SYBASE®

Installation Guide

Mainframe Connect DirectConnect™ for z/OS Option

12.6

[Microsoft Windows and UNIX]

DOCUMENT ID: DC34237-01-1260-01

LAST REVISED: May 2005

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

About This Book

This guide describes how to install DirectConnect™ for z/OS on the following platforms:

- HP 9000/800 HP-UX
- IBM RISC System/6000 AIX
- Sun Solaris
- Microsoft Windows

Audience

This guide is for System Administrators or other qualified persons familiar with their system environment, resources, and devices. This includes system administrators or communications specialists who are responsible for setting up communications for DirectConnect for z/OS.

How to use this book

This guide provides the following information:

- Chapter 1, “Introduction,” provides an overview of DirectConnect for z/OS, explains InstallShield, and describes the z/OS installed directory structure.
- Chapter 2, “Sybase Software Asset Management (SySAM),” describes Sybase licensing.
- Chapter 3, “Pre-installation Tasks,” describes the necessary tasks to perform before installation, which includes filling out installation worksheets and preparing for connectivity.
- Chapter 4, “Installing DirectConnect,” describes how to install a DirectConnect on UNIX and Windows platforms, and to configure for connectivity.
- Chapter 5, “Installing DirectConnect Manager,” describes how to install and use DirectConnect Manager.
- Chapter 6, “Performing Post-installation Tasks,” describes server and client post-installation tasks that you need to complete.
- Chapter 7, “Creating Database Tables,” explains how to run SQL scripts to set up database tables for using DB2 UDB data.

- Chapter 8, “Setting Up the DirectConnect TDS Driver,” tells how to set up the DirectConnect (target) TDS driver for using DirectConnect for z/OS.
- Appendix A, “Validating Connectivity Using cicsping and snaping,” describes how to use the cicsping and snaping utilities.
- Appendix B, “DirectConnect Utilities,” describes Sybase utilities used to start a server, create and start a new server, and migrate files from one server to another.
- The glossary provides definitions of technical terms used in this guide.

Related documents

To configure and administer DirectConnect products, use the following documents:

- Enterprise Connect and Mainframe Connect *Server Administration Guide* for DirectConnect
- Mainframe Connect DirectConnect for z/OS Option *User’s Guide for Access Services*
- Mainframe Connect DirectConnect for z/OS Option *User’s Guide for Transaction Router Services*
- DirectConnect Manager online Help

To install and administer the other Mainframe Connect products, use the following documents:

- Mainframe Connect Client Options *Programmer’s Reference* for Client Services Applications
- Mainframe Connect Client Option *Installation and Administration Guide* for CICS
- Mainframe Connect Client Option *Installation and Administration Guide* for IMS and MVS
- Mainframe Connect Client Options *Programmer’s Reference for PL/1*
- Mainframe Connect Client Options *Programmer’s Reference for C*
- Mainframe Connect Client Options *Programmer’s Reference for COBOL*
- Mainframe Connect Server Option *Installation and Administration Guide* for CICS
- Mainframe Connect Client Option *Installation and Administration Guide* for IMS and MVS

- Mainframe Connect Client Options *Programmer's Reference* for Remote Stored Procedures
- Mainframe Connect Server Options *Programmer's Reference for PL/1*
- Mainframe Connect Server Options *Programmer's Reference for COBOL*
- Mainframe Connect Client Options and Server Options *Messages and Codes*

For additional references, use the following documents:

- Open Client™ *Client-Library Reference Manual*
- Open Server™ *Server-Library Reference Manual*
- Open Client and Open Server *Installation Guides* (by platform)
- Open Client and Open Server *Programmer's Supplements* (by platform)

Other sources of information

Use the Sybase Getting Started CD, the SyBooks CD, and the Sybase Product Manuals Web site to learn more about your product:

- The Getting Started CD contains release bulletins and installation guides in PDF format, and may also contain other documents or updated information not included on the SyBooks CD. It is included with your software. To read or print documents on the Getting Started CD, you need Adobe Acrobat Reader, which you can download at no charge from the Adobe Web site using a link provided on the CD.
- The SyBooks CD contains product manuals and is included with your software. The Eclipse-based SyBooks browser allows you to access the manuals in an easy-to-use, HTML-based format.

  Some documentation may be provided in PDF format, which you can access through the PDF directory on the SyBooks CD. To read or print the PDF files, you need Adobe Acrobat Reader.

  Refer to the *SyBooks Installation Guide* on the Getting Started CD, or the *README.txt* file on the SyBooks CD for instructions on installing and starting SyBooks.
- The Sybase Product Manuals Web site is an online version of the SyBooks CD that you can access using a standard Web browser. In addition to product manuals, you will find links to EBFs/Maintenance, Technical Documents, Case Management, Solved Cases, newsgroups, and the Sybase Developer Network.

  To access the Sybase Product Manuals Web site, go to Product Manuals at http://www.sybase.com/support/manuals/.

Sybase certifications on the Web

Technical documentation at the Sybase Web site is updated frequently.

❖ **Finding the latest information on product certifications**

1. Point your Web browser to Technical Documents at http://www.sybase.com/support/techdocs/.
2. Select Products from the navigation bar on the left.
3. Select a product name from the product list and click Go.
4. Select the Certification Report filter, specify a time frame, and click Go.
5. Click a Certification Report title to display the report.

❖ **Creating a personalized view of the Sybase Web site (including support pages)**

Set up a MySybase profile. MySybase is a free service that allows you to create a personalized view of Sybase Web pages.

1. Point your Web browser to Technical Documents at http://www.sybase.com/support/techdocs/.
2. Click MySybase and create a MySybase profile.

Sybase EBFs and software maintenance

❖ **Finding the latest information on EBFs and software maintenance**

1. Point your Web browser to the Sybase Support Page at http://www.sybase.com/support.
2. Select EBFs/Maintenance. If prompted, enter your MySybase user name and password.
3. Select a product.
4. Specify a time frame and click Go. A list of EBF/Maintenance releases is displayed.

   Padlock icons indicate that you do not have download authorization for certain EBF/Maintenance releases because you are not registered as a Technical Support Contact. If you have not registered, but have valid information provided by your Sybase representative or through your support contract, click Edit Roles to add the “Technical Support Contact” role to your MySybase profile.
5. Click the Info icon to display the EBF/Maintenance report, or click the product description to download the software.

Style conventions

The following table explains the style conventions used in this guide.

Table 1: Style conventions

This type of information	Looks like this
Gateway-Library function names	TDINIT, TDCANCEL
Client-Library™ function names	CTBINIT, CTBCANCEL
Other executables (DB-Library™ routines, SQL commands) in text	the dbrpcparam routine, a select statement
Directory names, path names, and file names	*/usr/bin directory*, *interfaces* file
Variables	*n* bytes
SQL Server® datatypes	datetime, float
Sample code	`01 BUFFER PIC S9(9) COMP SYNC`
User input	*01 BUFFER PIC X(n)*
Client-Library and Gateway-Library function argument names	*BUFFER, RETCODE*
Names of objects stored on the mainframe	SYCTSAA5
Symbolic values used with function arguments, properties, and structure fields	CS_UNUSED, FMT_NAME, CS_SV_FATAL
Client-Library property names	CS_PASSWORD, CS_USERNAME
Client–Library and Gateway-Library datatypes	CS_CHAR_TYPE

Syntax conventions

Syntax statements that display options for a command look like this:

COMMAND [*object_name*, [{TRUE | FALSE}]]

The following table explains the syntax conventions used in this guide.

Table 2: Syntax conventions

Symbol	Convention
( )	Include parentheses as part of the command.
{ }	Braces indicate that you must choose at least one of the enclosed options. Do not type the braces when you type the option.
[ ]	Brackets indicate that you can choose one or more of the enclosed options, or none. Do not type the brackets when you type the options.

Symbol	Convention
|	The vertical bar indicates that you can select only one of the options shown. Do not type the bar in your command.
,	The comma indicates that you can choose one or more of the options shown. Separate each choice by using a comma as part of the command.

If you need help

Each Sybase installation that has purchased a support contract has one or more designated people who are authorized to contact Sybase Technical Support. If you cannot resolve a problem using the manuals or online help, please have the designated person contact Sybase Technical Support or the Sybase subsidiary in your area.

CHAPTER 1

Introduction

This chapter provides an overview of DirectConnect for z/OS. It contains the following topics:

Introducing DirectConnect for z/OS

Sybase DirectConnect for z/OS is one of many products that provide the following middleware services:

- Distributed database access services
- Transaction services
- Mainframe integration
- Data delivery
- Object interfaces

DirectConnect products are LAN-based middleware servers that provide access to non-Sybase data sources. DirectConnect for z/OS is Open Server-based software that supports CT-Library, and Open Database Connectivity (ODBC) application program interfaces (APIs).

DirectConnect provides the following middleware services for decision support and Online Transaction Processing (OLTP) applications:

- DirectConnect access services that provide access to DB2 UDB systems
- DirectConnect Transaction Router Services (TRSs) that provide access to remote transactions

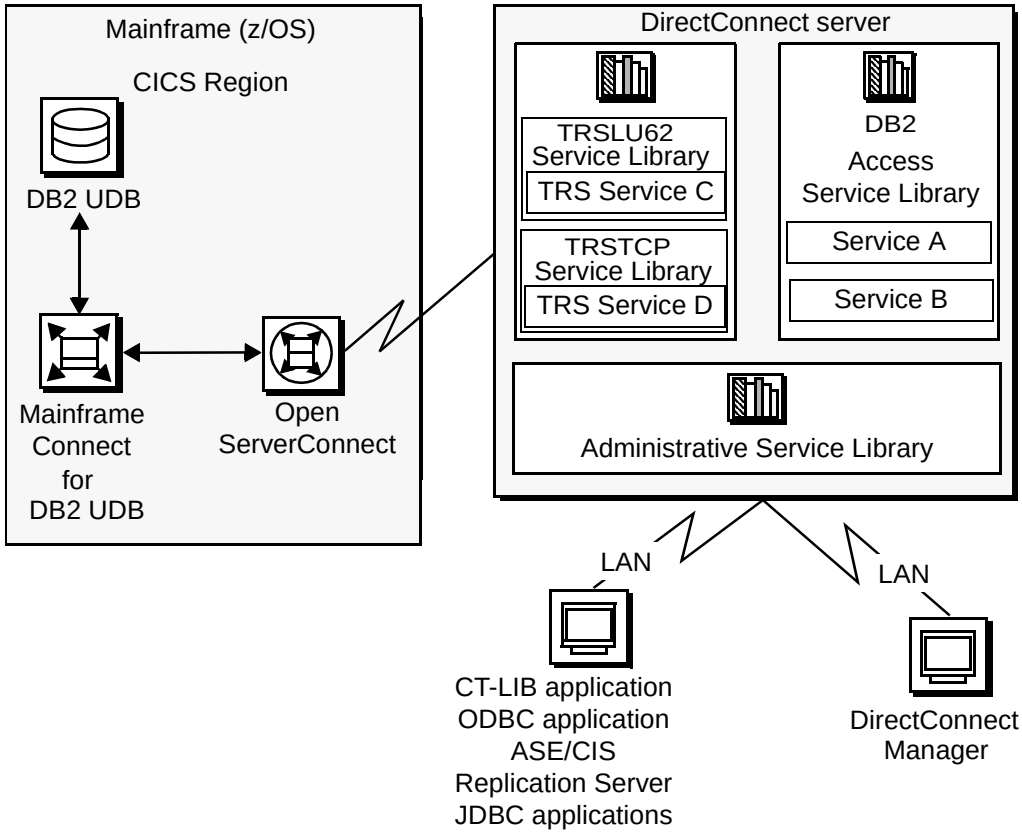
- DirectConnect administrative services that provide server-side systems management

DirectConnect for z/OS is compatible with the following products:

- Open ServerConnect™ 12.6 and later
- Adaptive Server® Enterprise (ASE) 12.5.2 and later
- Replication Server® 12.5 and later
- Replication Agent™ 12.5 and later
- DirectConnect Manager 12.6 and later
- Open Client 12.5.1 and later
- jConnect™ for JDBC™ 6.0 and later

Figure 1-1 shows the DirectConnect for z/OS environment.

Figure 1-1: DirectConnect for z/OS environment



Components of DirectConnect for z/OS

As shown, a request from a client application uses the LAN to communicate with the DirectConnect server. Then, either a TRS service or a DB2 access service routes the request to the appropriate CICS region to access the data on the DB2 UDB database.

The following subsections describe the components of DirectConnect for z/OS that are shown in Figure 1-1.

Mainframe (z/OS)

The CICS region of the Mainframe (z/OS) contains the following:

- DB2 UDB database
- Mainframe Connect for DB2 UDB
- Open ServerConnect

DirectConnect server

The DirectConnect server provides management and support functions for DirectConnect service libraries, such as:

- Routing client connections to the appropriate access service based on user ID, requesting application, and access service name.
- Providing a single log file for access services. TRS has its own Tabular Data Stream ™(TDS) trace file, LU 6.2 protocol trace file, and TCP/IP protocol trace file.
- Logging server, access service, and client messages.
- Tracing server, access service, and client events.
- Providing configuration management of all installed services.

For detailed information about configuring and starting the server, see the DirectConnect *Server Administration Guide*.

DirectConnect service libraries

Residing on the DirectConnect server, a service library is a set of configuration properties that describes how its access services will function. The following service libraries reside on the DirectConnect server:

- Transaction Router Service Library
- DB2 Access Service Library
- Administrative Service Library

Transaction Router Service (TRS)

Each TRS library contains a TRS that provides access to DB2 data and supports Open ServerConnect mainframe applications, defined to TRS as remote procedure calls (RPCs).

There are two TRS libraries:

- *TRSLU62* service library (TRS Service C) which uses the LU 6.2 communications protocol to talk to Mainframe Connect or to any Open ServerConnect application running in CICS.
- *TRSTCP* service library (TRS Service D) which uses the Transmission Control Protocol/Internet Protocol (TCP/IP) communications protocol to talk to MainframeConnect or any Open ServerConnect application running in CICS.

Having multiple instances of a TRS library on a server results in different physical copies of the shared library files that constitute the TRS component.

For an explanation of the TRS components of DirectConnect, see the DirectConnect *Transaction Router Service User's Guide*.

DB2 access service library

The DB2 access service library contains DB2 access services (Service A and Service B) that interface with MainframeConnect for DB2 to allow clients to access DB2 data in a DB2 UDB database.

Each DB2 access service is a specific set of configuration properties that perform the following:

- Transform SQL
- Convert datatypes
- Support RPCs
- Transfer data between DB2 UDB and other servers accessible through Open Client
- Support catalog stored procedures (CSPs) and system stored procedures
- Support RSPs and host-resident requests

Administrative service library

The Administrative service library provides specific administrative services for all DirectConnect libraries, including logging, tracing, and allowing remote configuration of DirectConnect access services (for example, through DirectConnect Manager).

Client Applications

Requests from the following client applications use the LAN to communicate with the DirectConnect server:

- CT-LIB applications
- ODBC applications
- ASE/CIS
- Replication Server
- JDBC applications
- DC Manager

Related products

This section describes Sybase products that work with DirectConnect for z/OS.

