponds to the product you have purchased. The name of the license file for these products is:

   db2udbee.lic
   DB2 Universal Database Enterprise Edition

   db2conee.lic
   DB2 Connect Enterprise Edition

   db2qp.lic
   DB2 Universal Database Query Patroller

   db2udbeee.lic
   DB2 Universal Database Enterprise - Extended Edition

   For example, if the CD-ROM is mounted in the `/cdrom` directory and the name of the license file is `db2udbee.lic`, the command should be as follows:

   ```
   /opt/IBMdb2/V6.1/cfg/db2licm -a /cdrom_unnamed/db2/license/db2udbee.lic
   ```

Step 7. Create Links for DB2 Files (Optional)

You can create links for the DB2 files to the `/usr/lib` directory, and for the include files to the `/usr/include` directory for a particular version and release level of the product.

You may want to create these links if you are developing or running applications and want to avoid having to specify the full path to the product libraries and include files.

Log in as a user with root authority and invoke the **db2ln** command as follows:

```
/opt/IBMdb2/V6.1/cfg/db2ln
```

If there are existing links to the /usr/lib and /usr/include directories from previous versions of DB2, they will automatically be removed by entering the **db2ln** command to create links for this version of DB2. If you want to re-establish the links to the libraries or previous versions, then you must execute the **db2rmln** command from Version 5 before you execute the **db2ln** command from the previous versions of DB2.

Links can be established for only one version of DB2 on a given system.

Now that you have installed DB2 Enterprise Edition, see “Chapter 9. Using the Command Line Processor to Configure Server Communications” on page 119 for information on how to set up your server to accept in-bound client connections.

Chapter 23. Contents of the DB2 Products for UNIX platforms

This section lists the contents of various DB2 Universal Database products available for UNIX platforms.

Packaging

The following DB2 Universal Database components, packages, or filesets are available for installation:

Table 32. DB2 Components, Packages or Filesets

Description	Fileset Name		
	DB2 for AIX	DB2 for HP-UX	DB2 for Solaris and Linux [3]
DB2 Client	db2_06_01.client	DB2V6CAE.client	db2cliv61
Java Support (JDBC)	db2_06_01.jdbc	DB2V6CAE.jdbc	db2jdbc61
DB2 Control Center	db2_06_01.wcc	DB2V6WCC.wcc	db2wcc61
Administration Server	db2_06_01.das	DB2V6WGRP.das	db2das61
DB2 Control Server	db2_06_01.ctsr	n/a	n/a
DB2 Run-time Environment	db2_06_01.db2.rte	DB2V6WGRP.db2rte	db2rte61
DB2 Sample Database Source	db2_06_01.db2.samples	DB2V6WGRP.dbsmpl	db2smpl61
DB2 Engine	db2_06_01.db2.engn	DB2V6WGRP.db2engn	db2engn61
DB2 Replication	db2_06_01.repl	DB2V6WGRP.repl	db2repl61
DB2 Connect	db2_06_01.conn	DB2V6CONN.conn	db2conn61
DB2 Communication Support - TCP/IP	db2_06_01.cs.rte	DB2V6WGRP.csrte	db2crte61
DB2 Communication Support - SNA	db2_06_01.cs.sna	DB2V6WGRP.cssna	db2csna61 [4]
DB2 Communication Support - DRDA Application Server	db2_06_01.cs.drda	DB2V6WGRP.csdrda	db2cdrd61 [4]
DB2 Communication Support - IPX	db2_06_01.cs.ipx	DB2V6WGRP.csipx	db2cipx61 [4]
DB2 Query Patroller Agent	db2_06_01.dqp.agt	n/a	db2dqpa61 [4]
DB2 Query Patroller Server	db2_06_01.dqp.srv	n/a	db2dqps61 [4]
DB2 Enterprise - Extended Edition Extentions	db2_06_01.pext	n/a	db2pext61 [4]

© Copyright IBM Corp. 1993, 1999

Table 32. DB2 Components, Packages or Filesets (continued)

Description	Fileset Name		
	DB2 for AIX	**DB2 for HP-UX**	**DB2 for Solaris and Linux [3]**
License Support of DB2 Enterprise - Extended Edition	db2_06_01.xlic	n/a	db2xlic61
License Support of DB2 Enterprise Edition	db2_06_01.elic	DB2V6ENTP.elic	db2elic61
License Support of DB2 Connect Enterprise Edition	db2_06_01.clic	DB2V6CONNclic	db2clic61
License Support of DB2 Workgroup Edition	n/a	n/a	db2wlic61 [5]
License Support of DB2 Personal Edition	n/a	n/a	db2pelic61 [5]
License Support of DB2 Connect Personal Edition	n/a	n/a	db2cplic61 [5]
Application Development Tools	db2_06_01.adt.rte	DB2V6SDK1.adtrte	db2adt61
DB2 ADT Sample Programs	db2_06_01.adt.samples	DB2V6SDK.adtsamp	db2adts61
Code Page Conversion Tables - Uni Code Support	db2_06_01.cnvucs	DB2V6CAE.convucs	db2cucs61
Code Page Conversion Tables - Japanese	db2_06_01.conv.jp	DB2V6WGRP.convjp	db2cnvj61
Code Page Conversion Tables - Korean	db2_06_01.conv.kr	DB2V6WGRP.convkr	db2cnvk61
Code Page Conversion Tables - Simplified Chinese	db2_06_01.conv.sch	DB2V6WGRP.convsch	db2cnvc61
Code Page Conversion Tables - Traditional Chinese	db2_06_01.conv.tch	DB2V6WGRP.convtch	db2cnvt61
DB2 Product Messages - %L [1, 2]	db2_06_01.msg.%L	DB2V6MSG.%L	db2ms%L61
DB2 Product Document (HTML) - %L [1]	db2_06_01.html.%L	DB2V6HTML.%L	db2ht%L61

Table 32. DB2 Components, Packages or Filesets (continued)

Description	Fileset Name		
	DB2 for AIX	DB2 for HP-UX	DB2 for Solaris and Linux [3]

Notes:

1. %L in the fileset name represents the locale name. There is a separate fileset for each locale. While DB2 Product Messages and Documentation are translated in several locales, not every message catalog or book is translated in every locale. For a complete list of DB2-supported locales, refer to the *Administration Guide.*
2. English messages are always installed.
3. All Linux component, package, and fileset names are the same as Solaris but have the extension `-6.1.0-0.i386.rpm`. For example, the Linux DB2 Client component name is `db2cliv61-6.1.0-0.i386.rpm`.
4. Not available on DB2 for Linux.
5. Only available on DB2 for Linux.

Products and Selectable Components

Table 33 lists the DB2 Universal Database products and selectable components that you can install.

Table 33. Components for DB2 Products

Product / Component Description	DB2 client	DB2 Universal Database Workgroup Edition	DB2 Universal Database Enterprise Edition	DB2 Enterprise - Extended Edition	DB2 Connect
DB2 client	✓✓	✓✓	✓✓	✓✓	✓✓
Java Support (JDBC)	✓	✓	✓	✓	✓
DB2 Control Center support	..	✓	✓	✓	✓
Administration Server	n/a	✓✓	✓✓	✓✓	✓✓
DB2 Run-time Environment	n/a	✓✓	✓✓	✓✓	✓✓
DB2 Sample Database Source	n/a	✓✓	✓✓	✓✓	n/a
DB2 Engine	n/a	✓✓	✓✓	✓✓	n/a
DB2 Connect	n/a	n/a	✓✓	✓✓	✓✓
DB2 Enterprise - Extended Edition Extensions	n/a	n/a	n/a	✓	n/a
Communication Support for TCP/IP	n/a	✓✓	✓✓	✓✓	✓✓

Table 33. Components for DB2 Products (continued)

Product / Component Description	DB2 client	DB2 Universal Database Workgroup Edition	DB2 Universal Database Enterprise Edition	DB2 Enterprise - Extended Edition	DB2 Connect
Communication Support for IPX/SPX	n/a	✓✓	✓✓	✓✓	✓✓
Communication Support for SNA	n/a	✓✓	✓✓	✓✓	✓✓
Communication Support for DRDA Application Server	n/a	✓✓	✓✓	✓✓	✓✓
Code Page Conversion Tables - Japanese	n/a	✓	✓	✓	✓
Code Page Conversion Tables - Korean	n/a	✓	✓	✓	✓
Code Page Conversion Tables - Simplified Chinese	n/a	✓	✓	✓	✓
Code Page Conversion Tables - Traditional Chinese	n/a	✓	✓	✓	✓
DB2 Replication	n/a	✓	✓	✓	✓
DB2 Product Messages (non-English)[1]	n/a	o	o	o	o
DB2 Product Library (HTML)[1]	n/a	o	o	o	o

✓✓ This is a required component that must be installed.

✓ This is a recommended component that is installed by default. You can choose not to install this component.

o This is an optional component that is not installed by default. If you want to install it, you must select it.

.. This component is available with the DB2 Administration Client only.

n/a This component is not available for installation.

Notes:

1. There is a separate component of the DB2 Product Messages and the DB2 Product Library for each locale.
2. The DB2 Software Developer's Kit is available only with the SDK Pack.

Part 8. Distributed Installation

© Copyright IBM Corp. 1993, 1999

Chapter 24. 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

You can install DB2 products as follows:

- Interactively, using a DB2 product CD-ROM or a system management tool, such as Microsoft Systems Management Server (SMS) on Windows NT.
- Remotely, using a shared CD-ROM drive or shared network hard drive with response file, or a response file and a system management tool.

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 normal 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.

© Copyright IBM Corp. 1993, 1999

Available Sample Response Files

The DB2 CD-ROM includes a ready-to-use sample response file with default entries. The sample response files are located in *x*:\db2\winnt95\common directory, where x: represents the CD-ROM drive.

You can use the following sample response files to install DB2 products on supported workstations:

db2admcl.rsp DB2 Administration Client

db2conee.rsp DB2 Connect Enterprise Edition

db2conpe.rsp DB2 Connect Personal Edition

db2dlm.rsp DB2 Data Links Manager

db2udbpe.rsp DB2 Universal Database Personal Edition

db2rtcl.rsp DB2 Run-Time Client

db2udbse.rsp DB2 Universal Database Satellite

db2sdk.rsp Software Developer's Kit

db2udbwe.rsp DB2 Universal Database Workgroup Edition

db2udbeee.rsp

For Windows NT, 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.

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.

db2udbeeenn.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 only).

db2udbeeesp.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 only).

db2qpa.rsp Query Patroller Agent

db2qps.rsp Query Patroller Server

Important 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.

FILE Specifies the destination directory for a DB2 product.

REBOOT
Specifies whether to restart the system when the installation has completed.

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).

PROD Specifies the product you want installed. (Windows 32-bit operating systems only). 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
- DB2_QP_SERVER for DB2 Query Patroller Server
- RUNTIME_CLIENT for DB2 Run-Time Client
- SDK for DB2 Software Developer's Kit
- UDB_EEE for DB2 Enterprise - Extended Edition
- UDB_ENTERPRISE for DB2 Enterprise Editon
- UDB_PERSONAL for DB2 Personal Edition
- UDB_SATELLITE for DB2 Satellite Edition
- UDB_WORKGROUP for DB2 Workgroup 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.

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.

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.

DB2SYSTEM
Specifies a name for the system which is unique within a network. This parameter must be specified.

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.

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.

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 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.

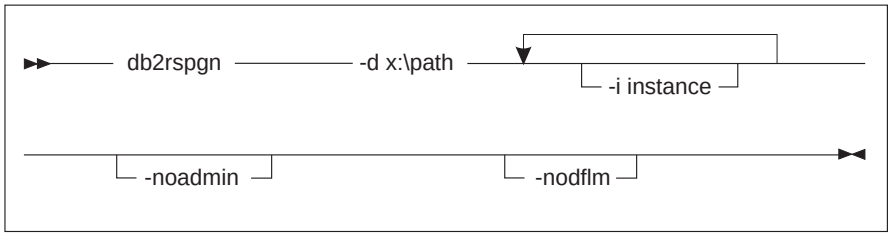
For example, you could install and configure a DB2 Run-Time client to connect to various databases across your network. Once this DB2 client is installed and configured to access all the database that your users to have access to, you can run the response file generator to create a response file and 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 supporting files. This parameter is required.

-i A list of instances for which you want to create a profile. The administration instance (DB2DAS00, reserved for use by the

Administration Server) 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, reserved for use by the Administration Server). The administration instance will then be created with the standard defaults. The default is to save the administration instance. This parameter is optional.

-nodlfm
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 inst2 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 25. Distributed DB2 Installation on Windows 32-Bit Operating Systems" on page 431
- "Chapter 26. Distributed DB2 Installation on UNIX Operating Systems" on page 443
- "Chapter 27. Distributed DB2 Installation on OS/2 Operating Systems" on page 445

Chapter 25. 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 DB2 Software Developer's Kit, see "Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor" on page 43

Step 1. 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 code server, perform the following steps:

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

Step 2. Create a directory on the code server by entering the following command:

```
md c:\db2prods
```

Step 3. Enter the **cpysetup.bat** command to copy the DB2 installation files to your code server. This command is located in the *x:*\db2\winnt95\common\ directory, where *x:* represents your CD-ROM drive.

The command syntax is as follows:

```
cpysetup.bat directory language
```

where:

© Copyright IBM Corp. 1993, 1999

- *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 36 on page 542 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:

```
cpysetup.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`.

If you are using Windows 9x, you do not need to specify the type of access when you set up a share. By default, everyone is granted read access. You can skip Step 6 and go to "Create a Response File" on page 433.

Step 7. To specify *read access* for everyone:

a. Click on 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 on the **Type of Access** drop down box and select the **Read** option.

d. Click on **OK**. You are returned to the properties window of the directory for which you want to set up shared access.

e. Click on **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 429.

If you have already generated a response file using the response file generator, go to "Step 3. Run Setup with the Response File from the Client Workstation" on page 434.

The DB2 CD-ROM includes a ready-to-use sample response file with default entries. The sample response files are located in *x*:\db2\winnt95\common directory, where x: represents the CD-ROM drive.

Response files are available for each DB2 product, see "Available Sample Response Files" on page 426 for more information.

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 "Step 3. Run Setup with the Response File from the Client Workstation" on page 434.

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 Keywords for OS/2 and Windows 32-Bit Operating Systems" on page 427.

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 must store the renamed response file on another drive.

For example, the following response file would install a DB2 Administration Client on the c:\sqllib directory, with the reboot and the catalog no authorization options enabled:

```
.
.
.
FILE                 = c:\sqllib
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.

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.

Step 3. 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 436.

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 the user account that you created to install your DB2 product in "Before You Begin" on page 431.

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.

Note: You may be prompted to enter a user name and password depending on how your domain security is set up.

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:

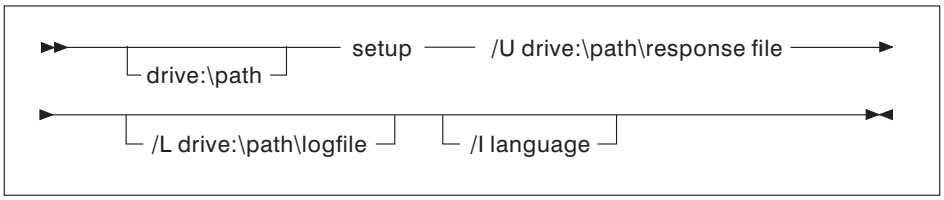
```
net use x: \\codesrv\db2nt
```

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 an 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 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, you must 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.

Go to "Configuring Client Settings" on page 440.

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 wish to roll out an installation on a large number of clients all based on the same setup.

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 for Windows NT Administrator's Guide* for information on how to:

- Set up SMS (including setting up primary and secondary sites).
- Add clients to the SMS system.
- Set up inventory collection for clients.

Step 1. 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 on **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 on **OK**. The **Open SMS Window** window opens.

Step 4. Select the **Packages** window type and click on **OK**. The **Packages** window opens

Step 5. Select **File->New** from the menu bar. The **Package Properties** window opens.

Step 6. Click on 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 on **OK**.

Step 2. 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, you must ensure that all the instance profiles are located in the same drive and directory as the response file that you specify.

Step 5. Click on **OK**.

Step 6. Click on the **Close** push button.

Step 7. Click on **OK** to close the opened windows. The Packages window shows the name of the new SMS package.

Step 3. 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 for Windows NT Administrator's Guide*.

Step 7. When the job specifications are complete, click on **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 on **OK** to close the **Job Schedule** window.

Step 11. Click on **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 on **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 on **Start** and select **Programs->SMS Client->Package Command Manager**. The **Package Command Manager** window opens.

Step 6. Click on 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 install your DB2 product, you can configure your product to access remote database 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 catalogues 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 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 or DB2 Connect Personal Edition 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 mvsdb at node tcptst1 authentication dcs
db2 catalog dcs database mvsdb as mvs_locator
db2 catalog system odbc data source mvsdb
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 on **Start** and select **Programs->Systems Management Server->SMS Administrator** . The **Open SMS Window** window opens.

Step 2. Select the **Packages** window type and click on **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 on the check box for the platforms that should be supported, under the **Supported Platforms** section.

Step 12. Click on **OK**.

Step 13. Click on **Close**.

Step 14. Click on **OK**.

Distribute this package in the same way as you did in “Step 3. Distribute the DB2 Installation Package from the SMS Server” on page 438.

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 an Instance 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 26. 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.

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. Performance will be particularly affected when there are several client installs occurring at the same time. If you are planning on installing multiple clients, you should set up a mounted file system on a code server to improve performance.

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 userids/groups before running the response file installation.

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 426.

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.

© Copyright IBM Corp. 1993, 1999

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 Keywords for OS/2 and Windows 32-Bit Operating Systems" on page 427.

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.

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 default location of the log file is: /tmp/db2setup.log

Chapter 27. 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; save 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. “Step 1. Make DB2 Files Available for Installation”

Step 2. “Step 2. Create a Response File for Distributed Installation” on page 446

Step 3. “Step 3. Run the CMD File from the Remote Workstation” on page 447

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.

Step 1. Make DB2 Files Available for Installation

To make the DB2 install files accessible to the machine where DB2will 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.

© Copyright IBM Corp. 1993, 1999

- 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 36 on page 542 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.

Step 2. 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 "Step 3. Run the CMD File from the Remote Workstation" on page 447.

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 426. 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:\sqllib
   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 Keywords for OS/2 and Windows 32-Bit Operating Systems" on page 427.

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.

Step 3. Run the CMD File from the Remote Workstation

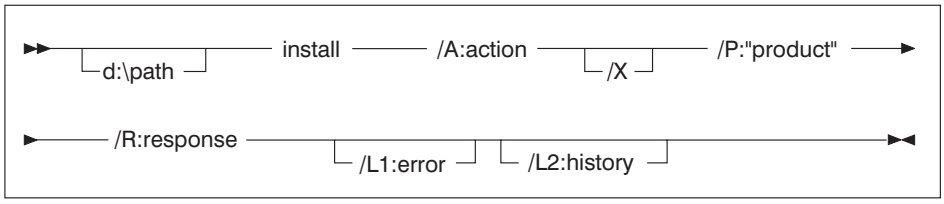
A command (CMD) file contains the commands which will start the installation program. You must edit this file before running the installation.

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:
     - db2admcl.cmd for a DB2 Administration client.
     - db2conee.cmd for DB2 Connect Enterprise Edition.
     - db2conpe.cmd for DB2 Connect Personal Edition.
     - db2rtcl.cmd for a DB2 Run-Time client.
     - db2sdk.cmd for DB2 Software Developer's Kit.
     - 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.
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\db2rtcl.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. 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 31.

Part 9. Configuring Distributed Requests and Access to Heterogeneous Data Sources

© Copyright IBM Corp. 1993, 1999

Chapter 28. 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—is achieved by assigning nicknames to all tables and views that will be involved in distributed request queries. A 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 database functionality is available for DB2 Enterprise Edition and DB2 Extended Enterprise Edition only.

For more information about federated database system concepts, see the *Administration Guide, Design and Implementation*.

Figure 3 on page 454 illustrates a typical federated system environment.

© Copyright IBM Corp. 1993, 1999

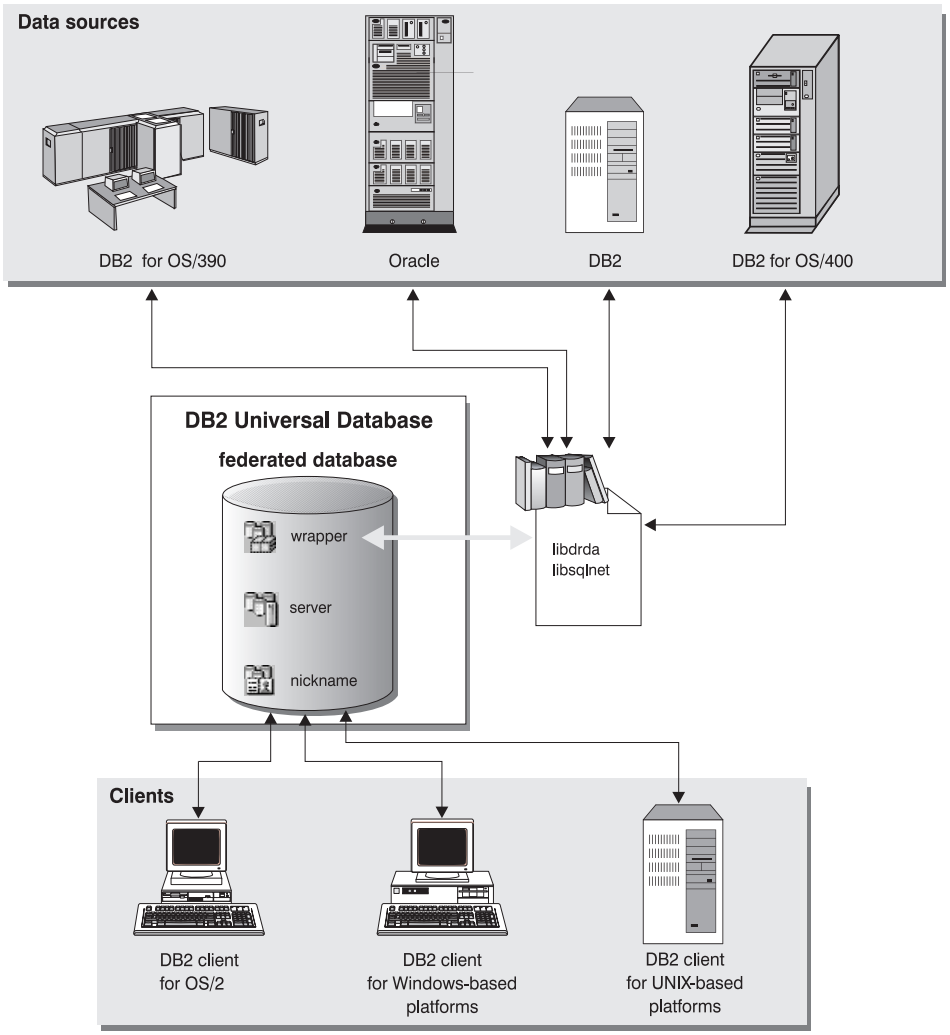


Figure 3. DB2 Federated System

Supported Data Sources

DB2 family and Oracle data sources can be included in distributed requests using nicknames. Table 34 lists the different supported versions and any required maintenance, and corresponding access methods.

Table 34. Data Sources and Their Access Methods

Data Source	Access Method
• 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 • DataJoiner	DRDA or DB2RA
Oracle V7.0.16 or later	Oracle SQL*Net or Net8

Chapter 29. Setting Up a Federated System to Access DB2 Family Data Sources

This chapter explains how to configure your federated server to access data stored on DB2 family data sources and contains the following sections:

- "Enabling Federated Database Functionality"
- "Adding DB2 Family Data Sources to a Federated System"
- "Verifying Connections to DB2 Family Data Sources" on page 460

The instructions in this section apply to Windows NT and supported UNIX platforms; platform-specific differences are noted where they occur.

Enabling Federated Database Functionality

To take advantage of federated database functionality, you must do several things at the DB2 Universal Database server:

- During installation, you must have selected the distributed join option. This option installs libraries in `SQLLIB/lib` that enable your DB2 federated server to access both DB2 family and Oracle tables and views using nicknames.
- You must set the FEDERATED parameter in the database manager configuration file to YES.

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 31
- "Chapter 7. Configuring Client-to-Server Communications Using the Command Line Processor" on page 43

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 catalog an APPC node, the access method is DRDA.

  To catalog an APPC node, issue the following command:

© Copyright IBM Corp. 1993, 1999

```
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 catalog a TCP/IP node, the access method is DB2RA unless the DB2 family data source is not capable of supporting DB2RA. DB2 for MVS, for example, does not support DB2RA. In these cases, DRDA is used as the access method.

  To catalog a TCP/IP node, issue the following 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.

See 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. See the *Administration Guide, Design and Implementation* 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. See the *Administration Guide, Design and Implementation* 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/OS390 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. See the *SQL Reference, Volume 2* 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 TO SERVER DB2SERVER
OPTIONS ( REMOTE_AUTHID "db2admin", REMOTE_PASSWORD "dayl1te")
```

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.
- dayl1te 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

See the *SQL Reference, Volume 2* for more information about the CREATE NICKNAME statement.

See the *Administration Guide, Design and Implementation* 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 457 before continuing with this section.

Step 1. If you have not already done so, create the sample database on your DB2 federated server by issuing the following command:

```
DB2SAMPL
```

Step 2. Connect to the sample database on your DB2 federated server:

```
CONNECT TO SAMPLE
```

Step 3. Issue the following SQL statement:

```
SELECT * FROM SYSCAT.SYSTABLES
```

You should receive the entire contents of the SYSCAT.TABLES system catalog table.

Step 4. If you have not already done so, create the sample database at your DB2 family data source.

Step 5. 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 6. 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 30. Setting Up a Federated System to Access Oracle Data Sources" on page 463.

Chapter 30. Setting Up a Federated System to Access Oracle Data Sources

Two different Oracle wrapper modules are included with DB2 Universal Database: 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:

- "Enabling Federated Database Functionality"
- "Adding Oracle Data Sources to a Federated System"
- "Oracle Code Page Options" on page 470
- "Verifying Connections to Oracle Data Sources" on page 470

The instructions in this chapter apply to Windows NT and supported UNIX platforms; platform-specific differences are noted where they occur.

Enabling Federated Database Functionality

To take advantage of federated database functionality, you must do several things at the DB2 Universal Database server:

- During installation, you must have selected the distributed join option. This option installs libraries in `SQLLIB/lib` that enable your DB2 federated server to access both Oracle and DB2 family tables and views using nicknames.
- You must set the FEDERATED parameter in the database manager configuration file to YES.

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.

© Copyright IBM Corp. 1993, 1999

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. For DB2 federated servers running on UNIX platforms, run the db2djlink.sh script to link-edit the Oracle SQL*Net or Net8 libraries to your DB2 federated server. The db2djlink.sh script is located in */install_directory*/lib. Run this script only after installing Oracle's client software on the DB2 federated server.

Step 3. 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 multiple parallel processing (MPP) environment, 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-MPP environment, there can be only one DB2DJ.ini file per instance.

Proceed to 3.b on page 465 if the default settings in DJ.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
```

See "Oracle Code Page Options" on page 470 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 NT 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-MPP environment, or if you want the values in this DB2DJ.ini file to apply to the current node only, issue:

```
db2set DB2_DJ_INI = sqllib/cfg/db2dj.ini
```

If you are using this DB2DJ.ini file in an MPP environment and you want the values in this DB2DJ.ini file to apply to all nodes within this instance, issue:

```
db2set -g DB2_DJ_INI = sqllib/cfg/db2dj.ini
```

If you are using this DB2DJ.ini file in an MPP environment, and you want the values in this DB2DJ.ini file to apply to a specific node, issue:

```
db2set -i INSTANCEX 3 DB2_DJ_INI = sqllib/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 4. 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 5. Recycle the DB2 instance:

For Windows NT servers:

```
NET STOP instance_name
NET START instance_name
```

For UNIX servers:

```
db2stop
db2start
```

Step 6. 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. See the *Administration Guide, Design and Implementation* for more information about wrapper library names.

Step 7. 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. See the *Administration Guide, Design and Implementation* for more information about wrapper library names.

Step 8. Use the CREATE SERVER statement to define each Oracle server to which communications are configured; for example:

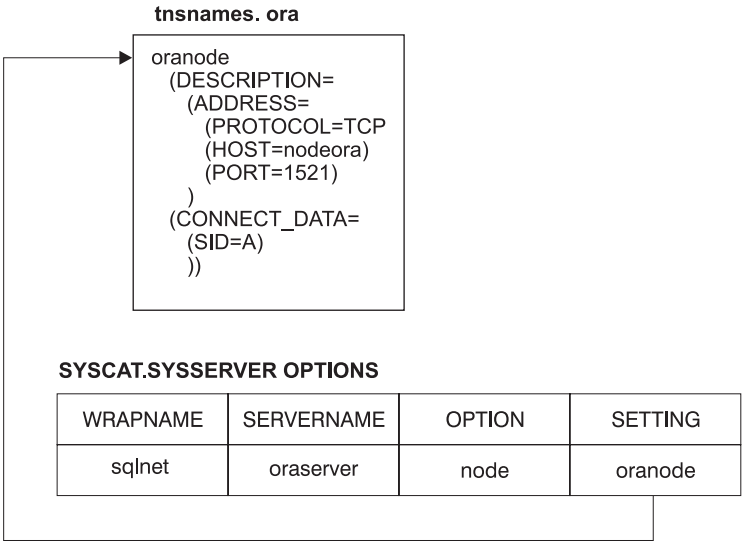
```
CREATE SERVER ORASERVER TYPE ORACLE VERSION '7.2' WRAPPER SQLNET
OPTIONS (NODE "oranode")
```

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 4 on page 468 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. See the *SQL Reference, Volume 2* for a comprehensive list of options.

  Figure 4 on page 468 shows the information in the tnsnames.ora file, the SYSCAT.SERVEROPTIONS view, and the SYSCAT.SERVERS view.