MainframeConnect for DB2 UDB

MainframeConnect for DB2 UDB is mainframe software that works with DirectConnect for z/OS to provide access to mainframe data. It performs the following functions:

- Supports full read-write, dynamic SQL access to DB2
- Allows applications to use cursors for flexible and efficient result set processing
- Permits the use of long-running transactions against DB2
- Provides access to mainframe transactions that use static SQL for access to virtually any z/OS data source

Clients can access MainframeConnect for DB2 UDB one of two ways:

- Directly through a DB2 access service or a DirectConnect Transaction Router Service (TRS)
- Indirectly through the CIS functionality of Adaptive Server Enterprise (ASE/CIS), formerly called OmniConnect™, or an ASE remote procedure call (RPC)

Open ServerConnect

Open ServerConnect™ is a programming environment that lets customers create mainframe transactions that are accessible to Sybase clients through DirectConnect. Open ServerConnect uses the following basic interfaces:

- Open Server, the traditional Open Server environment
- Remote stored procedure (RSP), the MDI-heritage programming environment

These transactions provide access to virtually any z/OS data source and are used for a variety of functions, including:

- Initiating mainframe batch jobs
- Providing source data for data transfer operations
- Providing data mapped to a table within the Component Integrated Services (CIS) functionality of ASE, thus allowing results to be accessed or joined with data from other targets

Clients access Open ServerConnect transactions one of two ways:

- Directly through DirectConnect
- Indirectly through the CIS functionality in ASE or through an ASE RPC

Open ClientConnect

Open ClientConnect™ allows customers to create mainframe applications that access LAN data in a Sybase ASE or any other supported data source. Open ClientConnect uses the following APIs:

- Open Client, the traditional Open Client environment
- Client Sservices application (CSA), the MDI-heritage programming environment

Open ClientConnect allows customers to treat the mainframe as just another node on a network. Open Client applications are used to provide referential integrity between mainframe and LAN data, initiating bulk data movements, for example, as part of nightly batch operations.

DirectConnect Manager

DirectConnect Manager is a graphical user interface (GUI) systems management tool that you can use to administer all DirectConnect products. DirectConnect Manager provides the capability to:

- Manage DirectConnect servers on multiple platforms.
- Change configuration properties of DirectConnect servers, service libraries, and services.
- Create and copy services by copying an existing service and giving it a unique name.
- Create new servers using DCDirector.
- Start and stop existing servers using DCDirector.
- Start, stop, and delete services remotely.
- Test the availability of all your data sources.
- Retrieve a DirectConnect server log file or a subset of the log, and view log file messages.
- Update DirectConnect server connection information.
- View the status of a service and data source.

The interface provided by DirectConnect Manager allows you to easily configure and manage all of the data access services directly from your Windows or UNIX platform. As a result, you greatly increase your ability to respond and manage a distributed environment.

You can install DirectConnect Manager and its required components from the DC Client CD.

Note When you install a DirectConnect product on a Windows or UNIX platform or machine, you may install DirectConnect Manager on a separate platform or machine. This allows you to control any DirectConnect from any machine.

Using InstallShield

InstallShield is a Java-based installation program. It unloads and installs all Sybase components using a consistent installation interface across all platforms. You can use this program to install and uninstall a DirectConnect product.

The InstallShield program performs the following functions:

- Creates all directories and copies all DirectConnect programs, utilities, and support files from the distribution media
- Confirms that necessary disk space is available to install and support the DirectConnect programs
- Provides a utility that you can use to uninstall DirectConnect products
- On Windows, modifies the Windows Registry

Directory information

If you install more than one DirectConnect on a machine, you need to take certain precautions. See the following section for more information.

Criteria for creating directories

During installation, the installation program checks the $SYBASE (for UNIX), or %SYBASE% (for Windows) environment variable for any existing Sybase directory that was created for another Sybase product. For example, this might be a directory created for Adaptive Server.

The following rules apply:

- If the InstallShield installation program locates the Sybase directory, it installs DirectConnect subdirectories under this directory. However, before installation begins, you can specify a different location.
- If a destination directory is not available, InstallShield creates a directory. The program creates only what it needs to install the product.
- If InstallShield detects a conflict between new and existing files, you are prompted to choose whether to overwrite the file. At any point, you can cancel the installation.

CHAPTER 2

Sybase Software Asset Management (SySAM)

This chapter describes licensing concepts that you need to know before you begin installing DirectConnect. This chapter contains the following topics:

Note Sybase recommends that you read this entire chapter before beginning the installation of DirectConnect.

SySAM overview

SySAM (Sybase Software Asset Management) is a licensing mechanism that:

- Provides System Administrators with a means to monitor their site's use of Sybase products and optional features.
- Records the Sybase software being used and licensed.

SySAM verifies that a valid license exists for a desired DirectConnect product or feature.

Note For purposes of licensing, DirectConnect products are referred to as DirectConnect features.

The basic components of SySAM are:

- One or more DirectConnect features
- A license file
- The SySAM software, which consists of a license manager and management utilities

When you install DirectConnect, a SySAM license manager is automatically installed. However, after installation, you have two choices:

- You can install the license manager on your local machine.
- You can install the license manager on a remote machine and then point to it from your local machine. To accomplish this, you enter the host name and port number of the remote machine on the License Host Request window. Then, you enter the licenses on the remote machine that will be checking them. For more information about SySAM during installation, see Chapter 4, “Installing DirectConnect.”

Sybase recommends that you have one license manager and have all remote machines point to that license manager. You can have many license managers on many different machines, but you must be sure that you have connectivity pointers to the remote machines, and that the licenses are entered on the machine the license manager is running on.

DirectConnect licenses

DirectConnect assumes that a System Administrator has installed, configured, and started a SySAM license daemon or service server. This license daemon or service server points DirectConnect to the correct license server.

A license file contains a set of information that enables a set of features of a Sybase product set. For purposes of licensing, DirectConnect products are referred to as DirectConnect features. The LM_LICENSE_FILE environment variable is used to point to a license file.

The following identifies the DirectConnect features that are available through the license:

- DB2 Access Service Library
- TRS Library

You must register the DirectConnect licenses into SySAM.

SySAM licenses

Sybase products are licensed individually as features. *Individual licenses* are licenses that must be purchased for each feature.

Required license information

When you purchase your DirectConnect product, Sybase provides a Sybase Software Asset Management certificate with the following information, which is required and must be entered for each license. When you configure the SySAM License Manager it prompts you for the following for each license:

- Order number – the order number of your software purchase.
- Feature name – the feature name from the certificate package or bundle license.
- Feature count – the number of licenses you purchased for the feature package or bundle.
- Software version –the version number specified on the license certificate.
- Authorization code – the authorization code shown on your license certificate for DirectConnect.

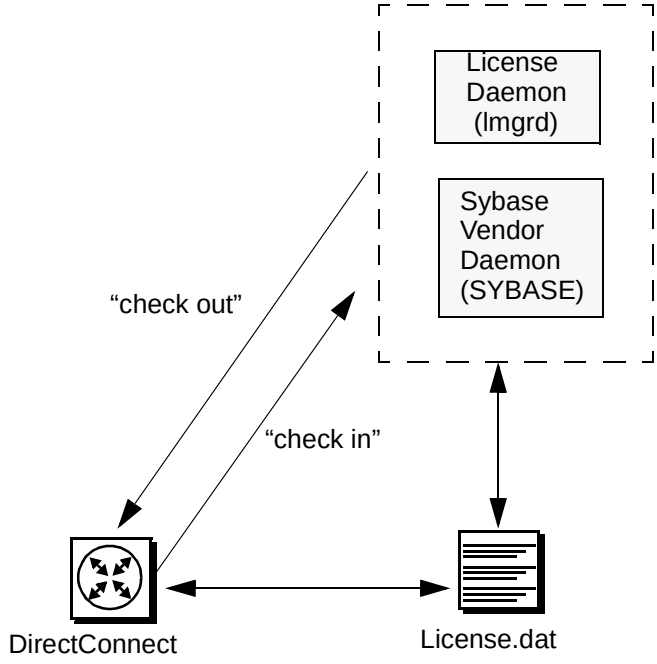
Be sure to keep this information available after you install DirectConnect. This information will be requested for each license.

How SySAM works

SySAM consists of two utilities, called lmgr and lmutil, which manage licensing activities, and two background processes: the license management, lmgrd, and SYBASE. The SySAM lmgrd daemon or service contains a pointer that locates the license file for verification and can reside either on the primary server, known as the local license host, or on a remote license host. The daemons or services handle requests to validate licensed features, as shown in Figure 2-1.

The following is an illustration of the relationship between these components, on UNIX.

Figure 2-1: How SySAM interacts with DirectConnect



Using information in the license file, DirectConnect connects to the SySAM daemon and attempts to check out a license for the base DirectConnect product. If the DirectConnect license is checked out successfully (a license exists in the license file), DirectConnect continues to operate.

If a DirectConnect license is not found, it continues to operate, but it does record this discrepancy.

For Windows only, if you have not registered a license for each service, a pop-up warning will appear for each service, each time that you activate the service. You can enter your license to prevent this from occurring or select cancel, to continue.

Adding feature licenses

Feature licenses can be added using UNIX or Windows.

❖ **To add a license on UNIX**

1 Verify that the license manager software is running. See "SySAM administration" on page 16.

2 Log on to the machine where the license manager is installed (license host).

3 Launch the license manager:

```
$SYBASE/$SYBASE_SYSAM/bin/lmgr
```

4 You should see the following prompt:

```
Do you have Sybase Software Asset Management
Certificates to register?
```

Click Yes.

5 The SySAM License Manager screen prompts you for:

- Order Number
- Feature Name
- Feature Count
- Software Version
- Authorization Code

6 Click More until you have entered all licenses.

7 Click Done.

8 Because you are adding licenses to an existing configuration, you must notify the license daemons of the changes by issuing a reread command. Run the license management utility lmutil lmreread from the *$SYBASE/$SYSAM_SYSAM/bin*/ directory:

```
$SYBASE/$SYBASE_SYSAM/bin/lmutil lmreread
```

9 The new license is appended to the end of the *license.dat* file.

If you encounter problems with new licenses, check the *lmgrd.log* file in the *$SYBASE/$SYBASE_SYSAM/log* directory to see that they were properly appended to the license file.

❖ **To add a license on Windows**

1 Start the license manager if it is not currently running:

From the Windows Start menu, select Programs | Sybase | Sybase Software Asset Management (SySAM).

2 The following prompt appears:

```
Do you have Sybase Software Asset Management
Certificates to register?”
```

Click Yes.

3 The SySAM License Manager screen prompts you for:

- Order Number
- Feature Name
- Feature Count
- Software Version
- Authorization Code

Click More until you have entered all available licenses, then Click Done.

4 Because you are adding additional licenses to an existing configuration, you must notify the license daemons of the changes.

From a command-line prompt, enter:

```
%SYBASE%\%SYBASE_ECON%\bin\lmutil lmreread
```

If you encounter problems with new licenses, check the *lmgrd.log* file in the *%SYBASE%\%SYBASE_SYSAM%\bin* directory to see that they were properly appended to the license file.

SySAM administration

This section describes how to verify the installation and how to start SySAM manually. It also provides additional information for administering SySAM.

Verifying that the License Manager is running

When the lmgrd starts, it automatically starts up SySAM

❖ **To verify that the license management software is running**

1 Enter the following:

- For UNIX:

```
$SYBASE/$SYBASE_SYSAM/bin/lmutil lmstat -c
```

- For Windows:

```
%SYBASE%\%SYBASE_SYSAM%\bin\lmutil lmstat -c
```

2 Verify that both lmgrd and SYBASE are running before you run DirectConnect.

If the SySAM software is not running, you see a message similar to the following UNIX message:

```
lmgrd is not running: Cannot connect to license server
```

If the SySAM software is not running, see "Starting the License Manager manually" on page 17.

Starting the License Manager manually

If the license manager is not running, you must start the license manager manually. Use one of the following methods.

❖ **To start the license manager on UNIX**

- Enter the following command:

```
$SYBASE/$SYBASE_SYSAM/bin/lmgrd -c $LM_LICENSE_FILE -l
        $SYBASE/$SYBASE_SYSAM/log/lmgrd.log &
```

Note If you need help, enter `lmgrd -h`

❖ **To start the license manager on Windows**

1 From the Windows Start menu, select Select Programs | Sybase | Sybase Software Asset management (SySAM).

2 The license manager can run either as a service or as a process.

- To run as a service, enter the following:

```
%SYBASE%\%SYBASE_SYSAM%\sysam.bat
```

- To run as a process, enter the following:

```
lmgrd -c %LM_LICENSE_FILE% -l %SYBASE%\%SYBASE_SYSAM%\log\lmgrd.log
```

CHAPTER 3

Pre-installation Tasks

This chapter outlines the tasks you must complete before you install DirectConnect products on UNIX platforms (HP 9000/800, RISC/6000 AIX, and Sun Solaris) and on Windows platforms, in that order.

Note Most processes and information you need for UNIX and Windows platforms are the same. However, information that differs between UNIX and Windows platforms (such as hardware and software requirements) is provided in separate subsections.

This chapter contains the following installation steps:

1. Gather your installation team

To install DirectConnect, you need a team with specific skills and experience, as well as the authority to carry out the variety of tasks in the installation process. At your site, identify the people who have the following skill sets and keep them informed of any changes.

Table 3-1: DirectConnect for z/OS team skill requirements

Role	Skill set
Operating system administrator	• Understanding of Sun Solaris, HP 9000, AIX, Windows operating systems • Knowledge of standards and conventions at the installation site
Network administrator	• Understanding of connectivity products used at your site, such as TCP/IP and SNA • Ability to design, establish, test, and troubleshoot communications between DirectConnect and the mainframe • Understanding of your network configuration
DirectConnect administrator	• Understanding of the DirectConnect environment • DirectConnect Server administrator privileges
Mainframe administrator	• Knowledge of DB2 UDB database • Knowledge of mainframe environment, including security operations

2. Lay the groundwork

Before you begin, you must build the foundation that will support your DirectConnect installation and subsequent services that you create. If you complete these tasks *first*, you can better ensure a successful installation.

Be sure that the following are in place and operational:

- Network connectivity
- Critical administrative tasks

Network connectivity

Before you begin installation, you need to have network connections in place and operational. This section provides preliminary information for setting up connectivity. To actually configure connectivity for the z/OS mainframe and LAN server, you need to refer to vendor documentation for the connectivity protocol that your site supports.

Following are some recommended tasks to perform before you set up connectivity. This is not a complete list; your team should prepare a list that is pertinent to your site and connectivity setup.

- 1. Hold a planning session
- 2. Check the system requirements
- 3. Research connectivity parameters
- 4. Set up server-to-mainframe connectivity

1. Hold a planning session

The primary purpose of conducting a preliminary planning session is to have the following team members meet:

- VTAM/NCP system programmer
- DirectConnect Administrator
- CICS system programmer
- Mainframe security administrator
- z/OS mainframe system programmer
- LAN system administrator or communications administrator

All team members are required, because each person knows a key piece of information for the configuration parameters.

During this planning session, the team members should perform the activities listed in the following sections as appropriate for a TCP/IP environment.

Assign LU 6.2 configuration tasks

For LU 6.2 connectivity only, the machine on which the DirectConnect server will reside requires SNA connectivity. For more information, see “4. Review hardware and software requirements” on page 26.

Use the following checklist to plan to configure an LU 6.2 environment. If a step contains a team member in parentheses, this means that this team member should perform the step.

1 Determine type of connection you want to use, such as one of the following:

- Token ring or Ethernet attachment through a local 3174 or 3172
- Token ring or Ethernet attachment through front-end processor (FEP)
- Remote SDLC connection

2 Determine the type of session to run:

- Single session
- Parallel sessions

3 Check the other z/OS mainframe configuration properties, including:

- VTAM and NCP node definitions (VTAM/NCP system programmer)
- CICS terminal definitions (CICS system programmer)

4 Create and fill out a LU 6.2 configuration worksheet for your platform (all team members).

5 Determine the server configuration values (LAN system administrator).

6 Determine who performs the z/OS mainframe configuration.

7 Determine who performs the server configuration.

8 Determine who runs the snaping connectivity test for the SNA environment.

Assign TCP/IP configuration tasks

Use the following checklist to plan for a TCP/IP environment.

1 Check the server configuration values (LAN system administrator).

2 Determine who performs the z/OS mainframe configuration.

3 Determine who performs the server configuration.

4 Determine who runs the cicsping connectivity test for the TCP/IP environment.

The following section provides preliminary information about system requirements. Refer to your connectivity protocol documentation for more specific information.

2. Check the system requirements

Verify that you have the platform and mainframe requirements that are listed in the Mainframe installation guides for the client or server option.

z/OS mainframe in an LU 6.2 or TCP/IP environment

- Check with your site's contact person for Sybase Technical Support to determine whether any new release levels or bug fixes of Sybase or vendor products are required.
- The Mainframe Connect DirectConnect Option *Release Bulletin* for Windows and UNIX version 12.6 for the most current information.

Compatibilities

For information about compatibilities for each UNIX platform, see "System requirements for UNIX platforms" on page 26, and for the Windows platforms, see "System requirements for Windows platforms" on page 27.

3. Research connectivity parameters

You need the following parameters for each platform when you set up connectivity for DirectConnect for z/OS using TCP/IP or LU 6.2:

- *ConnectionSpec1, ConnectionSpec2, ConnectionSpec3*
- *ConnectionProtocol*

Be sure to record these values in "5. Complete the installation worksheets" on page 27. Also, tables that describe each of these parameters for UNIX and Windows platforms are located in "Worksheet information and instructions" on page 27.

4. Set up server-to-mainframe connectivity

After you have reviewed the previous sections and prepared your connectivity information, set up connectivity between the machine where the DirectConnect server will reside and the target mainframe. For the steps involved in this process, refer to the documentation for your connectivity protocol or consult the network administrator at your site.

Note Be sure that Open ServerConnect is already installed in CICS on the z/OS mainframe.

After you have set up your SNA LU 6.2 or TCP/IP configurations to go from the local machine to the mainframe, verify this connectivity by running:

- snaping, if accessing any mainframe component over SNA LU 6.2
- cicsping, if accessing the CICS subsystem over TCP/IP
- imsping, if accessing an IMS subsystem over TCP/IP

See Appendix A for information about cicsping and snaping.

If you have problems with your installation *after* you have verified connectivity between the LAN, the DirectConnect server machine, and the target, *and* you have an active session, then notify your site's contact for Sybase Technical Support.

Critical administrative tasks

Perform the following administrative tasks:

- Read the Mainframe Connect DirectConnect Option *Release Bulletin* for 12.6 for Windows and UNIX.
  This document provides product information that may not be included in the DirectConnect for z/OS guides.
- Make a back-up copy of your current Sybase software, particularly if you plan to keep previously-installed DirectConnects or reuse their configurations.
- Make a copy of the installation worksheet for your platform in this chapter for each DirectConnect server that you will install.
- Verify that you have authority to sign in as user sybase.

3. Review the installation process

The following table lists the major installation steps for DirectConnect.

Table 3-2: Installation process steps

Step	Process	Resource
1	Identify the mainframe components.	This chapter
2	Complete the pre-installation tasks.	This chapter
3	Set up and test connectivity.	This chapter
4	Install DirectConnect and DirectConnect Manager using InstallShield.	• Chapter 4 • Mainframe Connect CD • PC Client CD
5	Create a new server: • Edit the *interfaces* file or *sql.ini* file. • Issue the AddServer command. • Use DirectConnect Manager to configure the server and service.	• Chapter 4 • Mainframe Connect CD • PC Client CD
6	Perform post-installation tasks.	• Vendor documentation for your connection • Systems administrator • Chapter 5 • Chapter 6
7	Perform these server tasks: • Set up and test connectivity to the mainframe. • Set up the database tables. • Setup the *locales* files (Windows 2000 and Windows 2003). • Start SySAM and enter the licenses.	• Vendor documentation for your connection • Systems administrator • Chapter 5 • Chapter 6
8	Perform these client tasks: • Configure client connectivity to the DirectConnect services. • Set up the DirectConnect TDS driver.	• Chapter 6 • Chapter 7
9	Install your applications.	Specific documentation for your application

4. Review hardware and software requirements

To use DirectConnect for z/OS, you must have the requirements described in the following section, depending on your platform.

System requirements for UNIX platforms

The following table shows the minimum system requirements for the UNIX-based platforms of HP 9000/800, RISC/6000 AIX, and Sun Solaris.

Table 3-3: System requirements for UNIX platforms

Item	HP 9000/800	RISC/6000 AIX	Sun Solaris
CPU	An HP 9000/800.	An IBM RISC System/6000.	A Sun Solaris (SPARC) system.
RAM	Minimum of 300MB to run HP-UX and any DirectConnect for z/OS products.	Minimum of 300MB to run AIX and any DirectConnect for z/OS products.	Minimum of 300MB to run Sun Solaris and any DirectConnect for z/OS products.
Storage	Minimum of 450MB, plus 512KB for each *locale* you plan to support.	Minimum of 450MB, plus 512KB for each *locale* you plan to support.	Minimum of 450MB, plus 512KB for each *locale* you plan to support.
Software	• Production release of HP 9000/800 and 900 HP-UX 11.23. Patch bundle 990P • HP-UX SNAplus2, R6.11.00.000	• Production release of AIX 5.3. • SNA communications 6.0 (SNA LU 6.2 connectivity only).	• Production release of Sun Solaris 2.9. • SNAP-IX 7.0.2.1 (SNA LU 6.2 connectivity only).

System requirements for Windows platforms

To use DirectConnect for z/OS on Windows platforms, you must have the requirements described in the following table.

Table 3-4: System requirements for Windows platforms

Item	Requirements
CPU	Sybase recommends an Intel-compatible processor with a minimum of 500 megahertz.
RAM	512MB of RAM to run Windows for any DirectConnect z/OS product.
Storage	A minimum of 300MB, plus at least 500KB for each locale you plan to support.

Software	• Microsoft Windows:
	  • Microsoft Data Access Component (MDAC) latest version. This is required for the DirectConnect server machine and the machine running Adaptive Server.
	  • For SNA LU 6.2 connectivity for Windows 2000, Microsoft Host Integration Server (HIS) 2000 Clients

5. Complete the installation worksheets

Use the installation worksheet to record information relating to installation and connectivity tasks for DirectConnect for z/OS. The worksheet covers the following:

- Information from products that must be installed before and after installing DirectConnect, both Sybase and non-Sybase
- Information that is requested during installation, connectivity setup, and configuration of DirectConnect service configuration files

Worksheet information and instructions

The following sections provide specific information for some of the areas on the worksheet. A worksheet for each platform is provided after this section, followed by other pre-installation tasks.

Remember to fill out a worksheet for each DirectConnect service that you will be installing.

Connection protocol

Specify the protocol that the access service uses to connect to the mainframe, either lu62 or tcpip.

TCP/IP connectivity parameters

The following tables describes the TCP/IP connectivity parameters.

Table 3-5: TCP/IP connectivity parameters for Windows platforms

Parameter	Description	Syntax	Range	Default
Connection protocol	Specifies protocol the access service uses to connect to target database, either lu62 or tcpip	*ConnectionProtocol= [lu62|tcpip]*	na	tcpip
ConnectionSpec1	Specifies name or IP address of mainframe host for TCP/IP communications	*ConnectionSpec1= char*	1-255 char	none
ConnectionSpec2	Specifies port number for CICS region	*ConnectionSpec2= char*	1-255 char	none
ConnectionSpec3	Specifies CICS region name running Open ServerConnect for TCP/IP communications	*ConnectionSpec3= char*	1-255 char	none

LU 6.2 connectivity parameters for UNIX

The following table describes the LU 6.2 parameters for each UNIX platform. Be sure to record these values on the installation worksheet.

Table 3-6: LU 6.2 connectivity parameters for UNIX platforms

Platform	ConnectionSpec1	ConnectionSpec2	ConnectionSpec3
RISC/6000 AIX	Specifies the SNA Server/6000 side profile name	Not used, but must not be left blank	Specifies the SNA mode name for LU 6.2 communications
HP 9000 /800	Specifies the local logical unit (LU) alias for LU 6.2 communi-cations	Specifies the name of the partner logical unit (PLU) alias for LU 6.2 communications	Specifies the SNA mode name for LU 6.2 communications

Platform	ConnectionSpec1	ConnectionSpec2	ConnectionSpec3
Sun Solaris	Specifies the local logical unit (LU) alias for LU 6.2 communi-cations	Specifies the name of the partner logical unit (PLU) alias for LU 6.2 communications	Specifies the SNA mode name for LU 6.2 communications

LU 6.2 connectivity parameters for Windows

The following table describes the LU 6.2 parameters for the Windows platform.

Table 3-7: LU 6.2 connectivity parameters for Windows

Parameter	Description	Syntax	Range	Default
ConnectionSpec1	Specifies the local logical unit (LU) alias for LU 6.2 communications	*ConnectionSpec1*= *char*	1-255 char	none
ConnectionSpec2	Specifies the name of the partner logical unit (PLU) alias for LU 6.2 communications	*ConnectionSpec2*= *char*	1-255 char	none
ConnectionSpec3	Specifies the SNA mode name for LU 6.2 communications	*ConnectionSpec3*= *char*	1-255 char	none

Sybase home directory name

Record the directory in which you will install DirectConnect for z/OS, and record it on your worksheet For DirectConnect, the directory names specified must not contain any uppercase letters, periods, symbolics, or tildes (~).

DirectConnect server name

Record the name of the DirectConnect server that you would like to use, subject to the following restrictions:

- The name must contain 30 or fewer characters and cannot contain spaces or tabs (white space characters).
- All subsequent characters can be letters, numerics (0-9), or the underscore ("_") character.
- It must not be any of the following; *bin*, *codesets*, *drivers*, *install*, *lib*, *locale*, *locales*, *messages*, *sample*, *scripts*, *svclib*, *tables*, or *work*.

DirectConnect service names

Record the name of the DirectConnect services you will be creating, subject to the following restrictions:

- The name must be unique within an installation.

- It must begin with a letter. All subsequent characters must be letters, numerics (0-9), or the underscore ("_") character.
- It must not be the same name as a DirectConnect server.

Port number for DirectConnect server/services

Identify and record the port number for the DirectConnect server and services.

Code set information

Determine the values for the following code sets and enter them on your worksheet:

- DefaultClientCodeset – enter the desired client code set on your worksheet.
- DefaultTargetCodeset – enter the desired target code set on your worksheet.

For additional information regarding code sets, refer to the DirectConnect for z/OS *Access Service User's Guide*.

DirectConnect TPName entry (Windows only)

Record the CICS transaction program name (TP) for a specific Mainframe Connect.

For all other items on the worksheets, consult your installation team or systems administrators.

Installation worksheet for HP 9000/800, RISC/6000 AIX, and Sun Solaris

Using the descriptions provided on the previous pages, fill out one of these worksheets for each DirectConnect server that you plan to install. Keep it with you for each step of the installation process.

Table 3-8: Worksheet for DirectConnect for HP9000/800, RISC/6000 AIX, and Sun Solaris

Description of installation information	Your installation information
Connection Protocol, either TCP/IP or LU 6.2 that the access service uses to connect to the target database.	*TCP/IP:* or *LU 6.2:*
TCP/IP Connectivity:	
ConnectionSpec1—specifies the IP address.	*IP ADDRESS:*
ConnectionSpec2—specifies the port number.	*PORT NUMBER:*
ConnectionSpec3—specifies the CICS region name running Open ServerConnect.	*CICS REGION NAME:*
LU 6.2 Connectivity:	
ConnectionSpec1—specifies the local logical unit (LU) alias for LU 6.2 communications.	*LOCAL LU ALIAS:*
ConnectionSpec2—specifies the name of the partner logical unit (PLU) alias.	*PLU ALIAS:*
ConnectionSpec3—specifies the SNA mode name for LU 6.2 communications.	*SNA MODE NAME:*
DirectConnect Directory Name: Record the *$SYBASE_ECON* and *$SYBASE_OCS* home directory for DirectConnect for z/OS (sets to *DC-12_6* and *OCS-12_5*).	*DIRECTORY NAMES:* *DC-12_6* *OCS-12_5*
DirectConnect Server Name: Identify the name of the DirectConnect server you want to create or update.	*SERVER NAME:*
DirectConnect Service Name(s): Identify the name of the DirectConnect service you want to create or update.	*SERVICE NAME(S):*
Port number used for DirectConnect Server/Service(s): Identify the port number for the DirectConnect server and service.	*PORT NUMBER* *(DC SERVER/SERVICE(S)):*

Installation worksheet for Windows platforms

Using the descriptions provided on the previous pages, fill out one of these worksheets for each DirectConnect server that you plan to install. Keep it with you, because you will be prompted for this information one or more times throughout the installation process.

Note Unlike the UNIX platforms, this worksheet includes two code set parameters and a transaction program name.

Table 3-9: Worksheet for DirectConnect for Windows

Description of installation information	Your installation information
Connection Protocol: either TCP/IP or LU 6.2, that the access service uses to connect to the target database.	*TCP/IP:* or *LU 6.2:*
TCP/IP Connectivity:	
ConnectionSpec1—specifies the IP address.	*IP ADDRESS:*
ConnectionSpec2—specifies the port number.	*PORT NUMBER:*
ConnectionSpec3—specifies the CICS region name running Open ServerConnect.	*CICS REGION NAME:*
LU 6.2 Connectivity:	
ConnectionSpec1—specifies the local logical unit (LU) alias for LU 6.2 communications.	*LOCAL LU ALIAS:*
ConnectionSpec2—specifies the name of the partner logical unit (PLU) alias.	*PLU ALIAS:*
ConnectionSpec3—specifies the SNA mode name for LU 6.2 communications.	*SNA MODE NAME:*
DirectConnect Directory Name: Record the *%SYBASE_ECON%* and *%SYBASE_OCS%* home directory for DirectConnect for z/OS (sets to *DC-12_6* and *OCS-12_5*).	*DIRECTORY NAMES:* *DC-12_6* *OCS-12_5*
DirectConnect Server Name: Identify the name of the DirectConnect server you want to create or update.	*SERVER NAME:*
DirectConnect Service Name(s): Identify the name of the DirectConnect service(s) you want to create or update.	*SERVICE NAME(S):*
Port number used for DirectConnect Server/Service(s): Identify the port number for the DirectConnect server and service(s).	*PORT NUMBER (DC SERVER/SERVICE(S))*:

Description of installation information	Your installation information
Default Client Codeset: Enter the desired code set for the configuration property.	*CLIENT CODESET:*
Default Target Codeset: Enter the desired code set for the configuration property.	*TARGET CODESET:*
DirectConnect TPName Entry: Enter the CICS transaction program name (TP) for a specific Mainframe Connect.	*TPNAME:*

6. Review previously installed Sybase products

When you install DirectConnect for z/OS into an existing directory structure, be aware of any previously installed Sybase software. InstallShield will install a version of Open Client and Open Server with DirectConnect for z/OS.

CHAPTER 4

Installing DirectConnect

This chapter describes how to use InstallShield to install a DirectConnect server, create a server and services, and uninstall DirectConnect on UNIX and Windows platforms.

This chapter covers the following topics:

Installing DirectConnect for z/OS

The InstallShield program has greatly simplified the installation process of DirectConnect products. Through a series of interactive windows, you can select the following:

- DirectConnect product or product set
- Type of installation
- Installation directory
- Component selection

Pre-installation instructions

Before you attempt to install DirectConnect, you must complete the following pre-installation tasks.

❖ **To perform pre-installation tasks**

1 Read Chapter 3, “Pre-installation Tasks,” to identify the hardware and software requirements and to perform the identified critical administrative tasks.

2 (For Windows) Back up the system environment before starting the installation.

Note (For Microsoft Windows only) Record your system environment variables if you are installing DirectConnect and other Sybase products on the same machine.