SYSCAT.SYSSERVER OPTIONS

WRAPNAME	SERVERNAME	OPTION	SETTING
sqlnet	oraserver	node	oranode

SYSCAT.SERVERS

SERVER	TYPE	VERSION	WRAPPER
oraserver	oracle	7.2	sqlnet

Figure 4. Relationship between DB2 System Files and the Oracle tnsnames.ora File

Step 9. 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 TO SERVER ORASERVER
OPTIONS ( REMOTE_AUTHID "orauser", REMOTE_PASSWORD "dayl1te")
```

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.

- `dayl1te` is the password associated with `"orauser"`. This value is case sensitive.

Step 10. 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

See the *SQL Reference, Volume 2* for more information about the CREATE NICKNAME statement.

See the *Administration Guide, Design and Implementation* for more information about nicknames in general.

Step 11. Repeat the previous step for all database objects that you want create nicknames for.

Step 12. For each HOST in the DESCRIPTION section of the tnsnames.ora file, update /etc/hosts for UNIX servers and the TCP/IP hosts file for Windows NT servers if necessary.

Whether you must update /etc/hosts or the TCP/IP hosts 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 /etc/hosts or 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" on page 470.

To verify that you successfully configured your federated server to access data sources, go to "Verifying Connections to Oracle Data Sources" on page 470.

Oracle Code Page Options

Table 35 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. See your data source documentation for more information.

Table 35. 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 463 before continuing with this section.

Step 1. Create the sample database on your DB2 federated server by issuing the following command:

```
DB2SAMPL
```

Step 2. Connect to the sample database on your DB2 federated server:

```
CONNECT TO SAMPLE
```

Step 3. Issue the following SQL statement:

```
SELECT * FROM SYSCAT.SYSTABLES
```

You should receive the entire contents of the SYSCAT.TABLES system catalog table.

Step 4. 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 5. 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.

Part 10. Appendixes

© Copyright IBM Corp. 1993, 1999

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 Software Registration Tool".
- "Starting the Client Configuration Assistant".
- "Starting the Control Center" on page 476.
- "Entering Commands Using the Command Center" on page 477.
- "Entering Commands Using the Command Line Processor" on page 478.
- "Working with the System Administrative Group" on page 480.

Starting the Software Registration Tool

Start the Software Registration Tool as follows:

OS/2 Click on **OS/2 Warp** and select **DB2 for OS/2–>Registration**

Windows 9x or Windows NT
Click on **Start** and select **Programs->DB2 for Windows->Registration**

UNIX DB2 software registration is handled automatically if you installed your DB2 product from CD-ROM using the DB2 Installer program. If you installed DB2 using your UNIX operating system's native installation tools, you must enter the **db2licm** command to register DB2. See the *Command Reference* for further information.

Starting the Client Configuration Assistant

Start the Client Configuration Assistant (CCA) as follows:

OS/2 Click on **OS/2 Warp**, and select **DB2 for OS/2–>Client Configuration Assistant**

Windows 9x or Windows NT
Click on **Start** and select **Programs->DB2 for Windows->Client Configuration Assistant**

© Copyright IBM Corp. 1993, 1999

You can also start the CCA by entering the **db2cca** command at a command prompt.

Starting the Control Center

Start the Control Center in one of the following ways:

On OS/2

- Enter the **db2cc** command at a command prompt.

  or

- Click on the **OS/2 Warp** button and select **DB2 for OS/2–>Control Center**.

  Note: You must have the prerequisite Java Runtime Environment (JRE) level to launch the Control Center using this command. For more information, refer to the Control Center readme, which can be found in the `x:\sqllib\cc\prime` directory, where *x:* is drive where you installed your DB2 product.

On UNIX

- Enter the **db2cc** command at a command prompt.

  Note: You must have the prerequisite Java Runtime Environment (JRE) level to launch the Control Center using this command. For more information, refer to the Control Center README, which can be found in the *INSTHOME*`/sqllib/cc/prime` directory, where *INSTHOME* is the home directory of the instance owner.

On Windows 9x or Windows NT

- Enter the **db2cc** command at a command prompt.

  or

- Click on **Start** and select **Programs->DB2 for Windows->Control Center**.

You can also run the Control Center as an applet through a web browser. For more information, refer to the Control Center readme, which can be found in the `x:\sqllib\cc\prime` directory, where *x:* is drive where you installed your DB2 product.

Entering Commands Using the Command Center

This section describes how to enter commands using the Command Center. The Command Center provides an interactive window to:

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

Start the Command Center in one of the follows ways:

- Click on the **Command Center** icon in the Control Center.
- Enter the **db2cctr** command.

  Note: You must have the prerequisite Java Runtime Environment (JRE) level to launch the Command Center using this command.
- Launch the Command Center from the toolbar as follows:

  On Windows 9x or Windows NT

  – Click on the **Start** button and select **Programs->DB2 for Windows->Command Center**.

  On OS/2

  – Click on the **OS/2 Warp** button and select **DB2 for OS/2 –>Command Center**.

The Command Center contains a large input area in which you enter your commands. To run the commands you have entered, click on the **Execution** icon (the gears icon), or press **CTRL+Enter**.

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 then **Enter** key to start the next command on a new line. The default termination character is a semicolon (;). To specify a different termination character, click on the **Tools Settings** icon in the menu toolbar.

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 **Execution** icon (or pressed **CTRL+Enter**), the Command Center switches to the Results window which informs you how the commands are proceeding.

To recall commands that you have entered, select the **Script** Tab, click on the drop down box, and select a command.

To save commands as scripts, select **Script->Save as** from the menu bar. For more information, click on the **Help** push button or press the **F1** key.

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

Command Line Mode
: The DB2 command line processor behaves like a command window from your operating system. You can enter operating system commands, DB2 commands, or SQL statements and view their output.

Interactive Input Mode
The db2 prefix that you use for DB2 commands (in Command Line Mode) 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*.

Command Line Mode

To invoke a DB2 Command window, do the following:

OS/2 Click on **OS/2 Warp**, and select **DB2 for OS/2->Command Line Processor**, or open any command window.

Windows 9x or Windows NT
Click on **Start** and select **Programs->DB2 for Windows->Command Window**

You can also invoke the DB2 command window by entering the **db2cmd** command at an MS-DOS 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"
```

If you need to enter a long command that does not fit on one line, use the line continuation character, \. When you have reached the end of the line, press the **Enter** key to continue entering the command on the next line. 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 **DB2 for OS/2–>Command Line Processor** or enter the **db2** command.

Windows 9x or Windows NT
Click on **Start**, and select **Programs->DB2 for Windows->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 an MS-DOS 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
```

If you need to enter a long command that does not fit on one line, use the line continuation character, \. When you have reached the end of the line, press the **Enter** key to continue entering the command on the next line. 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 username that belongs to the primary group of the instance owner's user ID.

Windows 9x Any Windows 9x user.

Windows NT 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 file to specify a new group. If you want that group enumerated on the local machine, the you must also set the `DB2_GRP_LOOKUP` registry variable.

By default, in a Windows NT domain environment, only domain users that belong to the Administrators group at the Primary Domain Controller (PDC) have SYSADM authority on an instance. 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 PDC, we suggest that you create a global group and add the users (both domain and local) that you want to grant SYSADM authority. 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*.

Appendix B. Migrating from Previous Versions and Releases

This section describes how to migrate previous versions of DB2 to the Version 6 format.

DB2 Universal Database Version 6 supports the migration of DB2 Version 2.x, Database Server Version 4x, and DB2 Version 5.x to a format usable by DB2 Version 6.

Migrating Instances

Before you can migrate an instance to use the latest version of DB2, you must install DB2 Version 6 on your system.

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

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

Step 1. Prepare the DB2 instance for migration.

Step 2. Migrate the DB2 instance.

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

Step 1. Prepare the DB2 Instance for Migration

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

Step 1. Log in as the DB2 instance owner.

Step 2. Stop all command line processor sessions by entering the **db2 terminate** command in each session that was running the command line processor.

Step 3. Enter the **db2_kill** command to clean up any remaining DB2 resources.

Step 4. Log off.

The DB2 instance is now ready for migration.

© Copyright IBM Corp. 1993, 1999

Step 3. Migrate the DB2 Instance

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

1. Log in as user with root authority.

If the *library_path* environment variable is set to `/usr/lib` on AIX or `/opt/lib` on HP-UX, or Solaris, and there is a link in `/usr/lib` or `/opt/lib` to the Version 6 `libdb2` shared library, this can cause an error when using the **db2imigr** command. To fix the error, you should reset the *library_path* environment variable so that it does not reference the libraries in those paths by entering the following command:

```
unset library_path
```

where *library_path* represents:

- *LIBPATH* on AIX
- *SHLIB_PATH* on HP-UX
- *LD_LIBRARY_PATH* on Solaris

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

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

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

where *DB2DIR* = `/usr/lpp/db2_06_01` on AIX
= `/opt/IBMdb2/V6.1` on HP-UX, Solaris, or SGI IRIX

and where:

-d Sets the debug mode that you can use for problem determination. This parameter is optional.

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

Notes:

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

-u *fencedID*

Is the user under which the fenced user-defined functions (UDFs) and stored procedures will execute. This parameter is optional only when a DB2 Run-Time Client is installed. It is required for all other DB2 products.

InstName

Is the login name of the instance owner.

If you are migrating a DB2 Version 2.1 or Version 5 instance, created on AIX, and the instance uses the environment variable *DB2SORT* set to a keyword *SMARTSORT*, you must set the registry value *db2sort* after the instance is migrated to Version 6. Set the *db2sort* registry value to the run time library for the sort command as follows:

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

Appendix C. Troubleshooting NetQuestion

This section describes installation prerequisites, workarounds for known issues, advanced configuration, and uninstall procedures for NetQuestion on OS/2 and Windows 32-bit operating systems. This section also describes workarounds for known NetQuestion install issues on UNIX operating systems.

NetQuestion is an HTML search system that is installed automatically if you choose to install product documentation with any DB2 product. Provided your operating system meets NetQuestion's installation prerequisites, there is nothing special you have to do to install and work with NetQuestion.

There are two versions of NetQuestion:

- The *single-byte character set* (SBCS) version is installed with documentation in languages such as English, French, German, and Spanish.
- The *double-byte character set* (DBCS) version is installed with documentation in Japanese, Simplified Chinese, Traditional Chinese, and Korean.

The following topics are covered in this section:

- "NetQuestion for Windows 32–bit Operating Systems"
- "NetQuestion for OS/2 Operating Systems" on page 497
- "Troubleshooting NetQuestion Installs on UNIX operating systems" on page 505

NetQuestion for Windows 32–bit Operating Systems

Provided your system meets the installation prerequisites detailed in "NetQuestion Prerequisites" on page 488, NetQuestion is usually installed automatically whenever DB2 product documentation is installed, and the search server is started whenever your Windows system is booted. To search the DB2 product documentation all you have to do is open the HTML search form (named **db2srch.htm**, located in `x:\sqllib\doc\html` where `x:\` is the DB2 installation drive) in a browser such as Netscape Navigator or Internet Explorer, enter your search criteria, and click on the **Search** button.

Restrictions on Use

- NetQuestion is not available on Windows 3.1.

© Copyright IBM Corp. 1993, 1999

- NetQuestion is not enabled for Thin Clients. If you attempt to search the documentation, you will receive a network error.

Installing NetQuestion

NetQuestion Prerequisites

The following prerequisites are required to install and work with NetQuestion:

- On Windows NT 4.0, for best results you should have installed Service Pack 3. For more information, go to `http://www.microsoft.com/ntserver/info/servicepack3.htm`.
- Ensure that you turn off proxy handling for **localhost** in the browser you use.
- 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.

Stopping Any Previously Installed Version of NetQuestion Before Installing DB2

If NetQuestion was previously installed with another product (for example, IBM VisualAge for Java), the search server must be stopped before you install DB2 with product documentation selected. To stop the search server, enter one of the following commands:

```
imnss stop server            //for SBCS versions
imqss -stop dbcshelp         //for DBCS versions
```

Locating the NetQuestion 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
```

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 `db2srch.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->DB2 for Windows–>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 startup. 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
```

Diagnosing NetQuestion Install Errors

If you received error messages while installing NetQuestion, perform the following steps to diagnose and fix the problem:

Step 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.
```

Step 2. If imnnq.err contains a message saying <*file_name*>.EXE DOES NOT EXIST, the NetQuestion executables could not be found. Try installing DB2 product documentation again. For more information see "If Search Does Not Work (ERROR 500 when trying to search)" on page 493.

Working with NetQuestion

Starting and Stopping the Search Server

After you install NetQuestion, the search server will start automatically when you reboot your system.

If the search server starts without any errors but you encounter the following error message when you attempt to search the DB2 product documentation:

```
A network error occurred: unable to connect to server.
The server may be down or unreachable.
Try connecting again later.
```

then stop and then re-start the search server by clicking on the appropriate icon in **Start->Programs->DB2 for Windows**.

If you run low on memory, stop the search server to reclaim the memory it used. To stop the search server, click on **Start->Programs->DB2 for Windows –>Stop HTML Search Server**.

To stop and start the SBCS search server from a command prompt, enter the following commands:

```
imnss stop server
imnss start server
```

To stop and start the DBCS search server from a command prompt, enter the following commands:

```
imqss -stop dbcshelp
imqss -start dbcshelp
```

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 3, perform the following steps to modify your proxy information:

Step 1. Select **Options->Network Preferences**.

Step 2. Click on the **Proxies** tab.

Step 3. Click **View at the Manual Proxy Configuration** selection.

Step 4. In the **No proxies for** box, type:

```
localhost:49213
```

If you have other entries here, separate them with commas.

Step 5. Click **OK** until all dialog boxes are closed.

In Netscape 4, perform the following steps to modify your proxy information:

Step 1. Select **Edit->Preferences**.

Step 2. Double-click **Advanced** in the **Category** tree.

Step 3. Click **Proxies** in the **Advanced** sub-tree.

Step 4. Click **View at the Manual Proxy Configuration** selection.

Step 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.

Step 6. Click **OK** until all dialog boxes are closed.

In Internet Explorer 3, perform the following steps to modify your proxy information:

Step 1. Select **View->Options**.

Step 2. Select **Connection**.

Step 3. 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.

Step 4. Select the **Do not use proxy server for local (intranet) addresses** box.

Step 5. Click **OK** to close the **Options** window.

In Internet Explorer 4, perform the following steps to modify your proxy configuration:

Step 1. Select **View->Internet Options**.

Step 2. Select the **Connections** tab.

Step 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.

Step 4. Click on **Advanced**.

Step 5. In the **Exceptions...Do not use proxy server for addresses beginning with** box, type

```
localhost:49213
```

Step 6. Click **OK** until all dialog boxes are closed.

In Internet Explorer 5 (beta), perform the following steps to modify your proxy configuration:

Step 1. Select **Tools->Internet Options**.

Step 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**.

Step 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.

Step 4. Click on **Advanced**.

Step 5. In the **Exceptions...Do not use proxy server for addresses beginning with** box, type

```
localhost:49213
```

Step 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 allow 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 (connected or disconnected from a LAN). TCPSwitch is one of these programs.

If you are using Netscape 3 and you dial-in to a secure site to access the internet with a proxy or socks server, you will need to delete these settings in Netscape before you can search. This is a limitation in Netscape 3.

If Search Does Not Work (ERROR 500 when trying to search)

If the DB2 product installation was successful, but searching the product documentation with NetQuestion does not work, perform the following steps to diagnose and fix the problem:

Step 1. Check that NetQuestion was properly installed. The environment variables `IMNINST` and `IMNINSTSRV` should be set and `IMNINSTSRV` should point to the NetQuestion directory. For more information on locating the NetQuestion directory see “Locating the NetQuestion Directory” on page 489.

Step 2. The NetQuestion directory should contain these files:

- A `DB2SRSxx` or `DB2SRDxx` executable file where `xx` is a two-character identifier of the language in which the documents are written.
- `DB2HEAD.HTM` and `DB2FOOT.HTM`

Step 3. Ensure that the product documentation indexes are registered with NetQuestion. To list all registered indexes enter one of the following commands:

```
nqmap -a              //for SBCS
tmmap -a              //for DBCS
```

The documentation indexes for DB2 is called DB2SR6xx or DB2CC6xx where xx is a two-character identifier of the language in which the documents are written. One or more of these names should appear in the list of names that NQMAP (or TMMAP) returns.

With the list of indexes the NQMAP (or TMMAP) command provides, you can get additional details on each index with the following command:

```
imnixsta <index_name>         //for SBCS
imqixsta <index_name>         //for DBCS
```

where <*index_name*> is the name of one of the indexes returned by the nqmap -a (or tmmap -a) command.

This command provides details such as the status of the index and the number of documents in the index.