3 Complete the installation worksheet. See the worksheet for your platform in Chapter 3, “Pre-installation Tasks.”

4 If using an SNA target, verify that the SNA libraries are in the library search path.

5 Set the environment variable %SYBASE% (Windows), and $SYBASE (UNIX) to point to the desired location where DirectConnect will be installed.

Note When installing on Windows, Sybase recommends that you exit all programs before running the InstallShield program.

Using InstallShield for installation

This section describes how to install DirectConnect using InstallShield. These procedures assume that:

- The target computer meets the requirements outlined in Chapter 3, “Pre-installation Tasks.”
- You have completed the list of tasks for DirectConnect in“Pre-installation instructions.”

The installation program creates the target directory (if necessary) and loads all the selected components into that directory.

Installation options

You have three options for installing Sybase components from the distribution media using the installation program:

- GUI (graphical user interface) mode, which allows you to install the components using the InstallShield interface.
- Console mode, which allows you to install components using a command line interface.
- Response file mode, which allows you to record or create a response file. Using a response file, you can install DirectConnect two different ways:
  - Silent, which lets you install the product without any interaction required on your part. This is convenient if you are performing identical installations on multiple machines.
  - Interactive installation using response file, which lets you install interactively, but with all the responses already filled in, so that you can accept or change the default values and install DirectConnect according to the responses in the response file. This can be convenient if several sites are installing DirectConnect and must conform to a standard installation.

Each of these options is described in the following sections.

Note Sybase recommends that you use the GUI mode for installing DirectConnect.

Installing DirectConnect using GUI mode

The following describes the DirectConnect installation process in GUI mode.

❖ **To install DirectConnect**

1 Verify that the drive on which you install the products has enough free disk space for the products being installed, and at least 10MB extra disk space for the installation program. The program frees this space after the installation is complete.

2 Insert the Mainframe Connect DirectConnect CD-ROM into the CD-ROM drive.

For UNIX, if auto-mount is not available, follow your local mounting procedures.

3 InstallShield should start automatically. If it does not start:

- For Windows, select Start | Run, and enter the following, where *x:* is your CD drive:

  ```
  x:\setupwin32.exe
  ```

- For UNIX, enter the following, depending on the platform:
  - For AIX:

    ```
    ./setupAIX
    ```

  - For HP11:

    ```
    ./setupHP11
    ```

  - For Solaris:

    ```
    ./setupSolaris
    ```

The Welcome window appears. Click Next.

You may see the following error message:

```
Error writing file =  There may not be enough
temporary disk space. Try using -is:tempdir to use a
temporary directory on a partition with more disk
space.
```

If so, set the temporary directory to another directory that has more disk space by entering the following at the command line:

- For Windows:

  ```
  x:\setupwin.exe -is:tempdir <directory_name>
  ```

- For UNIX:

  ```
  ./setup<platform> -is:tempdir <directory_name>
  ```

where *directory_name* is the name of the temporary directory to which the installation program will write its temporary files.

4 Read the License and Copyright Agreement. Using the drop-down list at the top of the window, select the geographic location where you are installing DirectConnect to display the agreement appropriate to your region. Select “I agree” and click Next.

You must agree to the license and copyright agreement before you can continue.

Note In certain cases when DirectConnect is installed with other Sybase products, you may see warnings about overwriting newer versions of files. In these cases, simply instruct the installer to overwrite these files and proceed with the installation.

5 In the installation directory window, click Next to accept the default directory for the installation, or enter a different directory where you want to install DirectConnect. One of the following occurs:

- If the installation directory you chose does not exist, InstallShield prompts:

  ```
  The directory does not exist. Do you want to
  create it?
  ```

  Click Yes.

- If the installation directory exists and contains files, the software prompts:

  ```
  You have chosen to install into an existing
  directory. Any older versions of the products you
  choose to install that are detected in this
  directory will be replaced.
  ```

  Note You will not see this message if the directory exists but is empty.

  If you are prompted to overwrite any *DLLs*, (Windows) or files (UNIX) select Yes *only* if the version of the new *DLL* or file is later than the one you are attempting to overwrite.

6 Select Yes or No to indicate if you have SNA installed on your system. Click Next.

7 Select one of the two installation types:

Note Sybase recommends that you select the Custom install.

- Full, which installs all DirectConnect products and components from the CD.
- Custom, which allows you to select the components for DirectConnect to install.

Click Next.

8 Select the components that you want to install. Because all the components are already “checked” or selected, you must “uncheck” or deselect the components that you do *not* want to install.

Note Be aware that if you deselect a component that is required, it is automatically installed if it is needed to run other selected components.

After you have selected the desired components, click Next.

9 Depending on your response to step 7, one of the following occurs:

- If you selected Yes, to indicate that SNA is installed, click Next.
- If you selected No, to indicate that SNA is not installed, a window appears with the following warning:

```
The "TRS Access Service for SNA option" while
shown as a selectable item on the next screen,
WILL NOT BE INSTALLED because you do not have SNA
on your system.
```

Click Next.

10 Before proceeding to the next window, InstallShield verifies the selections, and checks for dependencies and available disk space. The Product Summary window displays every component that is to be installed and the total disk space required for all the selected components.

Note If the target directory does not have enough free space, the space-required and the space-available information is displayed. If you click Next without sufficient hard disk space, an error occurs that stops the installation.

Verify that you have selected the correct type of installation, and that you have enough disk space to complete the process. Click Next to continue the installation.

11 The installation program unloads all the components from the CD and displays a progress indicator. When the installation is complete, a message appears indicating that the InstallShield Wizard has completed installing your product. Click Next.

12 A window displays a message indicating that the installation was successful and advising you to check for software updates. You are also advised to enter Software Asset Management Certificates by running the following tool:

- For Windows:

```
C:\%SYBASE%\%SYSAM-1_0%\bin\mgr
```

- For UNIX:

```
C:/$SYBASE/$SYSAM-1_0/bin/mgr
```

Click Finish to clear the window.

Installing DirectConnect in console mode

If you want to run the installer without the graphical user interface (GUI), you can launch InstallShield in console mode. In cases where InstallShield launches automatically, click Cancel to cancel the GUI installation and then launch the setup program from a terminal or console.

❖ **To install in console mode**

The steps for installing components in console mode are the same as those described in “Installing DirectConnect using GUI mode” on page 37, except that you execute the InstallShield installation program from the command line using the setup -console command, and you enter text to select the installation options, as follows:

1 Enter the following at the command line:

- For Windows:

```
x:\setupwin32 -console
```

- For UNIX, enter the following depending on your platform:
  - For AIX:

```
./setupAIX -console
```

  - For HP11:

```
./setupHP11 -console
```

  - For Solaris:

```
./setupSolaris -console
```

Note The setup command will be identified as setup<*platform*> for the remaining UNIX entries in this chapter. Enter the appropriate platform identifier in place of the <*platform*> entry.

The installation program starts and displays a Welcome message.

2 The flow of the installation is identical to a regular GUI installation, except that the display is written to a terminal window, and responses are entered using the keyboard. Follow the remaining prompts to install DirectConnect.

Installing using a response file

A silent installation (sometimes referred to as an unattended install) is done by running InstallShield and providing a response file that contains answers to all of InstallShield's questions.

Creating a response file

There are two methods of generating a response file for InstallShield:

- *Record* mode: In this mode, InstallShield performs an installation of the product, and records all of your responses and selections in the specified response file. You must complete the installation to generate a response file. To create a response file, enter the following:
  - For Windows:

```
x:\setupwin32.exe -options-record <responseFileName>
```

  - For UNIX:

```
./setup<platform> -options-record <responseFileName>
```

where *responseFileName* is a name you choose for the response file.

Note There should be no space between *-options* and *-record*.

The following are the results:

- An installation of DirectConnect on your computer
- A response file containing all of your responses from the installation

  If this response file is used for a silent installation, the resulting installation is identical to the one from which the response file was created: the same installation location, same feature selection, and all of the same remaining information. The response file is a text file that you can edit to change any responses prior to using it in any subsequent installations.

- *Template* mode: In this mode, InstallShield creates a response file containing commented-out values for all required responses and selections. However, you do not need to install the product, and you can cancel the installation after the response file has been created. To create this template file, enter the following:

  - For Windows:

```
x:\setupwin32.exe -console -options-template <responseFileName>
```

  - For UNIX:

```
./setup<platform> -console -options-template <responseFileName>
```

  where *responseFileName* is the absolute file name you chose for the response file, for example:

  - For Windows:

```
C:\DC\OptionsTemplate.txt
```

  - For UNIX:

```
/sybase/DC/OptionsTemplate.txt
```

  If run in console mode, as shown in the previous example, InstallShield provides a message indicating that the template creation was successful. If run in GUI mode, no message is provided. However, you can click Cancel immediately and a response file is created.

Then, if you use this response file for a silent installation, the default values for all responses are used. Edit the template with the values you want to use during installation.

Warning! If you created a response file using a Custom installation, you need to edit the response file to allow the custom selections to be chosen correctly during installation. The following is a workaround for an installer issue when using a response file.

Use a text editor and delete the "Custom" setup type in the following line

```
-W setupTypes.selectedSetupTypeId=Custom
```

The resultant line will look similar to the following:

```
-W setupTypes.selectedSetupTypeId=
```

Failure to delete the “Custom” setup type results in a Full installation of the product.

Interactive installation using a response file An interactive installation using a response file allows you to accept the default values obtained from the response file that you have set up, or to change any of those values for the specific installation. This is useful when you want multiple similar installations but with some minor differences that you want to change at installation time.

At the command line, execute the following command (enter the command all on one line):

- For Windows:

```
x.\setupwin32.exe -console -options
<responseFileName>
-W SybaseLicense.agreeToLicense=true
```

- For UNIX:

```
./setup<platform> -console -options
<responseFileName>
-W SybaseLicense.agreeToLicense=true
```

Installing in silent mode A silent mode installation, sometimes referred to as an unattended installation, allows you to install the product with all responses being taken from the response file that you have set up. There is no user interaction. This is useful when you want multiple identical installations or when you want to automate the installation process.

At the command line, execute the following command (enter the command all on one line):

- For Windows:

```
x.\setupwin32Console.exe -silent -options
<responseFileName>
-W SybaseLicense.agreeToLicense=true
```

- For UNIX

```
./setup<platform> -silent -options
<responseFileName>
-W SybaseLicense.agreeToLicense=true
```

where *responseFileName* is the name of the file containing the installation options you chose. The -W option specifies that you agree with the Sybase License Agreement text.

Warning! In windows only, Sybase recommends that you use the *setupwin32Console.exe* executable, which runs in the foreground, when running a silent installation. The normal *setupwin32.exe* executable runs in the background, giving you the impression that the installation has terminated immediately, without a completion status. This could result in duplicate installation attempts.

Except for the absence of the GUI screens, all actions of InstallShield are the same, and the result of an installation in silent mode is exactly the same as one done in GUI mode with the same responses.

Setting up and verifying your environment

Set up the environment

Sybase provides scripts to set the following variables, which DirectConnect needs to run:

- SYBASE
- SYBASE_ECON
- SYBASE_OCS
- PATH (Windows), SHLIB_PATH (HP-UX), LIB_PATH (AIX), LD_LIBRARY_PATH (Sun Solaris)

Set the environment variables from a command line prompt by running one of the following environment scripts, as appropriate for your platform. For Windows, use a DOS command session.

- *DC_SYBASE.csh or DC_SYBASE.sh* on UNIX
- *DC_SYBASE.bat* on Windows

For AIX, HP-UX, and Sun Solaris using SNA LU 6.2 protocol, you must locate the following line in the environment script that begins as follows (this example is for Sun Solaris):

```
set LD_LIBRARY_PATH.........
```

Add the following to the end of this line:

```
/opt/sna/lib:$LD_LIBRARY__PATH
```

Note Your SNA location may vary.

To verify that the environment script ran successfully on UNIX, execute the following:

```
echo $SYBASE
```

The response should match the home directory for your installation.

```
echo $SYBASE_ECON
```

The response should be DC-12_6.

```
echo $SYBASE_OCS
```

The response should be OCS-12_5.

```
echo $SHLIB_PATH
```

This variable (shown for HP) should equal the following:

```
$SYBASE/$SYBASE_ECON/lib
```

Note For UNIX: $SYBASE, $SYBASE_ECON, and $SYBASE_OCS will be expanded to their actual values.

Verify that the environment script ran successfully

To verify that the environment script ran successfully on Windows, execute the following in a DOS command window:

```
echo %SYBASE%
```

The response should match the home directory for your installation.

```
echo %SYBASE_ECON%
```

The response should be DC-12_6.

```
echo %SYBASE_OCS%
```

The response should be OCS-12_5.

```
echo %PATH%
```

This variable should equal the following:

```
%SYBASE%\%SYBASE_ECON%\lib
```

For Windows, keep your DOS command line session active to create the DirectConnect server, instructions for which follow in the next section.

Note For Windows: %SYBASE%, %SYBASE_ECON%, and %SYBASE_OCS% will be expanded to their actual values.

Creating a new DirectConnect server

At this point, InstallShield has automatically created the subdirectories, files, and tables needed for the DirectConnect server and for its selected access service components, the DB2 access service and TRS. Now, you must perform a number of tasks to create a new server and services.

Running the AddServer utility

Run the AddServer utility. Use the syntax in the following example:

- For UNIX:

  Note You must be in the *$SYBASE/$SYBASE_ECON/bin* directory for this C shell script to work properly.

  ```
  cd $SYBASE/DC-12_6/bin
  AddServer srvname 1234
  ```

- For Windows:

  Note You must be in the *%SYBASE%\%SYBASE_ECON%\bin* directory for this batch file to work properly.

  ```
  cd %SYBASE%\DC-12_6\bin
  AddServer srvname 1234
  ```

  where:

  - *srvname* is the name of the new server.
  - *1234* is the port that it will listen on.
  - AddServer is the utility that will:
    - Set the environment
    - Create the server
    - Start the server
    - Insert server name in Sybase *interfaces* file

Creating a new DirectConnect access service

You can use DirectConnect to create services by editing the service library configuration file one of two ways:

- Using DirectConnect Manager
- Using a text editor

Using DirectConnect Manager

You can use DirectConnect Manager to configure, start, and stop DirectConnect servers and access services. For instructions, refer to Chapter 5, “Installing DirectConnect Manager.”

Using a text editor

You can use a text editor to configure the service library configuration file (*db2.cfg* or *trslu62.cfg* or *trstcp.cfg).* The service library configuration files reside in the *cfg* subdirectory under the directory with the same name as the DirectConnect server.

❖ **To configure the service using a text editor**

1 Change to the *cfg* subdirectory of the directory with the name of the server you created. If you successfully created and started a new server, then this directory contains a basic version of *db2.cfg*, *trslu62.cfg* and *trstcp.cfg*.

2 At the end of the configuration file, enter a name for the service in brackets, for example:

```
[db2_acs]
```

3 Save and close the configuration file.

To make the configuration effective, you must restart the DirectConnect server. This procedure is described in the next section, called "Starting a DirectConnect access service."

Adding a new service to the interfaces file (UNIX)

The following describes how to allow an Open Client application, such as isql, to connect to the Open Client installation that will be used by your client application. Add the new service to the *interfaces* file as a Server Object using the dsedit utility, specifying the same connectivity information for this Server Object as specified in the *sql.ini* file for the DirectConnect server.

To preserve a unique directory structure and environment for Sybase applications, each Sybase product must have a unique user or start-up script for each application. The key file for maintaining this uniqueness is the Open Server network address file. On UNIX-based systems, this is the *interfaces* file, which resides in the *$SYBASE* directory.

The network address file configures the Open Server application listening network address. On UNIX systems, it configures the network type and TCP/IP values, plus the operating system type and network.

❖ **To modify the *interfaces* file using the *dsedit* utility**

1 Right-click the server object.

2 From the menu, click Add to display the Input Server Name window.

3 Enter the server object and modify its attributes.

4 On the Network Address window that appears, enter protocol information for your site.

5 Enter the machine name on which you installed the DirectConnect server and the connectivity parameter.

6 Click OK twice. Exit dsedit.

Defaults are provided for the server configuration parameters. For additional information, see the DirectConnect *Server Administration Guide*.

Adding a new service to the sql.ini file (Windows)

To allow an Open Client application, such as isql, to connect to the Open Client installation that will be used by your client application. Add the new service to the *sql.ini* file as a Server Object using the dsedit utility, specifying the same connectivity information for this Server Object as specified in the *sql.ini* file for the DirectConnect server.

Starting a DirectConnect access service

The only way you can start the new access service without stopping and restarting the DirectConnect server is by using DirectConnect Manager.

To start your new service without using DirectConnect Manager, be sure that the EnableAtStartup=Yes property is set in the service configuration (*.cfg*) file, which starts the service automatically when the server is started, and then restart the DirectConnect server.

Verifying a DirectConnect access service

Verify that the access service is connecting to its target data source and working properly by using one of the SQL utilities.

Using isql from a command line

Note If you are using isql on the same machine on which you installed DirectConnect, you must add a access service entry to the *interfaces* (for UNIX) or *sql.ini* (for Windows) file that points to the access service you are testing.

❖ **To verify the access service configuration using *isql***

1 Do one of the following:

- On the UNIX client machine, use dsedit to create an *interfaces* file entry for the access service.
- On the Windows client machine, use dsedit to create a *sql.ini* file entry for the access service.

Be sure to enter the access service name exactly as you defined it in the configuration file.

2 Set the environment variables from a command line prompt by running one of the following environment scripts, as appropriate for your platform:

- For UNIX:

```
DC_SYBASE.csh or DC_SYBASE.sh
```

- For Windows:

```
DC_SYBASE.bat
```

3 For UNIX and Windows, run isql from the command line by entering the following:

```
isql -SServiceName -Uuserid -Ppassword
```

where:

- *ServiceName* is the name of the access service exactly as you defined it in the *interfaces* or *sql.ini* file.
- *userid* is a valid user ID for the target database.
- *password* is a valid password for the user ID in the target database.

If the connection is successful, a 1> prompt appears.

Note If you cannot connect, the access service might not be running. Start the access service with DirectConnect Manager, edit the configuration file, and then verify that the Enable at Startup parameter is set to Yes. This starts the access service automatically when the server is started.

4 At the 1> prompt, query a table in the target database by entering a select statement and pressing Return.

5 At the 2> prompt, enter:

```
go
```

Press Return.

The query should run and return a result set, followed by a >1 prompt.

6 To exit isql, enter the following at the 1> prompt:

```
exit
```

Press Return.

A regular operating system command appears.

Note At this point, you can stop the DirectConnect server using the stopsvr utility, which shuts down the server and terminates all client connections. However, if you plan to perform post-installation tasks for the client and server, leave DirectConnect up and running.

Uninstalling DirectConnect

To uninstall a DirectConnect installation, use the InstallShield uninstall feature. This removes all servers, all common files, and all required components. After the uninstall runs, you may need to delete a number of existing files and directories in the directory where DirectConnect was installed.

Note You may need to stop the SySAM service first; otherwise, you will not be able to delete certain files and directories.

You can invoke the uninstall procedure using either the GUI or the console method. Sybase recommends that you use the GUI method.

Before uninstalling Sybase software, shut down all Sybase applications and processes. Log in to your machine using an account with "administrator" privileges, then shut down all other processes for the components you are uninstalling.

❖ **To uninstall in GUI mode**

1 Enter one of the following:

For Windows:

- Using Windows Explorer:
  - Go to the directory where your application is installed.
  - Select the uninstall directory.
  - Select the application you want to uninstall.
  - Double-click the uninstall icon. The wizard appears.
- Using the Add/Remove program:
  - Go to Start | Setting | Control panel | Add/Remove Programs dialog
  - Select the application you want to remove
  - Click on Change/Remove. The wizard appears.

For UNIX:

- Go to the directory where your application is installed.
- Go to the uninstall directory.
- Go to the directory for your application that you want to uninstall.
- Enter "uninstall" from the command line. The wizard appears.

2 The Welcome window appears.

3 Click Next to display the list of selected products and components that were installed.

4 All the products and components that were installed are displayed and already checked to allow you to remove the total installation. If you do not want to remove a product or component, deselect or "uncheck" that product or component. Click Next.

5 A summary of all the products and components is displayed. Click Next.

6 A message indicating that the removal of a z/OS installation is in progress. When this is completed, a message is displayed indicating a successful uninstall. Click Next to end the uninstall.

❖ **To uninstall in console mode**

1 Enter one of the following:

- For Windows, go to the *%SYBASE%* uninstall directory, and at the DOS window prompt, enter the following:

```
%SYBASE%\uninstall.exe -console
```

- For UNIX, execute the following at the command line:

```
$SYBASE/uninstall/uninstall -console
```

The uninstall program starts.

2 Choose the DirectConnect software product you want to uninstall.

The DirectConnect software product you chose is uninstalled.

CHAPTER 5

Installing DirectConnect Manager

This chapter describes how to install and use DirectConnect Manager to create a server and service, and to start a server and service.

This chapter contains the following topics:

Installing DirectConnect Manager software

This section describes the DirectConnect Manager installation requirements and the installation process.

Installation requirements

To install DirectConnect Manager, you must have, at a minimum, the system requirements in the following tables.

System requirements for UNIX

The following table shows the minimum system requirements for the UNIX-based platforms.

Table 5-1: DirectConnect Manager requirements for UNIX platforms

Item	HP 9000/800	RISC/6000 AIX	Sun Solaris	Linux
CPU	An HP 9000/800	An IBM RISC System/6000	A Sun Solaris (SPARC) system	Linux
RAM	Minimum of 256MB	Minimum of 256MB	Minimum of 256MB	Minimum of 256MB
Storage	Minimum of 150MB	Minimum of 150MB	Minimum of 150MB	Minimum of 150MB

System requirements for Windows

The following table shows the minimum system requirements for Windows.

Table 5-2: DirectConnect Manager system requirements for Windows

Item	Requirements
CPU	Microsoft Windows. Sybase recommends a Windows processor with a minimum of 1 gigahertz.
RAM	Sybase recommends 256MB.
Storage	120MB.

Note A Java Virtual Machine (JVM), is provided with the DirectConnect Manager installation program as a Java Runtime Environment executable.

Installing DirectConnect Manager

The following describes how to install DirectConnect Manager using InstallShield.

❖ **To install DirectConnect Manager**

1 Insert the PC Client CD into the CD drive.

2 Go to the DirectConnect Manager directory.

3 Execute one of the following:

- For Windows:

```
setupwin32.exe
```

- For UNIX, depending on the platform:
  - For AIX:

```
setupaix.bin
```

  - For Solaris:

```
setupsolarisSparc.bin
```

  - For HPUX:

```
setuphp11x.bin
```

  - For Linux:

```
setuplinux.bin
```

4 The Welcome window appears. Click Next to proceed with the installation.

5 Read the License and Copyright Agreement. Using the drop-down list at the top of the window, select the geographic location where you are installing to display the agreement appropriate to your region. Select "I agree" and click Next.

6 On the Choose Directory dialog box, enter the directory path for the installation, and click Next.

You must agree to the license and copyright agreement before you can continue.

7 Select one of the two types of installation:

- Install a new version of Sybase Central and render the other install invalid. Click Next.

  This type of installation will install DirectConnect Manager as well as all the other necessary components and create a new set of icons and Registry entries.

  Note This option will make your current Sybase Central installation invalid.

- Register with existing Sybase Central. Click Next.

  This type of installation will install the DirectConnect Manager plug-in and register it with the existing Sybase Central Application.

8 A Summary window appears and lists all the features that you are going to install. Click Next.

9 The installation process begins and a Setup window appears.

The Setup window shows the percentage (%) of installation that is complete, and monitors the decompressing and installing of the DirectConnect Manager files. The installation will register the plug-in with Sybase Central.

10 When installation is completed, a DirectConnect Manager *README* file appears. After reading the file, Click Finish to clear the window.

Using DirectConnect Manager

Before you can use DirectConnect Manager to start a server, or create and start an access service, you must identify and establish a connection between the server and DirectConnect Manager.

Configuring servers and access services

After you create a DirectConnect server or a DCDirector, you can configure the server through DirectConnect Manager. For a description of all the properties, refer to the following:

- For the DirectConnect server, refer to the Enterprise Connect Data Access and Mainframe Connect *System Administration Guide* for DirectConnect.
- For the DirectConnect access service, refer to the Mainframe Connect DirectConnect Option *User's Guide for DB2 Access Services*.

Creating a new DirectConnect access service

For instructions on how to use DirectConnect Manager to create a service, go to the DirectConnect Manager online Help and select Managing Access Services | Creating a New Service or Copying a Service.

Note If you use service name redirection, the connectivity parameters must match the connectivity parameters or the default, and the access service name must map through the redirection file to the DirectConnect server.

For more information about service name redirection, see Enterprise Connect Data Access Mainframe Connect *System Administration Guide* for DirectConnect.

Starting a DirectConnect access service

The only way you can start the new access service without stopping and restarting DirectConnect is by using DirectConnect Manager. For instructions on how to use DirectConnect Manager to start a service, go to the DirectConnect Manager online Help and select Managing Access Services | Starting a Service.

Stopping a DirectConnect access service

For instructions on how to use DirectConnect Manager to stop a service, go to the DirectConnect Manager online Help and select Managing Access Services | Stopping a Service.

Adding a new service

To add a new service to a DirectConnect server, use DirectConnect Manager to copy an existing service and modify it as necessary to connect to the desired target database or use the Create New Service option.

Configuring servers and access services

After you create a DirectConnect server or an access service, you can use DirectConnect Manager to configure them:

- To identify the configuration properties and to configure the DirectConnect server, refer to the Enterprise Connect Data Access Mainframe Connect *System Administration Guide* for DirectConnect.
- To identify the access service properties and to configure a service, refer to the Mainframe Connect DirectConnect z/OS Option *User's Guide for DB2 Access Services*.

Uninstalling DirectConnect Manager

To uninstall DirectConnect Manager, use the InstallShield uninstall feature.

❖ **To uninstall on Windows**

1 Go to Settings | Control Panel | Add/Remove Programs

2 Select the installed program, DirectConnect Manager Version 12.6, and then select Change/Remove

❖ **To uninstall on UNIX**

1 Go to the <*DCManager installation directory*>

2 Execute the following:

```
uninstall_DCM.bin
```

Warning! Some files will not be uninstalled during this process. Please review and delete them manually.

CHAPTER 6

Performing Post-installation Tasks

After you install DirectConnect for z/OS, you must perform a number of post-installation tasks. The tasks are divided into two groups: those completed in the UNIX environment, and those completed in your Windows environment.

This chapter provides guidelines for performing these tasks:

UNIX tasks

This section describes post-installation server and client tasks for UNIX platforms.

Post-installation server tasks for UNIX

The following server procedures ensure that the UNIX environment is properly configured for mainframe connectivity.

Perform the following tasks for installations using TCP/IP only:

- Test client-to-mainframe connectivity
- Test DirectConnect DB2 access service connectivity to mainframe
- Verify TRS connectivity
- Set up database tables
- Stop the DirectConnect server

Test client-to-mainframe connectivity

To test connectivity from your UNIX workstation to the z/OS mainframe, use one of the following (check your worksheet for the information requested):

For LU 6.2 connectivity to the CICS mainframe, using snaping

For testing connectivity to the CICS region, enter the following command using the values defined previously and recorded on your worksheet as *ConnectionSpec1*, *ConnectionSpec2*, and *ConnectionSpec3*:

snaping -C*ConnectionSpec1* **-R***ConnectionSpec2* **-M***ConnectionSpec3* **-U***hostuserid* **-P***hostpassword*

Use the values for these configuration properties found in a service on your worksheet that has *ConnectionProtocol=lu62*.

Note that this is an example of successful snaping output (for a Sun Solaris platform):

```
> snaping -C LOCAL -R CICSAMD2 -M MVSMODE -U userid
-P password

Verify LU6.2 connectivity with a host transaction.  A
connection may be specified individually, or an entire
configuration can be verified using the configuration
file created by DirectConnect's Transaction Router
Service.

Usage: snaping [-C Local_LU -R Partner_LU -M Modename]
[-L connection_file][-T Host_Tran_id]  [-P password]
[-U userid]

Defaults are:Connection file - $SYBASE/ngcid.[$DSQUERY]

Host Transaction - SYI1
Userid and passord - Null

Allocating to SYI1 on CICSAMD2 using LOCAL with mode
MVSMODE..Ok

Sending data...Ok
Waiting for response...Received data 187 bytes.

<Open Server Module: SYGWCICS, Version: ZRL/1260GA
/P /z/OS V2R9  /11/19/01    11.03. Open Client Module:
SYGWCICC, Version: ** SYGWZZRL/1260GA  /P       /z/OS
V2R9  /11/19/01 .>

Normal Deallocate

Done
```

```
Test complete
```

If you receive an error or you need additional information on how to use the snaping utility, refer to Appendix A, "Validating Connectivity Using cicsping and snaping.'

For TCP/IP connectivity to the CICS mainframe, using cicsping

For testing TCP/IP connectivity to a CICS region, enter the following command using the values defined previously and recorded on your worksheet as *ConnectionSpec1* and *ConnectionSpec2*. Note that this example is for a Sun Solaris platform:

```
cicsping -HConnectionSpec1 -NConnectionSpec2 -Uhostuserid -Phostpassword
```

Use the values for these configuration properties found in a service defined on your worksheet that has ConnectionProtocol=tcpip.

An example of successful cicsping:

```
Torreys> cicsping -H sungard -N 3020
Sybase CICSPING (cicsping/3.0.1/P/sun_svr4/Solaris 2.5/1/OPT/Mar 25 12:00:00
1997)
Verify TCP/IP connectivity to a CICS region.  A host and port may be specified
individually, or an entire configuration can be verified using the region
definition file created by the Sybase MSG.
Usage:  cicsping [-H hostname -N portnumber ]
                 [-T Host_Tran_id] [-U userid  [-P password]]
Defaults are:   Region file  - $SYBASE/ngreg.[$DSQUERY]
                Host Transaction - SYPG
                Userid and password - Null
Testing host sungard, port 3020
        get host by name ok
        connect to CICS listener ok
        transaction started by listener ok
        test message sent ok
        Reply from host transaction:
DWMCI410 : *** MODULE SYGWTCPS LOADED SUCCESSFULLY ***
VERSION STRING = SYGWTCPS/310 EBF /P       /z/OS R1.2  /05/19/97    10.55
ECHO DATA = abcdefghijklmnopqrstuvwxyz
        test completed in 1228ms
```

If you receive an error or you need additional information on how to use the cicsping utility, refer to Appendix A, "Validating Connectivity Using cicsping and snaping.'

Test DirectConnect DB2 access service connectivity to mainframe

❖ **To test DB2 access service connectivity to the mainframe**

1 To test DirectConnect DB2 access service connectivity to the mainframe without DirectConnect, you can use one of two methods:

- Start the DirectConnect Server that you created using the following syntax from the command line:

```
DCStart -Ssrvname
```

  where *srvname* is your DirectConnect server name.

- Start up DirectConnect Manager and click on the server icon.

2 Test the connectivity between the access service and DB2 by using the isql utility and entering:

```
isql -Sservice_name -Umainframe userid -Pmainframe password
```

where:

- service_name is the name of your service.
- *userid* and *password* are your mainframe user ID and mainframe password.