If any of the above commands reveal an error, you can run the DB2 Universal Database product installation program again. If the only problem is that the files DB2SRCH.EXE, DB2HEAD.HTM or DB2FOOT.HTM are missing, then you can copy them from the directory <db2path>\misc to the NetQuestion directory, (for example, C:\IMNNQ_NT). The product installation program will run the search server's installation and initialization again.

″File Not Found″ Errors

NetQuestion searches predefined indexes (not the actual HTML documentation installed with your DB2 product) and displays a list of ranked *hits* in a **Search Results** page. All of these indexes are installed with NetQuestion, regardless of the DB2 products you install.

However, depending on the DB2 product you installed, some of the HTML documentation may not be present. Therefore, the links generated from the predefined indexes may, if followed, return ″File not found″ errors.

You can select the most commonly installed sets of books: administration, application programming, and DB2 Connect. You can also search some individual books, such as the *SQL Reference*. For example, you install DB2 Universal Database Personal Edition with product documentation selected.

You load the NetQuestion search page and search for ″Embedded SQL″. Most of the hits returned in the **Search Results** page will (if you ″mouse over″ the links) point to URLs like FILE:///D:\SQLLIB\DOC\HTML\db2a0/db2a093.htm, which is a file in the HTML version of the *Application Development Guide*. However, this book (db2a0) is not installed with DB2 Universal Database Personal Edition. Therefore, the links in the **Search Results** page will, if followed, produce ″File not found″ errors.

You can determine which HTML books are installed by clicking on **Start->Programs->DB2 for Windows –>Information Center** and selecting the **Books** tab. You can also look in *x:*\sqllib\doc\html (where *x:* is the drive on which you installed DB2) and comparing the five-character folder names (such as db2s0 and db2h1) to the list of books in the ″How the DB2 Library is Structured″ appendix in your *Quick Beginnings* manual.

Uninstalling NetQuestion

Before uninstalling NetQuestion, make sure the search server is stopped. For information on how to stop the search server, see “Starting and Stopping the Search Server” on page 491.

NetQuestion is normally uninstalled when you uninstall DB2 Universal Database by selecting **Start->Programs->DB2 for Windows –>Uninstall**. 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 products) are still registered with NetQuestion. To determine which products are still registered with NetQuestion, issue one of the following commands:

```
nqmap -a            //for SBCS
tmmap -a            //for DBCS
```

If this command lists no indexes, go to “Manually Uninstalling NetQuestion” on page 497.

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 (DB2SR6xx or DB2CC6xx), DB2 could not unregister the indexes therefore causing the NetQuestion

uninstall to fail. This occurs if DB2 was incorrectly uninstalled (for example, the SQLLIB folder was deleted). In this case, you need to manually unregister the indexes one at a time by performing the following steps:

Step 1. Issue one of these commands to ensure that the search server is running:

```
imnss start server          //for SBCS
imqss -start dbcshelp       //for DBCS
```

Step 2. Issue one of these commands for each of the index files:

```
nqmap -d <index_name>       //for SBCS
tmmap -d <index_name>       //for DBCS
```

where <*index_name*> is the name of one of the indexes returned by the `nqmap -a` (or `tmmap -a`) command.

Step 3. Issue one of these commands for each of the index files:

```
nqdelet <index_name>        //for SBCS
tmdelet <index_name>        //for DBCS
```

where <*index_name*> is the name of one of the indexes returned by the `nqmap -a` (or `tmmap -a`) command.

Step 4. Stop the search server:

```
imnss stop server           //for SBCS
imqss -stop dbcshelp        //for DBCS
```

Step 5. Issue `nqmap -a` (or `tmmap -a`) and verify that there are no DB2 indexes remaining. If there are, please contact IBM service.

Step 6. Issue one of following commands to verify that no other indexes are active.

```
nqcounti <NetQuestion directory>  //for SBCS
tmcounti <NetQuestion directory>  //for DBCS
```

where <*NetQuestion directory*> is the fully qualified path under which NetQuestion is installed. See “Locating the NetQuestion Directory” on page 489 if you do not know the directory under which NetQuestion is installed.**CAUTION:**
If the `nqcounti` (or `tmcounti`) command return data that indicates one or more indexes are still active, NetQuestion cannot be removed because other 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 page 497.

Manually Uninstalling NetQuestion

If the `nqmap -a` (or `tmmap -a`) command does not list any active indexes, you can try removing NetQuestion manually by performing the following steps:

Step 1. From a command prompt, enter the **uninstnq** command.

Step 2. Remove the registry entries that are under **\\HKEY_LOCAL_MACHINE\SOFTWARE\IBM\NetQuestion**. Remove this entry including all of its subtrees.

Step 3. Remove the NetQuestion directory and all of its subtrees. For example, if you installed NetQuestion on the `C:` drive, remove the `C:\imnnq_nt` directory.

Step 4. Remove the environment variables `IMNINST` and `IMNINSTSRV` and remove the NetQuestion 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 doing another DB2 install. 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

Provided your system meets the installation prerequisites detailed in "NetQuestion Prerequisites", NetQuestion is usually installed automatically whenever DB2 product documentation is installed, and the search server is started whenever your OS/2 system is booted. To search the DB2 product documentation all you have to do is open the HTML search form (**db2srch.htm**, located in `x:\sqllib\doc\html` where `x:\` is the DB2 install drive) in a browser such as Netscape 2.02, enter your search criteria, and click on the **Search** button.

Installing NetQuestion

NetQuestion Prerequisites

The following prerequisites are required to install and work with NetQuestion:

- You must have TCP/IP Version 3 or higher installed on your machine. OS/2 Warp Version 4 is recommended because it has the appropriate level

of TCP/IP. OS/2 Warp Version 3 Internet Access Kit (IAK) will also work if you set it up for local loopback and if you install the latest FixPak.

- You need a browser, such as Netscape 2.02 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++.

Configuring TCP/IP

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.

Enabling Local Loopback: To enable local loopback on your system:

Step 1. Open the **OS/2 TCP/IP** folder.

Step 2. Open the **TCP/IP Configuration** notebook.

Step 3. View the **Network** page.

Step 4. In the **Interface to Configure** list box, highlight **loopback interface**.

Step 5. If the **Enable interface** check box is not selected, select it now.

Step 6. Verify that the **IP address** is `127.0.0.1` and **Subnet Mask** is empty.

Enabling Localhost: To enable localhost on your system:

Step 1. To check if localhost is enabled, enter the **ping localhost** command.
- If data are returned, localhost is enabled, and 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.

Step 2. If you are on a network, make sure that loopback is enabled. To enable local loopback see “Enabling Local Loopback”.

Step 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
```

Note: If you have OS/2 Warp with Internet Access Kit, add the line to the \STARTUP.CMD instead of the MPTN\BIN\SETUP.CMD file. If the file doesn’t exist, you will have to create it.

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.

Step 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 **TCPIP 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.

Step 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*> is 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

Click the **Start HTML Search Server** icon, which is located in the **DB2 for OS/2** folder. If an error message appears, TCP/IP is not configured properly. Follow the instructions in "Configuring TCP/IP" on page 498 to make sure that TCP/IP is correctly configured. Reboot OS/2 if you change any settings.

Locating the NetQuestion 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 this directory. To locate this directory enter one of the following commands:

```
echo %IMNINSTSRV%       //for SBCS versions
echo %IMQINSTSRV%       //for DBCS versions
```

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 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 `db2srch.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.
4. Stop and re-start the search server by clicking on the appropriate icons in the **DB2 for OS/2** folder.

Diagnosing NetQuestion Install Errors

If you received error messages while installing NetQuestion, enter the **SNIFFLE /P** command and follow the instructions that it returns

Make sure you reboot your system after you are finished correcting any NetQuestion install errors.

Working with NetQuestion

Starting and Stopping the Search Server

After you install NetQuestion, the search server will start automatically after you reboot.

If the search server starts without any errors but you encounter the following error message when you attempt to search the DB2 product documentation:

```
A network error occurred: unable to connect to server.
The server may be down or unreachable.
Try connecting again later.
```

then stop and re-start the search server by clicking on the appropriate icons in the **DB2 for OS/2** folder.

After you have finished searching, stop the search server to reclaim the memory it used.

Searching while Disconnected from the Network

If you are not on a network (for example, if you are using a laptop computer and are temporarily away from a LAN connection), you have to enable localhost to search the documentation. For more information on enabling localhost, see "Configuring TCP/IP" on page 498.

"File Not Found" Errors

NetQuestion searches predefined indexes (and not the actual HTML documentation installed with your DB2 product) and displays a list of ranked *hits* in a **Search Results** page. All of these indexes are installed with NetQuestion, regardless of the DB2 products you install.

However, depending on the DB2 product you installed, some of the HTML documentation may not be present. Therefore, the links generated from the predefined indexes may, if followed, return "File not found" errors.

For example, you install DB2 Universal Database Personal Edition with product documentation selected. You load the DB2 search page, select the **Application Development** index, and search for "Embedded SQL". Most of the hits returned in the **Search Results** page will (if you "mouse over" the links) point to URLs like `FILE:///C:\SQLLIB\DOC\HTML\db2a0/db2a093.htm`, which is a file in the HTML version of the *Application Development Guide*. However, this book (db2a0) is not installed with DB2 Universal Database Personal Edition. Therefore, the links in the **Search Results** page will, if followed, produce "File not found" errors.

You can determine which HTML books are installed by opening the DB2 for OS/2 folder, double-clicking on **Information Center**, and selecting the **Books** tab. You can also look in `<installation drive>:\sqllib\doc\html` and comparing the five-character folder names (such as `db2s0` and `db2h1`) to the list of books in the "How the DB2 Library is Structured" appendix in your *Quick Beginnings* manual.

Error 500 Detected When Trying to Search

If you installed NetQuestion but you get an `Error 500` when you try to search the documentation, perform the following steps to diagnose and fix the problem:

Step 1. Check that NetQuestion was properly installed. The environment variable `IMNNLPSSRV` should point to the NetQuestion directory. For

information on locating the NetQuestion directory see “Locating the NetQuestion Directory” on page 499.

Step 2. The NetQuestion directory should contain these files:

- A DB2SRSxx or DB2SRDxx executable file where xx is a two-character identifier of the language in which the documents are written.
- A DB2HEAD.HTM and DB2FOOT.HTM file. If these files are not present, run **db2netq.cmd** to set up index registration and copy the files to the NetQuestion directory.

Step 3. Ensure that the product documentation indexes are registered with NetQuestion. To list all registered indexes enter one of the following commands:

```
nqmap -a              //for SBCS
tmmap -a              //for DBCS
```

The documentation indexes for DB2 is called DB2SR6xx or DB2CC6xx where xx is a two-character identifier of the language in which the documents are written. One or more of these names should appear in the list of names that NQMAP (or TMMAP) returns.

With the list of indexes the NQMAP (or TMMAP) command provides, you can get additional details on each index with one of the following commands:

```
imnixsta <index_name>         //for SBCS
imqixsta <index_name>         //for DBCS
```

where <*index_name*> is the name of one of the indexes returned by the nqmap -a (or tmmap -a) command.

The imnixsta (or imqixsta) command provides details such as the status of the index and the number of documents in the index.

If any of the above commands reveal an error, either TCP/IP is incorrectly installed or the NetQuestion indexes are corrupt. Use the **SNIFFLE /P** command to diagnose and fix TCP/IP errors, and reinstall the documentation to fix index corruption problems.

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:\netqos2\data`.

Uninstalling NetQuestion

Before uninstalling NetQuestion, make sure the search server is stopped. For information on how to stop the search server, see "Starting and Stopping the Search Server" on page 500.

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 page 498 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:

```
nqmap -a              //for SBCS
tmmap -a              //for DBCS
```

If this command lists no indexes, go to "Manually Uninstalling NetQuestion" on page 505.

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 (DB2SR6xx or DB2CC6xx), DB2 could not unregister the indexes therefore causing the NetQuestion uninstall to fail. This occurs if DB2 was incorrectly uninstalled (for example, the SQLLIB folder was deleted). In this case, you need to manually unregister the indexes one at a time by performing the following steps:

Step 1. Issue one of these commands to ensure that the search server is running:

```
netq start server          //for SBCS
imqss -start dbcshelp      //for DBCS
```

Step 2. Issue one of these commands for each of the index files:

```
nqmap -d <index_name>     //for SBCS
tmmap -d <index_name>     //for DBCS
```

where <*index_name*> is the name of one of the indexes returned by the nqmap -a (or tmmap -a) command.

Step 3. Issue one of these commands for each of the index files:

```
nqdelet <index_name>      //for SBCS
tmdelet <index_name>      //for DBCS
```

where <*index_name*> is the name of one of the indexes returned by the nqmap -a (or tmmap -a) command.

Step 4. Stop the search server:

```
netq stop server           //for SBCS
imqss -stop dbcshelp       //for DBCS
```

Step 5. Issue nqmap -a (or tmmap -a) and verify that there are no DB2 indexes remaining. If there are, please contact IBM service.

Step 6. Issue the following command to verify that no other indexes are active.

```
nqcounti <NetQuestion directory>  //for SBCS
tmcounti <NetQuestion directory>  //for DBCS
```

where <*NetQuestion directory*> is the fully qualified path under which NetQuestion is installed. See “Locating the NetQuestion Directory” on page 499 if you do not know the directory under which NetQuestion is installed.**CAUTION:**
If the nqcounti (or tmcounti) command return data that indicate one or more indexes are still active, NetQuestion cannot be removed because other 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 page 497.

Manually Uninstalling NetQuestion

If the `nqmap -a` (or `tmmap -a`) command does not list any active indexes, you can try removing NetQuestion manually by performing the following steps:

Step 1. From a command prompt, enter

```
uninstnq.cmd
```

Ignore this message: ″SYS0016: The directory cannot be removed.″

Step 2. Remove the NetQuestion directory and all of its subtrees. For example, if you installed NetQuestion on the `D:` drive, remove the `d:\netqos2` directory.

Step 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*).

Step 4. Reboot your system.

Troubleshooting NetQuestion Installs on UNIX operating systems

This section describes solutions to problems that might be encountered while installing NetQuestion on UNIX platforms. NetQuestion is installed whenever product documentation is selected during the installation of DB2 Universal Database.

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.

Installing NetQuestion on HPUX-10 operating systems

When installing DB2 Universal Database Version 6 for HP-UX 10, you may find that NetQuestion cannot be installed from the product CD-ROM. You may discover that when you attempt to search the DB2 product documentation, you get a message stating that the ″connection is refused by server″. Check if NetQuestion is installed with the following command:

```
swlist -l product | grep -i imn
```

If nothing is returned, then NetQuestion has not been installed.

In this case, you must install NetQuestion manually. Note that you must be root to install NetQuestion.

Copy the entire /NetQ directory from the installation CD-ROM to your local hard drive. After copying is finished, you should have the following files in your local /NetQ/hpux10 directory: IMNSearch, IMNSearchB, IMNpkgadd, IMNpkgaddB, imnbinstall and imninstall.

To start the NetQuestion installation script, enter the **imninstall** command. Follow the on-screen directions to install either the SBCS search engine alone or the SBCS and DBCS search engines together.

Troubleshooting NetQuestion installs on HPUX-10 operating systems

If imninstall returns an error stating ″swcopy failed, see manual pages for swcopy″, it is possible that some other file systems completely unrelated to DB2 have failed to mount the filesystem. In this case, you must (as root) edit your local copy of /NetQ/hpux10/IMNpkgadd by adding the option:

```
-x mount_all_filesystems=false
```

to each occurrence of swcopy, swremove or swinstall.

Appendix D. 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 indexes 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 simple to use the Information Center from a remote machine to access files on the web server.
- You cannot use the HTML search facility and search indexes that come with DB2. You should replace or remove the default search form to avoid confusion.
- An administrator might need to manage network load on the web server.

© Copyright IBM Corp. 1993, 1999

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 (some of which are marked as **optional**):

Step 1. Install a DB2 server, Administration Client, Software Developer's Kit, or DB2 Connect product and select the option to install documentation. 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. On UNIX systems, or with the Application Developer's Kit, the documentation set also includes the programming books such as the *Administrative API Reference*, *Call Level Interface Guide and Reference*, or the *Application Development Guide*.

Step 2. As an **option**, you can customize the HTML files you have installed. The "master file" for the DB2 documentation library is `index.htm` in the product's `doc\html` subdirectory. 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.

Note: 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.

The "master file" for each book is `index.htm`, located in each book's subdirectory.

The search form used by all the HTML files is `nqsrch.htm` in the product's `doc\html` subdirectory. In cases where the search facility is not available or you want to use a different one, you can replace this file with one that notifies users that search is not available or performs the alternative search. For example:

```
<html>
<head>
<title>IBM DB2 Universal Database: Search Not Available</title>
</head>
<body>
<h1>Search Not Available</h1>
<p>The search function is not available on this platform.</p>
</body>
</html>
```

Step 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.

Step 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` subdirectory 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 `sqllib\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 the *SQL Reference* files are in a directory named `db2s0`, and so on.

Searching the DB2 Books from a Remote Client

DB2 Universal Database Version 6 enables remote clients (machines equipped only with a browser) to view and search DB2 documentation installed on a central server.

Enabling Remote Searching on Windows NT, Windows 9x, and OS/2

To enable remote searching of DB2 documentation on Windows NT, Windows 9x, and OS/2, perform the following steps:

Step 1. Edit the DB2 search form. This file is

```
x:\sqllib\doc\html\db2srch.htm
```

where x: is the drive on which DB2 is installed

Step 2. Change the action= attribute of the <form> tag. Replace localhost:49213 with *hostname*:49214, where *hostname* represents the machine where the DB2 documentation files are installed.

Step 3. Edit your IBM Internect Connection Server (ICS) Lite configuration file.

On a Windows NT, Windows 95, or OS/2 operating system, this file is httpd.cnf in the directory imnnq_nt, imnnq95, or netqos2, respectively. On Windows 98, the directory is imnnq_95.

a. Copy the original ICS Lite configuration file to another file name in the same directory as the original. In this example we'll call this copy db2httpd.cnf.

Note: By creating a copy of this configuration file, you can avoid affecting other products installed on your system that rely on the search server.

b. Replace the value of the Hostname line in db2httpd.cnf (or whatever you named the copy of the ICS Lite configuration file) with the same hostname:49214 as you did in the DB2 search form.

c. Add the following directives to db2httpd.cnf:

```
MaxActiveThreads 40
Allow <network_ip> <network_mask>
```

where:

- <network_ip> is the network IP address of the machine that will be connecting to your machine, in a.b.c.d form. A value of *0.0.0.0* is not allowed.
- <network_mask> is the network mask of the machine that will be connecting to your machine, either in a.b.c.d or x form, where x is the number of valid bits. For example, a value of *24* for x would be *255.255.255.0* in a.b.c.d form. The only valid network masks are those consisting of a consecutive sequence of 1 bits followed by an optional consecutive sequence of 0 bits.

Note: You can have more than one Allow directive, one for each machine that will be connecting to your machine.

The MaxActiveThreads directive will increase ICS Lite's performance when multiple requests are arriving over the network. You can set the value of MaxActiveThreads to greater than 40, but as this value increases more memory is consumed.

d. Change the following directive in your db2httpd.cnf file from

```
PASS /*  X:\IMNNQ_NT\*
```

to

```
PASS /*  X:\SQLLIB\*
```

where x: is drive on which NetQuestion and DB2 are installed (in this example we assume they are both on the same drive) and IMNNQ_NT is your NetQuestion installation directory. The NetQuestion directory name will vary for Win 9x and OS/2 operating systems.

e. Edit your **Start HTML Search Server** icon's properties so that it calls the new db2httpd.cnf file.

Editing the Start HTML Search Server icon on Windows systems

1) Click **Start –> Programs –> Windows Explorer**.
2) In Windows Explorer, click **Windows –> Profiles –>** <*Username*> **–> Start Menu –> Programs –> Startup** *or* **DB2 for Windows**, where <*Username*> represents the username under which DB2 was installed.

   Note: **Start HTML Search Server** icons will be located in the **Startup** and **DB2 for Windows** folders provided they were not removed after install.
3) Right-click the **Start HTML Search Server** icon, select **Properties**, and click the **Shortcut** tab.
4) In the **Target** field, add

   ```
   db2httpd.cnf
   ```

   as the last argument. If you put the new configuration file in a directory that is different than the original, you must specify the full path name.
5) Click **OK** to close.

Editing the Start HTML Search Server icon on OS/2 systems

1) Right-click on the **Start HTML Search Server** icon in your **DB2 for OS/2** folder.
2) Select **Settings** and click on the **Program** tab.
3) In the **Optional Parameters** field, add

```
config db2httpd.cnf
```

to the end of the string of parameters. If you put the new configuration file in a directory that is different than the original, you must specify the full path name.

4) Click **OK** to close.

5) Edit your db2init.cmd file. This file is located in x:\sqllib\bin\, where x: is the drive on which you installed DB2. Find the line in this file containing both "db2ss.exe" and "db2netqd.exe". At the end of this line add

```
config "db2httpd.cnf"
```

Note that you need double quotation marks around the file name. Save and exit.

Step 4. For each index that is installed, run the **nqmap** command to set the base location for the documentation files. The exact directory to use depends on your server's hostname. For example, if your server is named *yourserver*, your port number is 49214, and your DB2 documentation is in English, run the following set of commands to set the base location for both of the DB2 indexes:

```
nqmap -u DB2SR6EN "http://yourserver:49214/doc/html/" DB2SR6EN
nqmap -u DB2CC6EN "http://yourserver:49214/doc/html/" DB2CC6EN
```

If you have documentation installed in other languages, change EN in each of the above index names to the appropriate two-letter language identifier. If you have DBCS documentation installed, substitute tmmap for nqmap.

Enabling Remote Searching on AIX

To enable remote searching of DB2 documentation from an AIX client, perform the following steps:

Step 1. Log on as root.

Step 2. Install a DB2 Administration Client or any server product, plus one or more language versions of the DB2 Product Library, if they are not already installed.

Step 3. Go into the directory /etc/IMNSearch/httpdlite.

Step 4. Copy the file httpdlite.conf to db2httpd.conf.

Step 5. Edit the new file db2httpd.conf:

a. Uncomment the line with the **HostName** directive by removing the # character.

b. Change the hostname to the real machine name, such as *server.com*.

c. In the line with the **Port** directive, change the number from 49213 to 49214.

d. Add the following line:

```
Allow <network_ip> <network_mask>
```

where:

- `<network_ip>` is the network IP address of the machine that will be connecting to your machine, in `a.b.c.d` form. A value of *0.0.0.0* is not allowed.
- `<network_mask>` is the network mask of the machine that will be connecting to your machine, either in `a.b.c.d` or `x` form, where `x` is the number of valid bits. For example, a value of *24* for `x` would be *255.255.255.0* in `a.b.c.d` form. The only valid network masks are those consisting of a consecutive sequence of 1 bits followed by an optional consecutive sequence of 0 bits.

Note: You can have more than one `Allow` directive, one for each machine that will be connecting to your machine.

e. Save the file and exit from the editor.

Step 6. Go into the directory `/etc`.

Step 7. Edit the file `inittab`:

a. You will see a line that executes the **httpdlite** command with a filename argument. Make a copy of this line immediately below the original line.

b. In the new line:

1) Change the first field from **httpdlite** to **httpdlite2**.

2) Change the part of the line that reads `httpdlite.conf` to `db2httpd.conf`. The result should look like:

```
httpdlite:2:once:/usr/IMNSearch/httpdlite/httpdlite -r /etc/IMNSearch/httpdlite/ httpdlite.conf >/dev/console 2>&1
httpdlite2:2:once:/usr/IMNSearch/httpdlite/httpdlite -r /etc/IMNSearch/httpdlite/ db2httpd.conf >/dev/console 2>&1
```

c. Save the file and exit from the editor.

Step 8. Since the new **inittab** process will not be run until the machine is rebooted, run the command

```
/usr/IMNSearch/httpdlite/httpdlite -r /etc/IMNSearch/httpdlite/db2httpd.conf >/dev/console 2>&1
```

from the `/etc` directory.

Step 9. Go into the directory `/usr/lpp/db2_06_01/doc/en_US/html`

Step 10. Edit the file `db2srch.htm`:

a. Change the string `localhost` to reflect the name of your host, and change the number 49213 to 49214.

For example, if the original line is:

```
<form action="http://localhost:49213/cgi-bin/db2srsen.exe" method="POST">
```

The modified line might be:

```
<form action="http://server.com:49214/cgi-bin/db2srsen.exe" method="POST">
```

b. Save the file and exit from the editor.

Step 11. Issue the following commands, substituting your own hostname for `server.com`:

```
/usr/IMNSearch/cli/imndomap -u DB2SR6EN "http://server.com/doc/en_US/html/" DB2SR6EN
/usr/IMNSearch/cli/imndomap -u DB2CC6EN "http://server.com/doc/en_US/html/" DB2CC6EN
```

For documentation in other languages, replace the `EN` suffix in each of the index names with the appropriate two-letter locale, and replace `en_US` with the appropriate locale directory. For the Japanese, Korean, Simplified Chinese, and Traditional Chinese locales, use the **imqdomap** command instead the **imndomap** command.

Note: The exact URL to enter in the imndomap commands and to use for the search form depends on how you set up your web server to work with the DB2 Product Library. If you make a directory on the web server named **db2** or something that points to `/usr/lpp/db2_06_01`, then you would insert that directory name after the hostname in each case.

Step 12. You should now be able to start a browser on any machine in the local network, load the `doc/en_US/html/db2srch.htm` file underneath your web server, execute a search, and click on a link in the search results page to go to the documentation file.

Enabling Remote Searching on HP-UX and Solaris

To enable remote searching of DB2 documentation from a HP-UX or Solaris client, you need to perform the following steps:

Step 1. Log on as `root`.

Step 2. Install a DB2 Administration Client or any server product, plus one or more language versions of the DB2 Product Library, if they are not already installed.

Step 3. Go into the directory `/opt/IMNSearch/bin`.

Step 4. Edit the file `S990IMNSearch`:

a. On line 46, you will see a **nohup** command that executes the **httpdlite** command with a filename argument. Make a copy of this line immediately below the original line.

b. In the new line, change the part of the filename that reads `httpdlite.conf` to `db2httpd.conf`.

c. Save the file and exit from the editor.

Step 5. Go into the directory `/etc/IMNSearch/httpdlite`.

Step 6. Copy the file `httpdlite.conf` to `db2httpd.conf`.

Step 7. Edit the new file `db2httpd.conf`:

a. Uncomment the line with the `HostName` directive by removing the # character.

b. Change the hostname to the real machine name, such as *server.com*.

c. In the line with the `Port` directive, change the number from `49213` to `49214`.

d. Add the following line:

```
Allow <network_ip> <network_mask>
```

where:

- `<network_ip>` is the network IP address of the machine that will be connecting to your machine, in `a.b.c.d` form. A value of *0.0.0.0* is not allowed.
- `<network_mask>` is the network mask of the machine that will be connecting to your machine,, either in `a.b.c.d` or `x` form, where `x` is the number of valid bits. For example, a value of *24* for `x` would be *255.255.255.0* in `a.b.c.d` form. The only valid network masks are those consisting of a consecutive sequence of 1 bits followed by an optional consecutive sequence of 0 bits.

Note: You can have more than one `Allow` directive, one for each machine that will be connecting to your machine.

e. Save the file and exit from the editor.

Step 8. Run the command **/opt/IMNSearch/bin/S990IMNSearch stop**

Step 9. Run the command **/opt/IMNSearch/bin/S990IMNSearch start**

Step 10. Go into the directory `/opt/IBMdb2/V6.1/doc/en_US/html`

Step 11. Edit the file `db2srch.htm`:

a. Change the string localhost to reflect the name of your host, and change the number `49213` to `49214`.

For example, if the original line is:

```
<form action="http://localhost:49213/cgi-bin/db2srsen.exe" method="POST">
```

The modified line might be:

```
<form action="http://server.com:49214/cgi-bin/db2srsen.exe" method="POST">
```

b. Save the file and exit from the editor.

Step 12. Issue this set of commands, substituting your own hostname for `server.com`:

```
/opt/IMNSearch/cli/imndomap -u DB2SR6EN "http://server.com/doc/en_US/html/" DB2SR6EN
/opt/IMNSearch/cli/imndomap -u DB2CC6EN "http://server.com/doc/en_US/html/" DB2CC6EN
```

For documentation in other languages, replace the `EN` suffix in each of the index names with the appropriate two-letter locale, and replace `en_US` with the appropriate locale directory. For the Japanese, Korean, Simplified Chinese, and Traditional Chinese locales, use the **imqdomap** command instead the **imndomap** command.

Note: The exact URL to enter in the imndomap commands and to use for the search form depends on how you set up your web server to work with the DB2 Product Library. If you make a directory on the web server named **db2** or something that points to `/opt/IBMdb2/V6.1`, then you would insert that directory name after the hostname in each case.

Step 13. You should now be able to start a browser on any machine in the local network, load the `doc/en_US/html/db2srch.htm` file underneath your web server, execute a search, and click on a link in the search results page to go to the documentation file.

Typical Web Server Scenarios

Here are some ways that you might put the UDB 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:\sqllib\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*

manual. Assume that your company does not do any application development, so you do not install the Personal Developer's Edition, which includes the programming books such as the *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:\sqllib\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:\sqllib\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:\sqllib\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 2.02 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

Assuming that your company does not do any application development, you would not get the DB2 Universal Database Application Developer's Kit, and thus would not get the programming books such as the *API Reference*, *Call Level Interface Guide and Reference*, or the *Application Development Guide*. (Note that you do get these programming books as part of the database server documentation for UNIX systems.)

On this system, DB2 is installed on the `E:` drive, so the HTML files are all stored under `e:\sqllib\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:\sqllib\doc\html\index.htm`, the central list of DB2 books. To make the DB2 documentation available you need to perform the following steps:

Step 1. Run the **Administer Netscape Servers** program from the Netscape folder, and choose `udbserv` as the server.

Step 2. Under **Content Mgmt**, change the **Primary Document Directory** to `e:\sqllib\doc\html`.

Step 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:\sqllib\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 Online Documentation</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:\sqllib\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:\sqllib\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 your personal Windows NT 4.0 machine as the DB2 server, and decide to use the Peer Web Services to make the DB2 online documentation available to the rest of your workgroup. The steps are similar whether you use Peer Web Services or Internet Information Server.

Install the DB2 Universal Database Documentation
Install the DB2 documentation into the same directory as in the previous scenarios: `e:\sqllib\doc\html`.

Install the Web Server
If Peer Web Services is not installed at the same time as the Windows NT system, you run the program **c:\winnt\system32\inetins.exe** to install the code from the Windows NT installation CD-ROM. You use the **Services** dialog in the **Control Panel** to make this service start automatically.

Make the DB2 Documentation Available through the Web Server
You run the **Internet Service Manager** from the Microsoft Peer Web Services folder of the **Start** menu, going to the **WWW** section and then the **Directories** subsection. Initially, you edit the home directory entry, changing the path from `c:\inetpub\wwwroot` to `e:\sqllib\doc\html` and the default document to `index.htm`. Later, when you need to serve other documents as well as the DB2 books, you 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 Peer Web Services, you must replace the file `e:\sqllib\doc\html\db2srch.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 Peer Web Services, you need to

copy the DB2 HTML files to the c:\inetpub\wwwroot directory tree. You create a directory named db2docs, and use the **xcopy /s** command to preserve the directory structure during the copy. Now you 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 ″product library″, that is, the HTML books. 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** command 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_06_01/doc/html
- On HP-UX and Solaris systems: /opt/IBMdb2/V6.1/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** command, you must follow the instructions in the *Quick Beginnings* book 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_06_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_06_01/doc/html would point to the French versions of files that are translated into French, and English files for everything else.

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 (indeed, all non-English 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 a DB2 UNIX server is the simplest starting point, because it comes with more books than the OS/2 and Windows NT servers do. For example, it includes some programming books that on OS/2 and Windows NT only come with the Application Developer's Kit.

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.

DB2 Extender Documentation

The DB2 Extenders, which come with the Application Developer's Kit, come with online documentation of their own. If you installed using the default directory name dmb, the tables of contents for the books are in:

```
\dmb\doc\html\desu9\desu9mst.htm
\dmb\doc\html\dmba5\dmba5mst.htm
```

You can use the same techniques of copying or creating virtual directories on the web server to make the files in these directories available to all users.

Appendix E. How the DB2 Library Is Structured

The DB2 Universal Database library consists of SmartGuides, online help, books and sample programs in HTML format. This section describes the information that is provided, and how to access it.

To access product information online, you can use the Information Center. You can view task information, DB2 books, troubleshooting information, sample programs, and DB2 information on the Web. See "Accessing Information with the Information Center" on page 536 for details.

Completing Tasks with SmartGuides

SmartGuides help you complete some administration tasks by taking you through each task one step at a time. SmartGuides are available through the Control Center and the Client Configuration Assistant. The following table lists the SmartGuides.

Note: Create Database, Index, and Configure Multisite Update SmartGuide are available for the partitioned database environment.

SmartGuide	Helps You to...	How to Access...
Add Database	Catalog a database on a client workstation.	From the Client Configuration Assistant, click **Add**.
Back up Database	Determine, create, and schedule a backup plan.	From the Control Center, click with the right mouse button on the database you want to back up and select **Backup**->**Database using SmartGuide**.
Configure Multisite Update SmartGuide	Perform a multi-site update, a distributed transaction, or a two-phase commit.	From the Control Center, click with the right mouse button on the **Database** icon and select **Multisite Update**.
Create Database	Create a database, and perform some basic configuration tasks.	From the Control Center, click with the right mouse button on the **Databases** icon and select **Create**->**Database using SmartGuide**.

© Copyright IBM Corp. 1993, 1999

SmartGuide	Helps You to...	How to Access...
Create Table	Select basic data types, and create a primary key for the table.	From the Control Center, click with the right mouse button on the **Tables** icon and select **Create**->**Table using SmartGuide**.
Create Table Space	Create a new table space.	From the Control Center, click with the right mouse button on the **Table spaces** icon and select **Create**->**Table space using SmartGuide**.
Index	Advise which indexes to create and drop for all your queries.	From the Control Center, click with the right mouse button on the **Index** icon and select **Create**->**Index using SmartGuide**.
Performance Configuration	Tune the performance of a database by updating configuration parameters to match your business requirements.	From the Control Center, click with the right mouse button on the database you want to tune and select **Configure using SmartGuide**.
Restore Database	Recover a database after a failure. It helps you understand which backup to use, and which logs to replay.	From the Control Center, click with the right mouse button on the database you want to restore and select **Restore**->**Database using SmartGuide**.

Accessing Online Help

Online help is available with all DB2 components. The following table describes the various types of help. You can also access DB2 information through the Information Center. For information see "Accessing Information with the Information Center" on page 536.

Type of Help	Contents	How to Access...
Command Help	Explains the syntax of commands in the command line processor.	From the command line processor in interactive mode, enter: ? *command* where *command* is a keyword or the entire command. For example, `? catalog` displays help for all the CATALOG commands, while `? catalog database` displays help for the CATALOG DATABASE command.

Type of Help	Contents	How to Access...
Control Center Help *Client Configuration Assistant Help* *Event Analyzer Help* *Command Center Help*	Explains the tasks you can perform in a window or notebook. The help includes prerequisite information you need to know, and describes how to use the window or notebook controls.	From a window or notebook, click the **Help** push button or press the F1 key.

Message Help	Describes the cause of a message, and any action you should take.	From the command line processor in interactive mode, enter:
		
		`? XXXnnnnn`
		
		where *XXXnnnnn* is a valid message identifier.
		
		For example, `? SQL30081` displays help about the SQL30081 message.
		
		To view message help one screen at a time, enter:
		
		`? XXXnnnnn | more`
		
		To save message help in a file, enter:
		
		`? XXXnnnnn > filename.ext`
		
		where *filename.ext* is the file where you want to save the message help.

SQL Help	Explains the syntax of SQL statements.	From the command line processor in interactive mode, enter: `help statement` where *statement* is an SQL statement. For example, **help** SELECT displays help about the SELECT statement. **Note:** SQL help is not available on UNIX-based platforms.
SQLSTATE Help	Explains SQL states and class codes.	From the command line processor in interactive mode, enter: **?** *sqlstate* or **?** *class-code* where *sqlstate* is a valid five-digit SQL state and *class-code* is the first two digits of the SQL state. For example, `? 08003` displays help for the 08003 SQL state, while `? 08` displays help for the 08 class code.

DB2 Information – Hardcopy and Online

The table in this section lists the DB2 books. They are divided into two groups:

Cross-platform books
These books contain the common DB2 information for all platforms.

Platform-specific books
These books are for DB2 on a specific platform. For example, there are separate *Quick Beginnings* books for DB2 on OS/2, on Windows NT, and on the UNIX-based platforms.

Cross-platform sample programs in HTML
These samples are the HTML version of the sample programs that are installed with the SDK. They are for informational purposes and do not replace the actual programs.

Most books are available in HTML and PostScript format, or you can choose to order a hardcopy from IBM. The exceptions are noted in the table.

On OS/2 and Windows platforms, HTML documentation files can be installed under the doc\html subdirectory. Depending on the language of your system, some files may be in that language, and the remainder are in English.

On UNIX platforms, you can install multiple language versions of the HTML documentation files under the doc/%L/html subdirectories. Any documentation that is not available in a national language is shown in English.

You can obtain DB2 books and access information in a variety of different ways:

View See "Viewing Online Information" on page 535.

Search See "Searching Online Information" on page 538.

Print See "Printing the PostScript Books" on page 538.

Order See "Ordering the Printed Books" on page 539.

Name	Description	Form Number File Name for Online Book	HTML Directory
	Cross-Platform Books		

Name	Description	Form Number File Name for Online Book	HTML Directory
Administration Guide	*Administration Guide, Design and Implementation* contains information required to design, implement, and maintain a database. It also describes database access using the Control Center(whether local or in a client/server environment), auditing, database recovery, distributed database support, and high availability. *Administration Guide, Performance* contains information that focuses on the database environment, such as application performance evaluation and tuning. You can order both volumes of the *Administration Guide* in the English language in North America using the form number SBOF-8922.	Volume 1 SC09-2839 db2d1x60 Volume 2 SC09-2840 db2d2x60	db2d0
Administrative API Reference	Describes the DB2 application programming interfaces (APIs) and data structures you can use to manage your databases. Explains how to call APIs from your applications.	SC09-2841 db2b0x60	db2b0
Application Building Guide	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. This book combines the *Building Applications* books for the OS/2, Windows, and UNIX-based environments.	SC09-2842 db2axx60	db2ax
APPC, CPI-C and SNA Sense Codes	Provides general information about APPC, CPI-C, and SNA sense codes that you may encounter when using DB2 Universal Database products. **Note:** Available in HTML format only.	No form number db2apx60	db2ap

Name	Description	Form Number File Name for Online Book	HTML Directory
Application Development Guide	Explains how to develop applications that access DB2 databases using embedded SQL or JDBC, how to write stored procedures, user-defined types, user-defined functions, and how to use triggers. It also discusses programming techniques and performance considerations. This book was formerly known as the *Embedded SQL Programming Guide*.	SC09-2845 db2a0x60	db2a0
CLI Guide and Reference	Explains how to develop applications that access DB2 databases using the DB2 Call Level Interface, a callable SQL interface that is compatible with the Microsoft ODBC specification.	SC09-2843 db2l0x60	db2l0
Command Reference	Explains how to use the command line processor, and describes the DB2 commands you can use to manage your database.	SC09-2844 db2n0x60	db2n0
Data Movement Utilities Guide and Reference	Explains how to use the Load, Import, Export, Autoloader, and Data Propogation utilities to work with the data in the database.	SC09-2858 db2dmx60	db2dm
DB2 Connect Personal Edition Quick Beginnings	Provides planning, installing, and configuring information for DB2 Connect Personal Edition.	GC09-2830 db2c1x60	db2c1
DB2 Connect User's Guide	Provides concepts, programming and general usage information about the DB2 Connect products.	SC09-2838 db2c0x60	db2c0
Connectivity Supplement	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, and on how to use DRDA application servers with DB2 Connect application requesters. **Note:** Available in HTML and PostScript formats only.	No form number db2h1x60	db2h1
Glossary	Provides a comprehensive list of all DB2 terms and definitions. **Note:** Available in HTML format only.	No form number db2t0x50	db2t0

Name	Description	Form Number File Name for Online Book	HTML Directory
Installation and Configuration Supplement	Guides you through the planning, installation, and set up of platform-specific DB2 clients. This supplement contains information on binding, setting up client and server communications, DB2 GUI tools, DRDA AS, distributed installation, and the configuration of distributed requests and access methods to heterogeneous data sources.	GC09-2857 db2iyx60	db2iy
Message Reference	Lists messages and codes issued by DB2, and describes the actions you should take.	GC09-2846 db2m0x60	db2m0
Replication Guide and Reference	Provides planning, configuration, administration, and usage information for the IBM Replication tools supplied with DB2.	SC26-9642 db2e0x60	db2e0
SQL Getting Started	Introduces SQL concepts, and provides examples for many constructs and tasks.	SC09-2856 db2y0x60	db2y0
SQL Reference, Volume 1 and Volume 2	Describes SQL syntax, semantics, and the rules of the language. Also includes information about release-to-release incompatibilities, product limits, and catalog views. You can order both volumes of the *SQL Reference* in the English language in North America with the form number SBOF-8923.	SBOF-8923 Volume 1 db2s1x60 Volume 2 db2s2x60	db2s0
System Monitor Guide and Reference	Describes how to collect different kinds of information about databases and the database manager. Explains how to use the information to understand database activity, improve performance, and determine the cause of problems.	SC09-2849 db2f0x60	db2f0
Troubleshooting Guide	Helps you determine the source of errors, recover from problems, and use diagnostic tools in consultation with DB2 Customer Service.	S10J-8169	db2p0

Name	Description	Form Number File Name for Online Book	HTML Directory
What's New	Describes the new features, functions, and enhancements in DB2 Universal Database, Version 6.0, including information about Java-based tools.	SC09-2851 db2q0x60	db2q0
	Platform-Specific Books		
Administering Satellites Guide and Reference	Provides planning, configuration, administration, and usage information for satellites.	GC09-2821 db2dsx60	db2ds
DB2 Personal Edition Quick Beginnings	Provides planning, installation, migration, and configuration information for DB2 Universal Database Personal Edition on the OS/2, Windows 95, and Windows NT operating systems.	GC09-2831 db2i1x60	db2i1
DB2 for OS/2 Quick Beginnings	Provides planning, installation, migration, and configuration information for DB2 Universal Database on the OS/2 operating system. Also contains installing and setup information for many supported clients.	GC09-2834 db2i2x60	db2i2
DB2 for UNIX Quick Beginnings	Provides planning, installation, migration, and configuration information for DB2 Universal Database on UNIX-based platforms. Also contains installing and setup information for many supported clients.	GC09-2836 db2ixx60	db2ix
DB2 for Windows NT Quick Beginnings	Provides planning, installation, migration, and configuration information for DB2 Universal Database on the Windows NT operating system. Also contains installing and setup information for many supported clients.	GC09-2835 db2i6x60	db2i6
DB2 Enterprise - Extended Edition for UNIX Quick Beginnings	Provides planning, installation, and configuration information for DB2 Enterprise - Extended Edition for UNIX. Also contains installing and setup information for many supported clients.	GC09-2832 db2v3x60	db2v3

Name	Description	Form Number File Name for Online Book	HTML Directory
DB2 Enterprise - Extended Edition for Windows NT Quick Beginnings	Provides planning, installation, and configuration information for DB2 Enterprise - Extended Edition for Windows NT. Also contains installing and setup information for many supported clients.	GC09-2833 db2v6x60	db2v6
DB2 Connect Enterprise Edition for OS/2 and Windows NT Quick Beginnings	Provides planning, migration, installation, and configuration information for DB2 Connect Enterprise Edition on the OS/2 and Windows NT operating systems. Also contains installation and setup information for many supported clients. This book was formerly part of the *DB2 Connect Enterprise Edition Quick Beginnings.*	GC09-2828 db2c6x60	db2c6
DB2 Connect Enterprise Edition for UNIX Quick Beginnings	Provides planning, migration, installation, configuration, and usage information for DB2 Connect Enterprise Edition in UNIX-based platforms. Also contains installation and setup information for many supported clients. This book was formerly part of the *DB2 Connect Enterprise Edition Quick Beginnings.*	GC09-2829 db2cyx60	db2cy
DB2 Data Links Manager for AIX Quick Beginnings	Provides planning, installation, configuration, and task information for DB2 Data Links Manager for AIX.	GC09-2837 db2z0x60	db2z0
DB2 Data Links Manager for Windows NT Quick Beginnings	Provides planning, installation, configuration, and task information for DB2 Data Links Manager for Windows NT.	GC09-2827 db2z6x60	db2z6
DB2 Query Patroller Administration Guide	Provides administration information on DB2 Query Patrol.	SC09-2859 db2dwx60	db2dw
DB2 Query Patroller Installation Guide	Provides installation information on DB2 Query Patrol.	GC09-2860 db2iwx60	db2iw
DB2 Query Patroller User's Guide	Describes how to use the tools and functions of the DB2 Query Patrol.	SC09-2861 db2wwx60	db2ww

Name	Description	Form Number File Name for Online Book	HTML Directory
	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 for informational purposes (not all samples are available in all languages). Only available when the SDK is installed. See *Application Building Guide* for more information on the actual programs. **Note:** Available in HTML format only.	No form number	db2hs/c db2hs/cli db2hs/clp db2hs/cpp db2hs/cobol db2hs/cobol_mf db2hs/fortran db2hs/java db2hs/rexx

Notes:

1. The character in the sixth position of the file name indicates the language of a book. For example, the file name `db2d0e60` indicates that the *Administration Guide* is in English. The following letters are used in the file names to indicate the language of a book:

Language	Identifier
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. For late breaking information that could not be included in the DB2 books:
   - On UNIX-based platforms, see the `Release.Notes` file. This file is located in the `DB2DIR/Readme/%L` directory, where `%L` is the locale name and DB2DIR is:
     - `/usr/lpp/db2_06_01` on AIX
     - `/opt/IBMdb2/V6.1` on HP-UX, Solaris, SCO UnixWare 7, and Silicon Graphics IRIX
     - `/usr/IBMdb2/V6.1` on Linux.
   - On other platforms, see the `RELEASE.TXT` file. This file is located in the directory where the product is installed.
   - Under Windows `Start` menu

Viewing Online Information

The manuals included with this product are in Hypertext Markup Language (HTML) softcopy format. Softcopy format enables you to search or browse the information, and provides hypertext links to related information. It also makes it easier to share the library across your site.

You can 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 on all platforms other than SCO UnixWare 7:

- If you are running DB2 administration tools, use the Information Center. See "Accessing Information with the Information Center" on page 536 for details.
- Select the `Open Page` menu item of your Web browser. The page you open contains descriptions of and links to DB2 information:
  - On UNIX-based platforms, open the following page:

    `file:/`*`INSTHOME`*`/sqllib/doc/`*`%L`*`/html/index.htm`

    where *%L* is the locale name.
  - On other platforms, open the following page:

    `sqllib\doc\html\index.htm`

    The path is located on the drive where DB2 is installed.

If you have not installed the Information Center, you can open the page by double-clicking on the **DB2 Online Books** icon. Depending on the system you are using, the icon is in the main product folder or the Windows Start menu.

To view online books or sample programs on the SCO UnixWare 7:

- DB2 Universal Database for SCO UnixWare 7 uses the native SCOhelp utility to search the DB2 information. You can access SCOhelp by the following methods:
  - entering the ″scohelp″ command on the command line,
  - selecting the `Help` menu in the Control Panel of the CDE desktop or
  - selecting `Help` in the Root menu of the Panorama desktop

  For more information on SCOhelp, refer to the *Installation and Configuration Supplement*.

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.

Depending on your system, you can access the Information Center from the:

- Main product folder
- Toolbar in the Control Center
- Windows Start menu
- Help menu of the Control Center

The Information Center provides the following kinds of information. Click the appropriate tab to look at the information:

Tasks	Lists tasks you can perform using DB2.
Reference	Lists DB2 reference information, such as keywords, commands, and APIs.
Books	Lists DB2 books.
Troubleshooting	Lists categories of error messages and their recovery actions.
Sample Programs	Lists sample programs that come with the DB2 Software Developer's Kit. If the Software Developer's Kit is not installed, this tab is not displayed.
Web	Lists DB2 information on the World Wide

Web. To access this information, you must have a connection to the Web from your system.

When you select an item in one of the lists, the Information Center launches a viewer to display the information. The viewer might be the system help viewer, an editor, or a Web browser, depending on the kind of information you select.

The Information Center provides some search capabilities, so you can look for specific topics, and filter capabilities to limit the scope of your searches.

For a full text search, click the Search button of the Information Center follow the *Search DB2 Books* link in each HTML file.

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 by double-clicking its icon on the Windows or OS/2 desktop.

Refer to the release notes if you experience any other problems when searching the HTML information.

Note: Search function is not available in the Linux and Silicon Graphics environments.

Setting Up a Document Server

By default, the DB2 information is installed on your local system. This means that each person who needs access to the DB2 information must install the same files. To have the DB2 information stored in a single location, use the following instructions:

1. Copy all files and subdirectories from \sqllib\doc\html on your local system to a Web server. Each book has its own subdirectory containing all the necessary HTML and GIF files that make up the book. Ensure that the directory structure remains the same.
2. Configure the Web server to look for the files in the new location. For information, see the NetQuestion Appendix in *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. Once you are able to view the book files, you should bookmark commonly viewed topics. Among those, 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 setting up a search, see the NetQuestion Appendix in *Installation and Configuration Supplement* book.

Searching Online Information

To search for information in the HTML books, you can do the following:

- Click on **Search the DB2 Books** at the bottom of any page in the HTML books. Use the search form to find a specific topic. This function is not available in the Linux or Silicon Graphics IRIX environments.
- Click on **Index** at the bottom of any page in an HTML book. Use the index to find a specific topic in the book.
- Display the table of contents or index of the HTML book, and then use the find function of the Web browser to find a specific topic in the book.
- Use the bookmark function of the Web browser to quickly return to a specific topic.
- Use the search function of the Information Center to find specific topics. See "Accessing Information with the Information Center" on page 536 for details.

Printing the PostScript Books

If you prefer to have printed copies of the manuals, you can decompress and print PostScript versions. For the file name of each book in the library, see the table in "DB2 Information – Hardcopy and Online" on page 528. Specify the full path name for the file you intend to print.

On OS/2 and Windows platforms:

1. Copy the compressed PostScript files to a hard drive on your system. The files have a file extension of `.exe` and are located in the `x:\doc\`*`language`*`\books\ps` directory, where x: is the letter representing the CD-ROM drive and *language* is the two-character country code that represents your language (for example, EN for English).
2. Decompress the file that corresponds to the book that you want. Each compressed book is a self-extracting executable file. To decompress the

book, simply run it as you would run any other executable program. The result from this step is a printable PostScript file with a file extension of .ps.

3. Ensure that your default printer is a PostScript printer capable of printing Level 1 (or equivalent) files.
4. Enter the following command from a command line:

   ```
   print filename.ps
   ```

On UNIX-based platforms:

1. Mount the CD-ROM. Refer to your *Quick Beginnings* manual for the procedures to mount the CD-ROM.
2. Change to /cdrom/doc/%L/ps directory on the CD-ROM, where */cdrom* is the mount point of the CD-ROM and *%L* is the name of the desired locale. The manuals will be installed in the previously-mentioned directory with file names ending with .ps.Z.
3. Decompress and print the manual you require using the following command:
   - For AIX:

     ```
     zcat filename | qprt -P PSPrinter_queue
     ```

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

     ```
     zcat filename | lp -d PSPrinter_queue
     ```

   - For Linux:

     ```
     zcat filename | lpr -P PSPrinter_queue
     ```

   - For Silicon Graphics IRIX:

     ```
     zcat < filename | lp -d PSPrinter_queue
     ```

   where *filename* is the full path name and extension of the compressed PostScript file and *PSprinter_queue* is the name of the PostScript printer queue.

   For example, to print the English version of *DB2 for UNIX Quick Beginnings* on AIX, you can use the following command:

   ```
   zcat /cdrom/doc/en/ps/db2ixe60.ps.Z || qprt -P ps1
   ```

Ordering the Printed Books

You can order the printed DB2 manuals either as a set or individually. There are three sets of books available. The form number for the entire set of DB2 books is SBOF-8926-00. The form number for the books listed under the heading ″Cross-Platform Books″ is SBOF-8924-00.

Note: These form numbers only apply if you are ordering books that are printed in the English language in North America.

You can also order books individually by the form number listed in “DB2 Information – Hardcopy and Online” on page 528. To order printed versions, contact your IBM authorized dealer or marketing representative, or phone 1-800-879-2755 in the United States or 1-800-IBM-4YOU in Canada.

Appendix F. 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, do the following:

1. Ensure that the appropriate message option for the desired language has been installed.
2. Set the *LANG* environment variable to the desired locale.

   For example, to use `fr_FR` messages on DB2 for AIX, you must have the `fr_FR` message option installed and must set *LANG* to `fr_FR`.

The selected message catalog filesets are placed in the following directories on the target workstation:

DB2 for AIX
`/usr/lpp/db2_06_01/msg/%L`

DB2 for HP-UX
`/opt/IBMdb2/V6.1/msg/%L`

DB2 for Linux
`/usr/IBMdb2/V6.1/msg/%L`

DB2 for Solaris
`/opt/IBMdb2/V6.1/msg/%L`

where *%L* is equal to the locale name of the message catalog.

© Copyright IBM Corp. 1993, 1999

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 36 shows the languages into which the DB2 messages are translated.

Note: The code page values in the table that follows are also used as directory names on DB2 CD-ROMs. For example, a reference to *x*:*language*\win32\install would be *x*:\en\win32\install for English. For more detailed information on the languages and code pages support, refer to the *Administration Guide*.

Table 36. 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

Table 36. Languages and Code Pages (continued)

Country Code	Language
nl	Dutch
no	Norwegian
pl	Polish
pt	Portuguese
ru	Russian
se	Swedish
si	Slovenian
tr	Turkish
tw	Traditional Chinese (Taiwan)

Appendix G. 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" on page 546
- "Object Name Rules" on page 546
- "Username, User ID, Group Name, and Instance Name Rules" on page 547
- "Password Rules" on page 548
- "DB2SYSTEM Naming Rules" on page 548
- "Workstation Name (nname) Rules" on page 548

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*.

© Copyright IBM Corp. 1993, 1999

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.

When naming a database or database alias, see "General Naming Rules" on page 545.

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 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 545.

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; and
  - column 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 545.

In addition, the name you specify:

- Can contain 1 to 8 characters
- Cannot be any of the following:
  - USERS
  - ADMINS
  - GUESTS
  - PUBLIC
  - LOCAL
- Cannot begin with:
  - IBM
  - SQL
  - SYS
- Cannot include accented characters
- In general, when naming users, groups, or instances:

  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 545.

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 NT Computer name and assigns it to *DB2SYSTEM*.

When creating a *DB2SYSTEM* name, see "General Naming Rules" on page 545.

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 9x or Windows NT
A maximum of 14 characters.

Appendix H. Notices

Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent product, program or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the

IBM Director of Licensing
IBM Corporation, North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Canada Limited
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.

This publication may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

© Copyright IBM Corp. 1993, 1999

Trademarks

The following terms are trademarks or registered trademarks of the IBM Corporation in the United States and/or other countries:

ACF/VTAM
ADSTAR
AISPO
AIX
AIXwindows
AnyNet
APPN
AS/400
CICS
C Set++
C/370
DATABASE 2
DataHub
DataJoiner
DataPropagator
DataRefresher
DB2
DB2 Connect
DB2 Universal Database
Distributed Relational Database Architecture
DRDA
Extended Services
FFST
First Failure Support Technology
IBM
IMS
LAN Distance
MVS/ESA
MVS/XA
OS/400
OS/390
OS/2
PowerPC
QMF
RACF
RISC System/6000
SP
SQL/DS
SQL/400
S/370
System/370
System/390
SystemView
VisualAge
VM/ESA
VSE/ESA
VTAM
WIN-OS/2

Trademarks of Other Companies

The following terms are trademarks or registered trademarks of the companies listed:

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

HP-UX is a trademark of Hewlett-Packard.

Java, HotJava, Solaris, Solstice, and Sun are trademarks of Sun Microsystems, Inc.

Microsoft, Windows, Windows NT, Visual Basic, and the Windows logo are trademarks or registered trademarks of Microsoft Corporation in the United States, other countries, or both.

PC Direct is a trademark of Ziff Communications Company in the United States, other countries, or both and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium, and ProShare are trademarks of Intel Corporation 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 180
  ODBC 180

A

Access Profile
  creating 39
  using 38
accessing DB2 servers
  configuring communications
    IBM eNetwork Communication Server V5 for AIX 93
    IBM Personal Communications for Windows 32-Bit Operating Systems 78
    SNAplus2 for HP-UX 101
    SunLink SNA PU 2.1 for Solaris 105
accessing host servers
  configuring communications
    IBM eNetwork Communication Server V5 for AIX 295
    IBM Personal Communications for Windows 32-Bit Operating Systems 89, 263, 283
    SNA API Client 87, 280
    SNAplus2 for HP-UX 311
    SunLink SNA PU 2.1 for Solaris 324
accessing multiple servers 43
accessing servers
  IPX/SPX 62
  Named Pipes 43
  NetBIOS 43, 56
  overview 43
  TCP/IP 48, 235
Administering Satellites Guide and Reference 532
Administration Guide 528
Administrative API Reference 529
admintool command 394, 406, 414
APPC
  APPCLLU registry value 348
  server 135
APPC *(continued)*
  setting up communications for SPM 348
  software requirements 8, 9, 10
  support on AIX
    Bull SNA 98, 308
  support on HP-UX
    SNAplusLink 78, 87, 89, 93, 101, 105, 263, 280, 283, 295, 311, 324
  support on OS/2
    Communications Manager for OS/2 8, 73, 149, 251
    Communications Server for OS/2 8, 73, 149, 251, 347
  support on Solaris
    SunLink SNA 9
  support on Windows NT
    Communications Server for Windows NT SNA Client 87, 281
    Microsoft SNA Client 92
  verifying the connection 112
APPC, CPI-C and SNA Sense Codes 529
APPCLLU registry value 348
APPENDAPINAME, keyword 190
APPL 249
Application Building Guide 529
application development
  using ODBC 171
Application Development Guide 529
application requester
  steps to configure SPM 343
application server
  configuring 247
AS/400
  configuring 247
ASYNCENABLE, keyword 190
Asynchronous ODBC, enabling 190
authentication
  server 119
authorization IDs
  naming rules 394

B

binding
  utilities and applications 170
BITDATA, keyword 191

C

cataloging
  APPC node 108, 109, 327, 328
  databases 45, 52, 53, 59, 66, 109, 241, 242, 328, 329
  IPX/SPX node 65, 66, 134
  Named Pipe node 44, 45
  NetBIOS node 59, 129
  nodes
    in a federated environment 457
  TCP/IP node 51, 52, 125, 240, 241
CD-ROM
  installing DB2 Software Developer's Kit 19
  installing DB2 Universal Database 19
CLI Guide and Reference 530
Client Configuration Assistant
  administering clients communications 32
  changing privileges 481
  client profiles 32
    exporting 40
    using 39
  searching the network 32
  using access profiles 38
client profile
  exporting 40
clients
  configuring 43
  installing 13
CLISCHEMA, keyword 192
Command Center
  entering DB2 commands 477
  entering SQL statements 477
Command Reference 530
commands
  admintool 394, 406, 414, 415
  db2_install 409
  db2icrt 395, 407, 415
  db2imigr 484, 485
  db2licd 483
  db2rmln 396, 408, 417
  db2set 120
  groupadd 406, 415
  mkgroup 394
  mkuser 394

© Copyright IBM Corp. 1993, 1999

commands *(continued)*
mount 394
required to start SPM 348
rlogin 27
SAM, to add users 406
SAM, to update kernel 405
smit 390, 391
swinstall 400, 402, 403, 404
SYSADM 394, 406, 414
useradd 406
communication protocols