The connection is successful when you see the prompt.

Verify TRS connectivity

The following tests the connectivity between TRS and the mainframe.

Note This section assumes that you have already installed the *SYM2* RPC.

❖ **To define the test region**

1 Log into TRS as “sa”:

```
isql -Strstcpservice -Usa -P
```

2 At the isql prompt, enter the following command:

```
exec sgw_addregion region, "ipaddress"|hostname, "portnumber"
go
```

Note Any entry beginning with a number must be entered with double quotes.

where:

- *region* is the name of the destination CICS region, or you can use ConnectionSpec3 from a service definition worksheet.
- *ipaddress*|*hostname* is the IP address or the hostname, the name that corresponds to the TCP/IP network hostname, or you can use ConnectionSpec1 from a service definition worksheet.
- *portnumber* is the port number, or you can use ConnectionSpec2 from a service definition worksheet.

❖ **To define the test connection**

1 Log into TRS as "sa":

```
isql -Strslu62service -Usa -P
```

2 At the isql prompt, enter the following command:

```
exec sgw_addconn locallu, remotelu, snamode, "max_sessions"
go
```

Note Any entry beginning with a number must be entered with double quotes.

where:

- *locallu* is the local logical unit (LU), or you can use ConnectionSpec1from your worksheet.
- *remotelu* is the name of the partner logical unit (PLU), or it can be the ConnectionSpec2 on your worksheet.
- *snamode* is the mode name, or it can be ConnectionSpec3 from your worksheet.
- *max_sessions* is the maximum number of sessions that can run concurrently over this connection.

Define the test RPC

The following describes how to define an RPC to execute in the defined region. The SYM2 transaction is a simple CICS transaction that simulates data and requires no external resources such as DB2 or VTAM.

- At the isql prompt, enter the following command:

```
exec sgw_addrpc SYM2, SYM2, CICSregion, security
go
```

where:

- *sgw_addrpc* is the RPC name.
- SYM2 (1st occurrence) is the RPC name of the remote procedure.
- SYM2 (2nd occurrence) is the transaction ID at the mainframe.
- *CICSregion* is the CICS region name:
  - For TCP/IP, it must match the region name given in the sgw_addregion procedure.
  - For LU 6.2, it must match the remote lu parameter in the sgw_addcon procedure.
- security is the type of login information TRS passes to the transaction processing region. Enter one of the following:
  - none, specify none to indicate that no user IDs are passed to the mainframe for this test.
  - both, specify both to send both user ID and password to the mainframe for this test.
  - userid, specify userid to send user ID to the mainframe for this test.

Run the SYM2 sample

Enter the following at the isql prompt to run the SYM2 sample:

```
exec SYM2 a, 4
```

The output should be similar to the following:

```
TESTDATA
-----------------------------------------------
       U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
       U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
       U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
       U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
(4 rows affected, return status = 0)
```

CSP scripts

Sybase provides three scripts for you to use with CSPs:

- *addcat* - adds the CSPs to TRS.
- *dropcat* - drops the CSPs from TRS.

- *testcat* - tests the CSPs (requires that the AMD2 transaction be installed at the mainframe).

Installing CSPs

The addcat script executes the sgw_addrpc procedure automatically for each CSP. Before you run addcat, modify the script to suit your installation.

Use your text editor to specify the value of these parameters:

- *region* parameter – name of the region you want the CSPs to execute against.
- *security* parameter – value you can change to meet the security requirements at your installation. If you do not change it, the value is none.
- *rpc_name* parameter – name or value must be coordinated with any change to the RPC names with the mainframe system programmer. If you are using ODBC applications, do not change the RPC names.
- *tran_id* parameter – value or name of this parameter must be coordinated with any change to the transaction ID with the mainframe system programmer.

After you edit the script to suit your installation, run the addcat script as input to your TRS. The following isql example shows how to run the addcat script with a TRS named “new_TRS”:

```
isql -Snew_TRS -Usa -P < addcat
go
```

This script automatically executes the sgw_addrpc procedure for each CSP.

Testing CSPs

The testcat script uses the AMD2 transaction to create temporary tables and execute each CSP. At least one row is returned for each CSP and the testcat script then drops the temporary tables.

Run the testcat script as input to your TRS. The following isql example shows how to run the testcat script with a TRS named “new_TRS”:

```
isql -Snew_TRS -Usa -P < testcat
go
```

This script automatically tests each of the CSPs.

Dropping CSPs

The dropcat script drops the CSPs from TRS. Run the dropcat script as input to your TRS.

The following isql example shows how to run the dropcat script with a TRS named “new_TRS”:

```
isql -Snew_TRS -Usa -P < dropcat
go
```

This script automatically drops the CSPs.

Set up database tables

You can use the isql scripts in the *$SYBASE/scripts* subdirectory to set up PUBS and CSP tables for DB2. For instructions and guidelines, see Chapter 7, "Creating Database Tables."

Stop the DirectConnect server

Stop the DirectConnect server using the stopsrvr utility that shuts down the server and terminates all client connections.

The stopsrvr format is as follows:

```
stopsrvr [-v|-?|-h] -Sserver_name -ddelay
go
```

where:

- -v displays the program version only.
- -? or -h displays the stop server parameters you entered.
- -S shows the name of the server to be shut down.
- -d is the delay, in seconds, before client connections are terminated. The default is 3.

Post-installation client tasks for UNIX

The following procedure ensures that the UNIX client environment is properly connected to the LAN and to the DirectConnect server.

This section covers the following topics:

1. Configure your clients to connect to DirectConnect access services
2. Set up the DirectConnect TDS driver
3. Install your application

1. Configure your clients to connect to DirectConnect access services

Use sqledit or DirectConnect Manager to add, edit, or delete entries in the client *interfaces* file:

- Configure your client *interfaces* file. The access service name and port number must match the entry for the DirectConnect Service on your worksheet. For more information, refer to the DirectConnect *Server Administration Guide*.
- Add a server Query entry for each service you want to connect to.

2. Set up the DirectConnect TDS driver

ODBC clients must use the DirectConnect Tabular Data Stream (TDS) driver to connect with DirectConnect access services. For instructions on setting up the driver, refer to Chapter 8, "Setting Up the DirectConnect TDS Driver."

3. Install your application

If you have connectivity between your server, your workstation, and the mainframe (target database), you are ready to install your applications.

See the appropriate documentation to install your client applications.

Troubleshooting for UNIX platforms

The following describes the errors that can occur during and after installation on UNIX platforms and suggested steps to locate and correct the errors.

System does not work correctly following installation

If your system does not work properly after you install DirectConnect for z/OS products, and if you already performed the connection steps listed previously, try the following:

- Confirm the version number of Open Server software. DirectConnect is compatible with Open Server and Adaptive Server version numbers identified in Chapter 1.
- Check the log and trace files in the server subdirectory *$SYBASE/$SYBASE_ECON/servername/log* subdirectory for more information. Following are examples of error messages that can occur:

```
Error : 16029 Severity : 20 State: 0 OS Error: -
```

```
1 : Failed to start any network listeners OS Error
Text : <srv-lib>
```

Following are the possible causes:

- One of the specified port numbers is in use. Change the port number to one that is not in use and try again.

  or

- The wrong machine name or IP address was specified. Enter the machine name or the IP address running the DirectConnect server.

Refer to the DirectConnect *Error Message Guide* for errors that occur.

If the server fails before the log files initialize, error messages are written to the console. If this occurs, see the Enterprise Connect Data Access and Mainframe Connect *Server Administration Guide* for DirectConnect for an explanations of these “pre-log” messages.

DirectConnect server fails to start after installation

If the DirectConnect server fails to start after installation and the SNA library files are listed in error in the DirectConnect log, one of the following messages appears, depending on the platform.

Messages for Sun Solaris

```
LoadLibrary failed : .../dc12621ss/
DC-12_6/svclib/db2.so : ld.so.1: direct: fatal:
libappc.so: open failed: No such file or directory

LoadLibrary failed : .../dc12621ss/
DC-12_6/svclib/db2.so : ld.so.1: direct: fatal:
libcsv.so: open failed: No such file or directory

LoadLibrary failed : .../dc12621ss/
DC-12_6/svclib/db2.so : ld.so.1: direct: fatal:
libmgr.so: open failed: No such file or directory

LoadLibrary failed : .../dc12621ss/
DC-12_6/svclib/db2.so : ld.so.1: direct: fatal:
libsna_r.so: open failed: No such file or directory
```

Messages for HP-UX

```
usr/lib/dld.sl: Can't find path for shared library:
libmgr.sl

/usr/lib/dld.sl: Can't find path for shared library:
libappc.1
```

```
/usr/lib/dld.sl: Can't find path for shared library:
libcsv.1
```

Messages for RISC/6000 AIX

```
LoadLibrary failed : /.../dc12621aix/
DC-12_6/svclib/db2.so : No such file or directory

Could not load service library: /.../dc12621aix/
DC-12_6/svclib/db2.so

LoadLibrary failed : /.../dc12621aix/
DC-12_6/svclib/trslu62.so : No such file or directory

Could not load service library: /.../dc12621aix/
DC-12_6/svclib/trslu62.so
```

Error conditions and their fixes

Error condition #1

SNA software does not exist on the UNIX machine, and during installation the response to the question about SNA software existing on the machine was "yes."

Fix for error #1

❖ **To correct error condition #1**

1 Go to the following directory:

```
cd $SYBASE/$SYBASE_ECON/bin
```

2 Execute the following script:

```
DCConfig
```

3 Respond *NO* to the question, “Is SNA installed on your system.”

This creates the soft links to the *SNA stub* libraries as a substitute for the actual libraries.

Error #2

SNA software does exist on the UNIX machine. However, the LD_LIBRARY_PATH (for Sun Solaris and HP-UX) or LIBPATH (for RISC/6000 AIX) environment variable is missing in the directory where the *SNA* libraries reside.

Fix for error #2

To fix error #2, change your LD_LIBRARY_PATH or LIBPATH to include the */opt/sna/...* (for Sun Solaris and HP-UX), or */usr/lib/sna/...* (for AIX) directory, or the custom directory where your SNA software is installed on your UNIX machine. The SNA libraries are now in the DirectConnect library path.

Windows tasks

This section describes post-installation server and client tasks for Windows platforms.

Post-installation server tasks

The following server procedures ensure that the Windows environment is properly configured for mainframe connectivity.

For installations using TCP/IP only:

- Test Windows-to-mainframe connectivity
- Test DirectConnect DB2 access service connectivity to the mainframe
- Set up database tables
- Verify TRS connectivity
- Stop the DirectConnect server

Each of these tasks are described in the following subsections.

Test Windows-to-mainframe connectivity

To test connectivity from the Windows workstation to the z/OS mainframe, use one of the following (the information requested is located on your worksheet).

For LU 6.2 connectivity to the CICS mainframe, using *snaping*

For testing connectivity to the CICS region, use the snaping utility. Enter the following command using the values defined previously and recorded on your worksheet as *ConnectionSpec1*, *ConnectionSpec2*, and *ConnectionSpec3*:

snaping -C*ConnectionSpec1* **-R***ConnectionSpec2* **-M***ConnectionSpec3* **-U***hostuserid* **-P***hostpassword*

Use the values for these configuration properties found in a service that has ConnectionProtocol=*lu62* in this subdirectory:
$SYBASE/$SYBASE_ECON/server_name/cfg/db2.cfg

An example of successful snaping output:

```
C:\sql10\bin>snaping -C LOCAL -R DWMCI410 -M MVSMODE -U DWMD33 -P GOODHART
Sybase SNAPING (snaping/3.0.1/P/PC Intel/Windows 3.51/1/OPT/Mar 17 10:00:00
1997)
```

```
Verify LU 6.2 connectivity with a host transaction. A connection may be
specified individually, or an entire configuration can be verified using the
configuration file created by the Sybase Net Gateway.
This program is intended to be run against the CICS transaction SYI1.
Usage:  snaping [-C Local_LU -R Partner_LU -M Modename]
                [-T Host_Tran_id] [-U userid [-P password]]
                [-L Connection_file]
Defaults are:   Connection file  - %SYBASE%\[%DSQUERY%].cid
                Host Transaction - SYI1
                Userid and Password - Null
Allocating to SYI1 on DWMCI410 using LOCAL with mode MVSMODE...Ok
       Sending data...Ok
       Waiting for response...Received data 194 bytes.
       <Received 000000000000026 bytes total from userid: DWMD33  . Version
SYGWCICC//310GA BU/P       /MVS
/ESA 4.2  /09/12/96   14.44   SYGWCICS/310 EBF /P    /z/OS R1.2  /05/19/9
>
       Normal Deallocate
Done
Test complete
```

If you receive an error or you need additional information on how to use the snaping utility, refer to Appendix A, "Validating Connectivity Using cicsping and snaping.'

For TCP/IP connectivity to the CICS mainframe, using cicsping

For testing TCP/IP connectivity to a CICS region, enter the following command using the values defined previously and recorded on your worksheet as *ConnectionSpec1* and *ConnectionSpec2* (items 2a and 2b):

cicsping -H*ConnectionSpec1* **-N***ConnectionSpec2* **-U***hostuserid* **-P***hostpassword*

Use the values for these configuration properties found in a service that has ConnectionProtocol=*tcpip* in this subdirectory:
$SYBASE/$SYBASE_ECON/server_name/cfg/db2.cfg

An example of successful cicsping output:

```
C:\sql10\bin>cicsping -H sungard -N 3020
Sybase (CICSPING/3.0.1/P/PC Intel/Windows 3.51/1/OPT/Mar 17 10:00:00 PST 1997)
Verify TCP/IP connectivity to a CICS region. A host and port may be specified
individually, or an entire configuration can be verified using the region
definition file created by the Sybase MSG.
Usage:  cicsping [-H hostname -N portnumber ]
                 [-T Host_Tran_id] [-U userid  [-P password]]
Defaults are:   Region file  - %SYBASE%\[%DSQUERY%].reg
                Host Transaction - SYPG
```

```
                Userid and password - Null
Testing host sungard, port 3020
        gethostbyname ok
        connect to CICS listener ok
        transaction started by listener ok
        test message sent ok
        Reply from host transaction:
DWMCI410 : *** MODULE SYGWTCPS LOADED SUCCESSFULLY ***
VERSION STRING = SYGWTCPS/310 EBF /P       /z/OS R1.2  /05/19/97    10.55
ECHO DATA = abcdefghijklmnopqrstuvwxyz
        test completed in 1071ms
```

If you receive an error, or you need additional information on how to use the cicsping utility, refer to Appendix A, "Validating Connectivity Using cicsping and snaping," or to vendor documentation for your connectivity protocol.

Test DirectConnect DB2 access service connectivity to the mainframe

❖ **To test connectivity to the mainframe**

1 Start the DirectConnect server using the following syntax from the command line:

```
DCStart -Ssrvname
```

where *srvname* is your DirectConnect server name.

2 Test the connectivity between the access service and DB2 by using isql and entering:

```
isql -Sservice_name -Umainframe userid -Pmainframe password
```

where:

- service_name is the name of your service.
- *userid* and *password* are your mainframe userID and mainframe password.

Your connection is successful if you see the prompt.

Set up database tables

You can use the isql scripts in the *%SYBASE%\%SYBASE_ECON%\scripts* subdirectory to set up PUBS and CSP tables for DB2. For guidelines, see Chapter 7, "Creating Database Tables."

Verify TRS connectivity

The following procedure tests the connectivity between TRS and the mainframe.

Define the test region (TCP/IP only)

❖ **To define the test region**

1 Log into TRS as "sa":

```
isql -Sservice_name -Usa -P
```

2 At the isql prompt, enter the following command:

```
exec sgw_addregion region, "ipaddress"|hostname, "portnumber"
go
```

Note Any entry beginning with a number must be entered with double quotes.

where:

- *region* is the name of the destination CICS region, or you can use ConnectionSpec3 from a service definition worksheet.
- *ipaddress*|*hostname* is the IP address or the hostname, the name that corresponds to the TCP/IP network hostname, or you can use ConnectionSpec1 from a service definition worksheet.
- *portnumber* is the port number, or you can use ConnectionSpec2 from a service definition worksheet.

Define the test region (LU 6.2 only)

❖ **To define the test connection**

1 Log into TRS as "sa":

```
isql -Strslu62service -Usa -P
```

2 At the isql prompt, enter the following command:

```
exec sgw_addconn lualias, plualias, snamode, "max_sessions"
go
```

where:

- *lualias* is the local logical unit (LU) from your worksheet.

- *plualias* is the name of the partner logical unit (PLU) on your worksheet.
- *snamode* is the SNA mode name from your worksheet.
- *max_sessions* is the maximum number of sessions that can run concurrently over this connection.

Define the test RPC

Define an RPC to execute in the defined region. The SYM2 transaction is a simple CICS transaction that fabricates data and requires no external resources such as DB2 or VTAM.

❖ **To define the test RPC**

- At the isql prompt, enter the following command:

```
exec sgw_addrpc SYM2, SYM2, CICSregion, security
go
```

where:

- *sgw_addrpc* is the RPC name.
- SYM2 (1st occurrence) is the RPC name of the remote procedure.
- SYM2 (2nd occurrence) is the transaction ID at the mainframe.
- *CICSregion* is the CICS region name:
  - For TCP/IP, it must match the region name given in the sgw_addregion procedure.
  - For LU 6.2, it must match the region parameter in the sgw_addcon procedure.
- *security* is the type of user login information TRS passes to the transaction processing region. Enter one of the following:
  - none – indicates that no user IDs are passed to the mainframe for this test.
  - both – sends both user ID and password to the mainframe for this test.
  - userid – sends the user ID to the mainframe for this test.

Run the sample

❖ **To run the sample**

- Enter the following at the isql prompt to run the "SYM2" sample:

```
exec SYM2 a, 4
```

The output should be similar to the following:

```
TESTDATA
--------------------------------------------
   U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
   U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
   U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
   U6T42P01
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
(4 rows affected, return status = 0)
```

Check for error messages

The MSG request can return any of several types of error messages. Some mainframe access products messages are written to the error log at MSG; others are returned to the client.

CSP scripts

Sybase provides three scripts for you to use with CSPs:

- addcat – adds the CSPs to TRS.
- dropcat – drops the CSPs from TRS.
- testcat – tests the CSPs (requires that the AMD2 transaction be installed at the mainframe).

Installing CSPs

The addcat script executes the sgw_addrpc procedure automatically for each CSP. Before you run addcat, modify the script to suit your installation.

Use your text editor to specify the value of these parameters:

- *region* parameter – name of the region you want the CSPs to execute against.
- *security* parameter – value you can change to meet the security requirements at your installation. If you do not change it, the value is none.

- *rpc_name* parameter – name or value must be coordinated with any change to the RPC names with the mainframe system programmer. If you are using ODBC applications, do not change the RPC names.
- *tran_id* parameter – value or name of this parameter must be coordinated with any change to the transaction ID with the mainframe system programmer.

After you edit the script to suit your installation, run the addcat script as input to your TRS. The following isql example shows how to run the addcat script with a TRS named new_TRS:

```
isql -Snew_TRS -Usa -P < addcat
go
```

This script automatically executes the sgw_addrpc procedure for each CSP.

Testing CSPs

The testcat script uses the AMD2 transaction to create temporary tables and execute each CSP. At least one row is returned for each CSP and the testcat script then drops the temporary tables.

Run the testcat script as input to your TRS. The following isql example shows how to run the testcat script with a TRS named new_TRS:

```
isql -Snew_TRS -Usa -P < testcat
go
```

This script automatically tests each of the CSPs.

Dropping CSPs

The dropcat script drops the CSPs from TRS. Run the dropcat script as input to your TRS.

The following isql example shows how to run the dropcat script with a TRS named new_TRS:

```
isql -Snew_TRS -Usa -P < dropcat
go
```

This script automatically drops the CSPs.

Stop the DirectConnect server

Stop the DirectConnect server using the stopsrvr utility that shuts down the server and terminates all client connections.

The stopsrvr format is as follows:

```
stopsrvr [-v|-?|-h] -Sserver_name [-ddelay]
```

- -v displays the program version only.

- -? or -h displays the stop server parameters you entered.
- server_name is the name of the server to be shut down.
- delay is the delay, in seconds, before client connections are terminated. The default is 3.

Post-installation client tasks for Windows

The following procedures ensure that the Windows client environment is properly connected to the LAN and to the DirectConnect servers.

This section covers the following tasks:

- Install a DirectConnect server as a Windows service
- Configure your client connectivity
- Set up the DirectConnect TDS driver
- Install your application

Install a DirectConnect server as a Windows service

Using the new InstallShield, DirectConnect no longer automatically creates the server as a Windows service. However, you can run a DirectConnect server as a Windows service. Instructions can be found in the Mainframe Connect DirectConnect *Server Administration Guide*.

Configure your client connectivity

Use the dsedit Windows program to add, edit, or delete entries in the *sql.ini* and *libtcl.cfg* files:

- Configure your client *sql.ini* file. The service name and port number must match the entry for the DirectConnect Service.

  For additional information, see the DirectConnect *Server Administration Guide*.

- Add Query for each service to which you want to connect.

Set up the DirectConnect TDS driver

ODBC clients must use the DirectConnect TDS driver to connect with DirectConnect access services. For instructions on setting up the driver, refer to Chapter 8, “Setting Up the DirectConnect TDS Driver.”

Install your application

If you have connectivity between your server, your workstation, and the mainframe (target database), you are ready to install your applications.

See the appropriate documentation for the client application you plan to install.

Troubleshooting for Windows platforms

The following describes the errors that can occur during and after installation on Windows platforms and suggested steps to locate and correct the errors.

System does not work correctly following installation

If your system does not work properly after you install DirectConnect for z/OS products, and if you already performed the connection steps listed previously, try the following:

- Confirm the version number of Open Server software. DirectConnect is compatible with Open Server and Adaptive Server version numbers identified in Chapter 1.
- Check the log and trace files in the server subdirectory *%SYBASE%\%SYBASE_ECON%\ServerName\log* subdirectory for more information. Following are examples of error messages that can occur:

  ```
  Error : 16029 Severity : 20 State: 0 OS Error: -
  1 : Failed to start any network listeners OS Error
  Text : <srv-lib>
  ```

  Following are the possible causes:

  - One of the specified port numbers is in use. Change the port number to one that is not in use and try again.

    or

- The wrong machine name or IP address was specified. Enter the machine name or the IP address running the DirectConnect server.

Refer to the DirectConnect *Error Message Guide* for errors that occur.

If the server fails before the log files initialize, error messages are written to the console. If this occurs, see the Enterprise Connect Data Access and Mainframe Connect *Server Administration Guide* for DirectConnect for an explanations of these "pre-log" messages.

DirectConnect server fails to start after installation

If the DirectConnect server fails to start after installation and the SNA library files are listed in error in the DirectConnect log, the following message appears:

```
The application has failed to start because WAPPC32.dll
was not found. Re-installing the application may fix
this problem.
```

Fix for error condition

To fix the error condition, install the *stub* libraries for DirectConnect 12.6 on Windows by executing the following:

```
%SYBASE%\%SYBASE_ECON%\bin\DCConfig.bat.
```

CHAPTER 7

Creating Database Tables

This chapter explains how to run SQL scripts to set up database tables. The tables are necessary for using DB2 with DirectConnect for z/OS.

This chapter contains the following topics:

SQL scripts

Sybase provides SQL scripts that allow you to create a sample database and system tables on DB2. The installation program installs the scripts in the *scripts* subdirectory.

The *scripts* subdirectory is at *$SYBASE/$SYBASE_ECON/scripts* (UNIX) and *%SYBASE%\%SYBASE_ECON%\scripts* (Windows).

The scripts are as follows:

- The *pubsdb2.sql* script creates sample PUBS tables that simulate the PUBS database provided with ASE. By installing PUBS tables on DB2, you can more easily test the transfer process between ASE and DB2. These tables are not required for DirectConnect for z/OS setup, but Sybase recommends installing them to assist Sybase Technical Support.
- The *cspdb2.sql* script creates tables for catalog stored procedures (CSPs). These tables are required for DirectConnect for z/OS setup with DB2.

Use the isql utility to run the scripts. Sybase recommends that you complete all of these tasks from the platform server before you attempt to install Open Client/DB-Library files on any client machines.

Creating PUBS tables

PUBS is a sample database that contains tables that you can use to test the transfer process between DB2 and SQL Server. Use the *pubsdb2.sql* script to create the tables by performing the following steps:

❖ **To create PUBS tables**

1 Start the DirectConnect server.

2 If you are using Windows, display the Windows command line.

3 Edit *pubsdb2.sql* to create the tables in the desired DB2 database and tablespace.

For example, you can modify the script to say *one* of the following:

- IN *databasename.tablespacename*
- IN DATABASE *dataspacename*

Note The user ID must have CREATE TABLE privileges in the specified *databasename.tablespacename*.

4 Run pubsdb2.sql:

```
isql -Uuserid -Ppassword -Sservice_name -ipubsdb2.sql
```

where:

- isql starts the utility of the same name.
- *userid* is a valid DB2 user ID with CREATE TABLE privileges.
- *password* is a valid password for the DB2 user ID you enter.
- *service_name* is the desired DirectConnect service name.

The script drops any tables with the same names as the tables it creates. If the table does not exist, the DropTable command returns an error message. This does not prevent successful execution of the rest of the script. The script creates indexes for the remaining tables.

5 To verify that you successfully created the PUBS database in DB2, log in to DirectConnect:

isql -U*userid* -P*password* -S*service_name*

6 Issue the following select statement against PUBS:

```
select * from authors
```

If you successfully created the PUBS database, DB2 returns data from the authors table.

If you receive an error, run the *pubsdb2.sql* script again or review the log files.

For information about the transfer function, see the DirectConnect for z/OS *Access Service User's Guide*.

Creating CSP tables

The system tables described in the following table are required if you are using CSPs in DB2.

Table 7-1: CSP tables

CSP	System table	Table description
sp_stored_procedure	SYSPROCEDURES	This table contains specific information about stored procedures created at your site.
sp_sproc_columns	SYSPROCCOLUMNS	These tables contain specific information about the arguments in the stored procedures created at your site.

The system administrator maintains the information in each table.

❖ **To use the *cspdb2.sql* script to create the tables**

1 Start the DirectConnect server by issuing the direct -S command.

2 If you are using Windows, display the Windows command line.

3 Run *cspdb2.sql* by entering the following:

```
isql -Uuserid -Ppassword -Sservice_name -icspdb2.sql
```

where:

- isql starts the utility of the same name.
- *userid* is a valid DB2 user ID with create table privileges.
- *password* is a valid password for DB2 for the user ID you entered.
- *service_name* is the desired DB2 service name.

The script drops any tables with the same names as the tables created. If the table does not exist, the DropTable command returns an error message. This does not prevent successful execution of the rest of the script.

4 Log in to DirectConnect and issue a SELECT statement against each table to verify that you successfully created each one.

5 Grant SELECT authorization to the following DB2 system tables for all CSP users:

- SYSIBM.SYSCOLAUTH
- SYSIBM.SYSCOLUMNS
- SYSIBM.SYSDATABASE
- SYSIBM.SYSFOREIGNKEYS
- SYSIBM.SYSINDEXES
- SYSIBM.SYSKEYS
- SYSIBM.SYSRELS
- SYSIBM.SYSSYNONYMS
- SYSIBM.SYSTABAUTH
- SYSIBM.SYSTABLES

Although *cspdb2.sql* automatically grants select authorization to PUBLIC for the tables it creates, you must grant authorization to these DB2 system tables.

CHAPTER 8

Setting Up the DirectConnect TDS Driver

The DirectConnect Tabular Data Stream (TDS) driver allows applications to access DB2 data. ODBC clients must use this driver to connect with DirectConnect for z/OS access services.

The driver supports DirectConnect for z/OS in Microsoft Windows 32-bit environments. InstallShield has installed Sybase Open Client Net-Library and Client-Library to gain access to DirectConnect for z/OS target database objects.

This chapter contains the following topics:

Configuring the data source

❖ **To configure a DirectConnect for z/OS data source**

1 Start the ODBC Administrator by selecting the TDS icon from the Windows control panel.

2 Do one of the following:

- If you are configuring an existing data source, select the source name and choose OK.
- If you are configuring a new data source, choose Add to view the list of installed drivers, then select DirectConnect and choose OK.

3 Specify values for the data source using the information in the following table.

Table 8-1: Required data source configuration properties

Property	Description
DataSourceName	A string that identifies this DirectConnect data source configuration in the *ODBC.ini* file. Examples include "Accounting" or "Sys11-Serv1."
Description	An optional long description of a data source name. Examples include "My Accounting Database" or "DB2 via region DWS1."
ServerName	The name of the DirectConnect server.

4 Specify values for any optional properties that you require. The properties are listed in the following table.

Table 8-2: Optional data source configuration properties

Property	Description
ServerList	The list of servers that appear in the logon dialog box. Use commas to separate the server names.
DatabaseName	The name of the database to which you want to connect by default. If you do not specify a value, the default is the database defined for each user.
DatabaseList	The databases that appear in the logon dialog box. Use commas to separate the database names.
DefaultLogonID	The default logon ID used to connect to DB2. This ID is case sensitive. A logon ID is required only if security is enabled for the database to which you are connecting. Your ODBC application can override this value, or you can override this value in the logon dialog box or the connection string.
InterfacesFile	The path name of the *interfaces* file. The default is the normal Sybase interfaces file.
Initialization String	A string of commands that execute when the data source connection initializes. Use semicolons to separate multiple commands.
ModifySQL Statement	An instruction to the driver to convert a single- or two-part table name to a three-part format for create table SQL statements. Valid values are: • 0 instructs the driver to treat all SQL statements as being in Passthrough transformation mode and to not modify them. • 1 instructs the driver to modify the SQL statements, including convert table names to three-part format and change NULL and NOT NULL statements.

Property	Description
Password Encryption	A number that determines whether password encryption can be performed from the Open Client Library to the server. Valid values are: • 0 prevents encryption from being performed. • 1 allows encryption to be performed.
Charset	The name of a character set corresponding to a subdirectory in *Sybase_root\charsets*. The default is the setting on the DirectConnect server.
WorkstationID	The workstation ID used by the client.
Language	The national language corresponding to a subdirectory in *Sybase_root\locales*. The default is U.S. English.
ApplicationName	The name used by the DirectConnect Server to identify your application.
YieldProc	A numeric value that determines whether you can work in other applications when the DirectConnect server is busy. Valid values are: • 0 (peek and dispatch) causes the driver to check the Microsoft Windows message queue and send any messages to the appropriate application. • 1 (no yielding; the default) prevents you from working in other applications. • 3 (dispatch via Windows Yield function) gives control to the Windows kernel, which checks the message queue and sends messages to the appropriate applications. Sybase recommends that you use value 1.
OptimizePrepare	A value that determines whether stored procedures are created on the server for every call to SQLPrepare. Valid values are: • 0 causes stored procedures to be created for every call to SQLPrepare. This setting can decrease performance. • 1 causes the driver to create stored procedures only if the statement contains parameters. Otherwise, the statement is cached and executed directly at SQLExecute time. This is the initial default value. • 2 prevents the driver from creating stored procedures.
ArraySize	The number of rows the driver retrieves from the server for each fetch. This is not the number of rows given to the user. The default is 10 rows.