APPC 6, 7, 8, 9, 10, 70, 73, 135, 149, 247, 251
IPX/SPX 6, 7, 8, 9, 10, 62, 63, 130
Named Pipes 8, 9, 10, 43, 121
NetBIOS 8, 9, 10, 56, 125
TCP/IP 6, 7, 8, 9, 10, 48, 122, 235
configuring 48, 235
communications
configuring the client 32, 43, 48, 56, 62, 70, 235, 247
configuring the server 116, 119
managing 43, 119
Client Configuration Assistant 32
Control Center 116
supported protocols
APPC 70, 247
IPX/SPX 130
Named Pipes 43, 121
NetBIOS 56, 125
TCP/IP 48, 62, 122, 235
Communications Server
required for SPM 340
Communications Server for Windows NT SNA Client
configuring manually 87, 281
version required 87, 281
components
on UNIX operating systems 421
selecting 421
configuration 78, 87, 89, 93, 101, 105, 263, 280, 283, 295, 311, 324
configuration parameters
kernel 405
setting DB2 119, 481
SYSADM_GROUP 481
configuring
APPC 70, 135, 247
for OS/2 73, 149, 251
IPX/SPX 62, 63
Named Pipes 43, 121
NetBIOS 56, 125
configuring *(continued)*
ODBC driver 70, 176
steps to configure SNAPlus 94, 101, 105, 295, 311, 324
steps to configure SunLink SNA 108, 327
TCP/IP 48, 122, 235
configuring client communications
setting configuration parameters 43
using the Client Configuration Assistant 32
using the command line processor 43
configuring communications
overview 43
using Client Configuration Assistant 32
using Control Center 116
configuring server communications
setting DB2COMM 39, 120
using the command line processor 119
using the Control Center 116
configuring servers 116, 119
Connectivity Supplement 530
CONNECTNODE, keyword 193
CONNECTTYPE, keyword 193
Control Point LU
cannot be the SPM LU 346, 348, 349
control point name 72, 136, 250
CREATE NICKNAME statement
for DB2 family data sources 460
for Oracle data sources 469
CREATE SERVER statement
for DB2 family data sources 459
for Oracle data sources 467
CREATE USER MAPPING statement
for DB2 family data sources 459
for Oracle data sources 468
CREATE WRAPPER statement
for DB2 family data sources 458
for Oracle data sources 466
creating links for the DB2 libraries, installation 408
CURRENTFUNCTIONPATH, keyword 194
CURRENTPACKAGESET, keyword 195
CURRENTREFRESHAGE, keyword 196
CURRENTSCHEMA, keyword 196
CURRENTSQLID, keyword 197
CURSORHOLD, keyword 197

D

Data Movement Utilities Guide and Reference 530
database
naming rules 546
database alias 241, 242, 328, 329
naming rules 546
database objects
naming rules 546
DB2 client
installing 13
Windows 9x 15
Windows NT 15
installing on OS/2 19
software requirements 6
DB2 Connect
configuring on OS/2 73, 149, 251
DB2 Connect Enterprise Edition
configuring to accept APPC clients 138
DB2 Connect Enterprise Edition for OS/2 and Windows NT Quick Beginnings 533
DB2 Connect Enterprise Edition for UNIX Quick Beginnings 533
DB2 Connect Personal Edition Quick Beginnings 530
DB2 Connect User's Guide 530
DB2 Data Links Manager for AIX Quick Beginnings 533
DB2 Data Links Manager for Windows NT Quick Beginnings 533
DB2_DJ_COMM environment variable
setting for DB2 family data sources 459
setting for Oracle data sources 467
DB2 Enterprise - Extended Edition
disk requirements 3
DB2 Enterprise - Extended Edition for UNIX Quick Beginnings 532
DB2 Enterprise - Extended Edition for Windows NT Quick Beginnings 532
DB2 family data sources in a federated system
cataloging nodes 457
creating nicknames 460
creating server mappings 459
creating user mappings 459
creating wrappers 458
setting DB2_DJ_COMM 459

DB2 family data sources in a federated system *(continued)*
verifying connections 457
DB2 libraries
creating links 396, 408, 416
DB2 library
books 528
Information Center 536
language identifier for books 534
late-breaking information 535
online help 526
ordering printed books 539
printing PostScript books 538
searching online information 538
setting up document server 537
SmartGuides 525
structure of 525
viewing online information 535
DB2 Personal Edition Quick Beginnings 532
DB2 Query Patroller Administration Guide 533
DB2 Query Patroller Installation Guide 533
DB2 Query Patroller User's Guide 533
db2 terminate command 483
DB2 Universal Database
software requirements 6
DB2 Workgroup Edition
disk requirements 3
db2cli.ini 180
DB2COMM 120
DB2CONNECTVERSION, keyword 198
DB2DEGREE, keyword 199
DB2DJ.ini file 464
db2djlink.sh script 464
DB2ESTIMATE, keyword 199
DB2EXPLAIN, keyword 200
db2icrt command 395, 407, 415
db2imigr command 485
DB2NBADAPTERS 126
DB2NODE 193
DB2OPTIMIZATION, keyword 201
db2rmln command 396, 408, 417
db2set command
using 120
DBALIAS, keyword 202
DBNAME, keyword 203
DEFAULTPROCLIBRARY, keyword 204
DEFERREDPREPARE, keyword 204
DISABLEMULTITHREAD, keyword 205
disk requirements
client 3
server 3
Distributed Computing Environment
software requirements 6, 7, 8, 9
DRDA
listing indoubt transactions 350
resolving indoubt transactions 350
DRDA server
configuring 247
DRDA1 Application Server
running with DRDA2 348
specifying a non-SPM LU 347
SPM must use the default LU name 348

E

EARLYCLOSE, keyword 206
environment variables
db2profile script 21
setting for DB2 family data sources 459
setting for Oracle data sources 464

F

federated database system
data sources, supported 455
DB2 family data sources
cataloging nodes 457
creating nicknames 460
creating server mappings 459
creating user mappings 459
creating wrappers 458
setting DB2_DJ_COMM 459
installation considerations 457, 463
Oracle data sources
creating nicknames 469
creating server mappings 467
creating user mappings 468
creating wrappers 466
setting DB2_DJ_COMM 467
overview 453
files
restoring 399
filesets
choosing on AIX 390, 419
choosing on HP-UX 419
choosing on Solaris 419
fixed disks
hardware requirements 3

G

Glossary 530
GRANTEELIST, keyword 207
GRANTORLIST, keyword 207
GRAPHIC, keyword 208
groups
creating 406, 414

H

hardware requirements
fixed disk 3

I

IGNOREWARNINGS, keyword 209
IGNOREWARNLIST, keyword 210
IN DATABASE command 203
indoubt transactions
listing 350
resolving 350
initialization file, ODBC 180
installation
CID using SystemView LAN 17, 20
clients
AIX 21, 389, 399
HP-UX 21
OS/2 19
Solaris 21
Windows 9x 15
Windows NT 15
creating links for DB2 libraries 396
DB2 server on UNIX systems
assigning groups and users 393, 406, 414
creating an instance 395
creating groups and users 393, 406, 414
creating links 396, 408, 416
DB2 Product Library 392, 401, 411
DB2 Product Messages 393, 403, 412
restoring product files 399
updating kernel components 21, 22, 23, 405, 413, 414
disk requirements
client 3
server 3
distributed join option 457, 463
errors 17, 20

installation *(continued)*
federated database system considerations 17, 463
log 17, 20
memory requirements
client 3
server 3
mounting the CD-ROM 23
AIX 24, 25
HP-UX 25
Solaris 26
remote 27
servers
AIX 389
HP-UX 399, 409
OS/2 19
unattended
OS/2 425
Windows 3.x 425
UNIX installations
using installp 389, 390
using SMIT 389, 390, 392
Installation and Configuration Supplement 530
instances
naming restrictions 547
IPX/SPX
client 62
configuring 63
server 130
software requirements 8, 9
supported platforms 63, 131
verifying the connection 54, 61, 68, 111, 244

J

Java, running programs 182
JDBC, running programs 182

K

KEEPCONNECT, keyword 210
KEEPSTATEMENT, keyword 211
kernel configuration parameters
updating 21, 405, 413

L

LANG environment variable 541
LOBMAXCOLUMNSIZE, keyword 212
local adapter address 72, 250
local control point name 72, 249
local LU name 72, 136, 250
LOCATION NAME (MVS, OS/390) 250
Logical LAN Adapter Number (Lana Number)
default value 128
LONGDATACOMPAT, keyword 212
LU 72, 136, 250
LU 6.2 Syncpoint Manager
prerequisites 340

M

managing connections
client
overview 43
using the Client Configuration Assistant 32
using the command line processor 43
server
using the Client Configuration Assistant 32
using the Command Line Processor 43, 119
MAXCONN, keyword 213
memory requirements
client 3
estimating 3
recommended 3
server 3
Message Reference 531
Microsoft ODBC Driver Manager 173
Microsoft SNA Client
configuring 92, 292
version required 92, 292
Microsoft SNA Server
version required 10
migration
data 483
DB2 Version 2.x 483
instance 483, 484, 485
post-installation 483
MODE, keyword 214
mode name 72, 136, 250
MODEENT 136, 250
MULTICONNECT, keyword 214
MVS
configuring 247

N

Named Pipes
client 43
configuring 43
server 121
software requirements 8
naming rules
authorization IDs 394
database 546
database alias 546
database objects 546
general 545
groups 547
instance names 547
password 549
userids 547
username 15, 547
workstation name (nname) 548
National Language Support (NLS)
code set 541
codepage support 541, 542
determining codepage 545
Oracle code page options 470
Nbf
network route required by DB2 128
Net.Data 8
net8 wrapper 466
NetBIOS
autostarting NetBIOS 129
on the client 8, 56
on the server 119, 125
Netfinity Server 8
network ID 72, 249
network name 72, 249
network route
setting to Nbf 126, 128
nicknames
for DB2 family data sources 460
for Oracle data sources 469
nodes
cataloging in a federated environment 457

O

ODBC
catalog for DB2 Connect 192
odbc.ini file 180
odbcinst.ini file 180
registering the driver manager 173
running ODBC-enabled applications 171
running programs 171
odbcad32.exe 173
optimize for N rows
configuration keyword 215
OPTIMIZEFORNROWS, keyword 215
OPTIMIZESQLCOLUMNS, keyword 216

ORA_NLS 465
ORACLE_BASE 465
Oracle data sources
  code page options 470
  creating nicknames 469
  creating server mappings 467
  creating user mappings 468
  creating wrappers 466
  installing client software 463
  linking to client libraries 464
  setting DB2_DJ_COMM 467
  setting environment variables 464
  tnsnames.ora file 467, 469
  verifying connections 470
ORACLE_HOME 464
OS/400
  configuring 247

P

Parallelism, Setting degree of 199
parameters
  FEDERATED 457, 463
  kernel configuration 405
  setting kernel configuration 21, 413
  SYSADM_GROUP 481
partner LU name 72, 249
partner node name 249
password
  naming rules 549
PATCH1, keyword 216
PATCH2, keyword 217
planning
  DB2 configuration 3
  DB2 Connect configuration 3
POPUPMESSAGE, keyword 218
privileges
  required 481
product
  components 419
product files, restoring 399
protocols
  APPC 70, 135, 247
  IPX/SPX 62, 130
  Named Pipes 43, 121
  NetBIOS 56, 125
  TCP/IP 48, 122, 235
PU 72, 136, 250
PWD, keyword 218

Q

Quick Beginnings for OS/2 532
Quick Beginnings for UNIX 532
Quick Beginnings for Windows NT 532

R

RDB_NAME (VSE or VM) 250
REFRESH DEFERRED 196
REFRESH IMMEDIATE 196
registering
  ODBC driver manager 173
registry values
  db2comm 120
  db2nbadapters 126
relational database name 72, 250
remote link address 72, 250
remote transaction program 72, 250
Replication Guide and Reference 531
response files
  OS/2 446
  Windows 3.x 446
restoring product files, installation 391
restrictions
  instance name 547
running applications
  database client 169
  ODBC considerations 171

S

SCHEMALIST, keyword 219
server mappings
  for DB2 family data sources 459
  for Oracle data sources 467
SET CURRENT SCHEMA 196
setting configuration parameters 43, 119
setting up client communications
  using the Client Configuration Assistant 32
  using the command line processor 43
setting up document server 537
setting up server communications
  using the command line processor 119
  using the Control Center 116
SNA
  configuring Communications Server for Windows NT SNA Client manually 87, 281
  configuring Microsoft SNA Client manually on Windows 92, 292
  steps to configure SNAPlus 94, 101, 105, 295, 311, 324
  steps to configure SunLink SNA 108, 327
snapmanage 108, 327
software bundles
  overview 391
  using to install software 391
software requirements
  communication protocols 6
  DB2 client 6, 8, 9, 10
  DB2 Connect 6
  DB2 Software Developer's Kit 6, 7, 8, 9, 10
  DB2 Universal Database 6
  Net.Data 6, 8, 9, 10
SPM_LOG_FILE_SIZE 351
SQL/DS
  configuring 247
SQL Getting Started 531
SQL Reference 531
sqlnet wrapper 466
SQLSTATEFILTER, keyword 220
SSCP 249
steps for configuration 78, 87, 89, 93, 101, 105, 263, 280, 283, 295, 311, 324
SunLink SNA subsystem
  checking status 108, 327
  starting 108, 327
  stopping 108, 327
symbolic destination name 72, 250
SYNCPOINT, keyword 220
Syncpoint Manager (SPM)
  cannot use the CP name LU 346, 348, 349
  configuring manually 346
  logging 341, 351
  memory utilization 352
  migration 350
  performance 352
  prerequisites 340
  problem determination 352
  registering the SPM LU 347
  resolving indoubt transactions 350
  restriction 348
  setting the SPM_NAME 341, 347, 348, 349
  SQL30081N error on DRDA1 calls 348
  stopping and starting 345
SYSADM 394, 406, 414
  privileges
    controlling 481
SYSADM_GROUP parameter 481
SYSSCHEMA, keyword 221
System Monitor Guide and Reference 531

T

TABLETYPE, keyword 222
target database name 72, 250
target logical node 193
TCP/IP
  client 48, 235
  configuring 48, 235
  preventing socket collisions 48, 122, 235
  resolving host address 125
  server 122
  setting up client-to-server 48, 235
  software requirements 6, 7, 8, 9
  supports only one-phase commit 357
  troubleshooting 48, 122, 235
TEMPDIR, keyword 223
TNS_ADMIN 465
tnsnames.ora file 467, 469
TRACE, keyword 224
TRACECOMM, keyword 225
TRACEFILENAME, keyword 225
TRACEFLUSH, keyword 226
TRACEPATHNAME, keyword 227
transaction manager
  database 342
Troubleshooting Guide 531
TXNISOLATION, keyword 228

U

UID, keyword 228
UNDERSCORE, keyword 229
user mappings
  for DB2 family data sources 459
  for Oracle data sources 468
username
  naming rules 547
users
  creating 406, 414
using the command line processor (CLP)
  cataloging a database 45, 53, 59, 66, 109, 242, 329
  cataloging a node 44, 51, 59, 65, 108, 240, 327
  connecting to a database 47, 54, 61, 68, 111, 244
  updating the database manager configuration 58, 124, 128, 132
utilities
  binding 169

V

verifying the connection
  using the command line processor
    APPC 70, 247
    IPX/SPX 62, 130
    Named Pipes 43, 121
    NetBIOS 56, 125
    TCP/IP 48, 122, 135, 235
VM
  configuring 247
VSE
  configuring 247
VTAM
  application name is Partner LU name 249

W

WARNINGLIST, keyword 230
What's New 531
wrappers
  for DB2 family data sources 458
  for Oracle data sources 466

Contacting IBM

This section lists ways you can get more information from IBM.

If you have a technical problem, please take the time to review and carry out the actions suggested by the *Troubleshooting Guide* before contacting DB2 Customer Support. Depending on the nature of your problem or concern, this guide will suggest information you can gather to help us to serve you better.

For information or to order any of the DB2 Universal Database products contact an IBM representative at a local branch office or contact any authorized IBM software remarketer.

Telephone

If you live in the U.S.A., call one of the following numbers:

- 1-800-237-5511 to learn about available service options.
- 1-800-IBM-CALL (1-800-426-2255) or 1-800-3IBM-OS2 (1-800-342-6672) to order products or get general information.
- 1-800-879-2755 to order publications.

For information on how to contact IBM outside of the United States, see Appendix A of the IBM Software Support Handbook. You can access this document by accessing the following page:

`http://www.ibm.com/support/`

then performing a search using the keyword “handbook”.

Note that in some countries, IBM-authorized dealers should contact their dealer support structure instead of the IBM Support Center.

World Wide Web

http://www.software.ibm.com/data/
http://www.software.ibm.com/data/db2/library/

The DB2 World Wide Web pages provide current DB2 information about news, product descriptions, education schedules, and more. The DB2 Product and Service Technical Library provides access to frequently asked questions, fixes, books, and up-to-date DB2 technical information. (Note that this information may be in English only.)

Anonymous FTP Sites

ftp.software.ibm.com

© Copyright IBM Corp. 1993, 1999

Log on as anonymous. In the directory /ps/products/db2, you can find demos, fixes, information, and tools concerning DB2 and many related products.

Internet Newsgroups

comp.databases.ibm-db2, bit.listserv.db2-l

These newsgroups are available for users to discuss their experiences with DB2 products.

CompuServe

GO IBMDB2 to access the IBM DB2 Family forums

All DB2 products are supported through these forums.

To find out about the IBM Professional Certification Program for DB2 Universal Database, go to http://www.software.ibm.com/data/db2/db2tech/db2cert.html

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC09-2857-00