Property	Description
SelectMethod	A value that determines whether database cursors are used for select statements. Valid values are: • 0 allows database cursors to be used. This is the initial default. In some cases, performance degradation can occur when large numbers of sequential select statements execute because of the large amount of overhead associated with creating the cursors. • 1 allows select statements to execute directly without database cursors. The data source is limited to one active statement and one active connection.
ReuseFailed Cursor	A value that determines whether the server attempts to reuse a cursor that contains an error. Valid values are: • 0 causes the server to deallocate a cursor that contains an error and force the application to start over. • 1 causes the server to attempt to reuse a cursor containing an error. This is the default value.
PacketSize	A number that determines the number of bytes per network packet transferred from the database server to the client. The correct setting of this attribute can improve performance. Valid values are: • 0 directs the driver to use the default packet size as specified in the System 11 or DirectConnect Server configuration. This is the initial default configuration. • -1 directs the driver to compute the maximum allowable packet size on the first connect to the data source and save the value in the *ODBC.ini* file. • x (an integer from 1 to 10) directs the driver to set the packet size to x times 512 bytes. For example, a value of 6 sets the packet size at 6 times 512, or 3072 bytes. To use this connection attribute, you must configure the System 11 or DirectConnect server for a maximum network packet size greater than or equal to the value of this property. The ODBC specification specifies a connect option sql_packet_size that offers this same functionality. To avoid conflicts with applications that can set both the connection string attribute and the ODBC connect option, both functions are defined as mutually exclusive. If PacketSize is specified, you receive the message "Driver Not Capable" if you attempt to call sql_packet_size. If you do not set PacketSize, the driver accepts application calls to sql_packet_size.

Property	Description
Translate	An instruction that displays the Select Translator dialog box, from which you can perform a data translation from one character set to another. Choose the INTERSOLV OEM ANSI translator to translate data from the IBM PC character set to the ANSI character set.

5 Choose OK to write these values to the *ODBC.ini* file.

The values are now the default values when you connect to this data source.

6 To change the values, do one of the following:

- Change the defaults by reconfiguring the data source.
- Override the defaults by connecting to the data source using a connection string with alternate values.

Connecting to the data source

You can use one of two methods to connect to the data source:

- A logon dialog box
- A connection string

Using a logon dialog box

Some ODBC applications display a logon dialog box when you attempt to connect to a data source. In such cases, the data source name already is specified.

To use the logon dialog box, enter the information shown in the following table.

Table 8-3: Logon dialog box information

Entry	Instructions
Server Name	Enter the case-sensitive name of the server containing the database tables you want to access via DirectConnect, or select the name from the Server Name drop-down list, which displays the name you specified when you configured the data source.
Login ID	If required, enter your Login ID (case sensitive).
Password	If required, enter your system password (case sensitive).
Database	Enter the name of the database you want to access (case sensitive), or select the name from the Database drop-down list, which displays the name you specified when you configured the data source.

Using a connection string

If your application requires a connection string to connect to a data source, you must specify the data source name that tells the ODBC driver the *ODBC.ini* section to use for the default connection information. Optionally, you can specify attribute=value pairs in the connection string to override the *ODBC.ini* default values. The override values are not written to the *ODBC.ini* file.

You can specify either long or short names in the connection string. The connection string has the following syntax:

```
DSN=data_source_name{;attribute=value
[;attribute=value];...}
```

An example of a DirectConnect for z/OS connection string is as follows:

```
DSN=SYBDC TABLES;
SRVR=QESRVR;DB=PAYROLL;UID=JOHN;PWD=XYZZY
```

Long and short names for each attribute are shown in the following table.

Table 8-4: Connection string long and short names

Attribute long name	Short name
DataSourceName	DSN
ServerName	SRVR
LoginID	UID
Password	PWD
Database	DB
Language	LANG
Charset	CS
HostName	HOST
ApplicationName	APP
InterfacesFile	IFILE
ArraySize	AS
YieldProc	YLD
InitializationString	IS
ModifySQLStatement	MSS

Mapping datatypes

You can map DirectConnect for z/OS datatypes to standard ODBC datatypes using the lists in the following table.

Table 8-5: DirectConnect for z/OS and ODBC datatypes

DirectConnect for z/OS (DB2)	ODBC
binary	SQL_VARBINARY
bit	SQL_BIT
char	SQL_VARCHAR
datetime	SQL_TIMESTAMP
decimal	SQL_DECIMAL
float	SQL_FLOAT
image	SQL_LONGVARBINARY
int	SQL_INTEGER
money	SQL_DECIMAL
numeric	SQL_NUMERIC
real	SQL_REAL
smalldatetime	SQL_TIMESTAMP
smallint	SQL_SMALLINT
smallmoney	SQL_DECIMAL
sysname	SQL_VARCHAR
text	SQL_LONGVARCHAR
timestamp	SQL_VARBINARY
varbinary	SQL_VARBINARY
varchar	SQL_VARCHAR

ODBC conformance level

The DirectConnect for z/OS ODBC driver supports the minimum SQL grammar and the Core, Level 1, and Level 2 API functions in the following lists.

Core functions

- SQLAllocConnec
- SQLAllocEnv
- SQLAllocStmt
- SQLBindCol
- SQLBindParameters
- SQLCancel
- SQLColAttributes

- SQLConnect
- SQLDescribeCol
- SQLDisconnect
- SQLDrivers
- SQLError
- SQLExecDirect
- SQLExecute
- SQLFetch
- SQLFreeConnect
- SQLFreeEnv
- SQLFreeStmt
- SQLGetCursorName
- SQLLNumResultsCols
- SQLPrepare
- SQLRowCount
- SQLSetCursorName
- SQLTransact

Level 1 functions

- SQLColumns
- SQLDriverConnect
- SQLGetConnectOption
- SQLGetData
- SQLGetFunctions
- SQLGetInfo
- SQLGetStmtOption
- SQLGetTypeInfo
- SQLParamData
- SQLPutData
- SQLSetConnectOption

- SQLSetStmtOptions
- SQLSpecialColumns
- SQLStatistics
- SQLTables

Level 2 functions

- SQLBrowseConnect
- SQLColumnPrivileges
- SQLDataSources
- SQLDescribeParam
- SQLExtendedFetch (forward scrolling only)
- SQLForeignKeys
- SQLMoreResults
- SQLNativeSql
- SQLNumParams
- SQLParamOptions
- SQLPrimaryKeys
- SQLProcedureColumns
- SQLProcedures
- SQLSetPos
- SQLSetScrollOptions
- SQLTablePrivileges

APPENDIX A

Validating Connectivity Using cicsping and snaping

After you have established connectivity between the LAN and the z/OS mainframe, test the connectivity by using either the cicsping or snaping utility.

This appendix contains the following topics:

Using cicsping

The following sections cover these topics:

- Description of cicsping
- When to use cicsping
- How cicsping works
- Installing the cicsping utility
- Defining SYPG to CICS
- Syntax for cicsping
- Examples of using cicsping
- If you receive errors

Description of cicsping

The cicsping utility verifies that the TCP/IP connection between the z/OS mainframe and the server is configured and operating correctly. Sybase recommends running cicsping before you install Transaction Router Service (TRS) or DirectConnect Access Service Library (ACSLIB).

DirectConnect does not need to be active when you run cicsping. The mainframe component of cicsping, SYPG, does not use Open ServerConnect or MainframeConnect for DB2 UDB; the cicsping utility only tests the network connection between the machine where you are running DirectConnect on the LAN and CICS at the z/OS mainframe.

You can use cicsping to collectively test all CICS regions defined to TRS or to test connectivity to one region at a time.

When to use cicsping

To check connectivity between various parts of your network, use the cicsping utility at each of the following times:

- After you configure the TCP/IP connection between the DirectConnect Server and the mainframe
- After Open ServerConnect is installed (it loads SYPG)
- Before you start or configure any DirectConnect services

How cicsping works

The cicsping utility opens a TCP/IP connection to a CICS region. Then it starts the SYPG transaction on the mainframe and sends a message to SYPG. Finally, cicsping receives an “echo” response from SYPG when it executes successfully.

If your network is configured correctly, cicsping returns a short message. The last two lines should read:

```
ECHO DATA = abcdefghijklmnopqrstuvwxyz
        test completed in 1081ms
```

Installing the cicsping utility

The cicsping utility ships with Open ServerConnect. Instructions for defining these programs to the mainframe are listed in the following guides:

- Open ServerConnect for CICS *Installation and Administration Guide*
- MainframeConnect for DB2 UDB *Installation and Administration Guide*

Defining SYPG to CICS

For you to run cicsping at the server, the z/OS mainframe system programmer must define the cicsping transaction and program to the CICS region. Verify that this is done before you use the cicsping utility.

The cicsping transaction ID is SYPG, and the cicsping program name is SYGWCAI2.

Syntax for cicsping

Following is the syntax for the cicsping utility and details about each cicsping parameter. Parameters enclosed in square brackets are optional. The default is the *-Rregion_file* for the specific platform.

```
cicsping [-Hhostname -Nportnumber]
         [-Thost_tramsaction_id]
         [-Rregion_file
         [-Uuser_name [-Ppassword]]][-v]
```

The following table lists each parameter, the operating system for which it is valid, and a description of the parameter.

Table A-1: cicsping parameters

Parameter	Valid operating system	Description
-Hhostname	HP-UX IBM RISC System/6000 Sun Solaris	The name of the mainframe as defined in your NIS map or your /etc/hosts file. This is the same name you use in the *hostname* property of the exec sgw_addregion command of the Transaction Router Service or the ConnectionSpec1 value of a DB2 access service (see the Mainframe Connect DirectConnect for z/OS Option *User's Guide for Transaction Router Services* for more information). When you use the *-H* parameter, you must also use the *-N* parameter.
	Windows	The name of the mainframe as defined in your Domain Name Server (DNS) or your *%windr%\system32\drivers\etc\hosts* file. This is the same name you use in the *hostname* property of the exec sgw_addregion command of the Transaction Router Service or the ConnectionSpec1 value of a DB2 access service (see the Mainframe Connect DirectConnect for z/OS Option *User's Guide for Transaction Router Services* for more information). When you use the -H parameter, you must also use the -N parameter.
-Nportnumber	All	The number of the mainframe port where the CSKL (TCP/IP Listener) transaction listens. This is the same number you use in the *portnumber* property of the exec sgw_addregion command of the Transaction Router Service, or the ConnectionSpec2 value of a DB2 access service (see the DirectConnect *Transaction Router Service User's Guide* for more information). When you use the -N parameter, you must also use the -H parameter.
-Rregion_file (This parameter has no application to the DB2 Access Service Library)	HP-UX RISC/6000 Sun Solaris	The name of the TRS connection file. The default is *$SYBASE/$DSQUERY.reg*. You can use this parameter if you are running cicsping after you configure TRS. TRS example files: *$SYBASE/$SYBASE_ECON/server_name/cfg/ngreg.trstcp*. When you run *cicsping* before you configure TRS, omit this parameter and use the *-H* and *-N* parameters.
	Windows	The name of the TRS connection file. The default is *%SYBASE%\%DSQUERY%.reg*. You can use this parameter if you are running cicsping after you configure TRS. TRS example files: *%SYBASE%\$%SYBASE_ECON%\server_name\cfg\trstcp.ngreg*. When you run cicsping before you configure TRS, omit this parameter and use the -H and -N parameters.
-Uuser_name	All	The mainframe user ID of the client issuing the command. It must be in uppercase.

Parameter	Valid operating system	Description
-Ppassword	All	The mainframe password of the client issuing the command. You must use the -U parameter to use the -P parameter. It must be in uppercase.
-Thost_transaction_id	All	The name of the z/OS mainframe (host) transaction called by cicsping that runs the cicsping program SYGWCAI2. This program is required for cicsping to execute successfully. The default transaction, SYPG, is called if you run cicsping without the -T parameter and value. Use the -T parameter if the cicsping transaction has a name other than SYPG. The value of this parameter must be in uppercase.
-v	All	Use this parameter to display information about the current version of cicsping. When it executes with the -v parameter, cicsping does not run.

Examples of using cicsping

This section provides the following examples of how to use cicsping:

- Testing the connectivity between the z/OS mainframe and the server
- Testing all defined connections
- Using a new transaction ID

Note Since you set up only the TCP/IP communication environment in this guide, you test only the server-to-mainframe connectivity.

Testing the connectivity

Use cicsping after you configure the z/OS mainframe and the server and before you configure DirectConnect TRS or DB2 access services. You need to use the -H and -N parameters to test each region, one at a time, that you use when you configure TRS or ACSLIB.

The following example tests a single region located at host, listening on "BIGSYB" port number "3003." Replace these example parameter values with the correct hostname and port number values for your site in the following example:

```
cicsping -HBIGSYB -N3003
```

Testing all regions defined to TRS

If you configured TRS and you must run cicsping again, execute the command without the parameters. This tests all regions defined to TRS.

To test all defined regions:

- For HP-UX, IBM RISC System/6000, and Sun Solaris:

```
cicsping -R $SYBASE/SYBASE_ECON/server_name/cfg/ngreg.trstcp
```

- For Windows:

```
cicsping -R %SYBASE%\%SYBASE_ECON%\server_name\cfg\trstcp.ngreg
```

where *server_name* is the name of your DirectConnect Server.

Using a new transaction ID

The cicsping utility requires the SYGWCAI2 program, which is accessed through the default SYPG transaction.

Note The cicsping utility requires the SYPG transaction. You can rename the transaction.

If your system programmer changes the name of the cicsping mainframe transaction, run cicsping using the -T parameter to specify the new transaction ID.

In the following example, cicsping runs against a mainframe transaction that was renamed SYP2:

```
cicsping -TSYP2
```

Replace SYP2 in the example with your own transaction ID.

If you receive errors

Execute cicsping successfully and without errors before you configure TRS. An example of successful cicsping output is as follows:

```
Testing host bugguy, port 3011
        gethostbyname ok
        connect to CICS listener ok
        transaction started by listener ok
```

```
        test message sent ok
        Reply from host transaction:
 CICSTEST        : *** MODULE SYGWTCPS LOADED
SUCCESSFULLY ***
 VERSION STRING = 02/24/9411.45 SYGWCICx/3.0/370/CICS/1
 ECHO DATA = abcdefghijklmnopqrstuvwxyz
        test completed in 1081ms
```

If cicsping stops prematurely or fails to operate correctly, check for one or more of the following conditions:

- A mainframe configuration error. For example,

  "CICS TRANID - SYPG IS INVALID"

  This message means that the SYPG transaction is not installed in the CICS region.
- A TCP/IP network configuration error.
- A network outage problem.

You can determine where the configuration error or network outage occurred based on the type of error message cicsping returns or by the point at which it stops prematurely.

- If the error occurred at the mainframe, the system programmer or operator must make the necessary changes in the network configuration.
- If the error occurred at the server, the operating system administrator must make the necessary changes to the network.

Using snaping

The following sections cover these topics:

- Description of snaping
- When to use snaping
- What snaping does
- Installing the SYI1 utility
- Defining SYI1 to the z/OS mainframe
- Syntax for snaping
- Examples of using snaping

- If you receive errors

Description of snaping

The snaping utility verifies that the LU 6.2 connections between the z/OS mainframe and the server are configured and operating correctly. Sybase recommends running snaping before you install Transaction Router Service (TRS) or DirectConnect for DB2 Access Service Library.

DirectConnect TRS or ACSLIB programs do not need to be active when you run snaping. The z/OS mainframe component of snaping, SYI1, does not use Open ServerConnect or MainframeConnect for DB2 UDB. The snaping utility only tests the SNA support between the machine where you are running DirectConnect on the LAN and the transaction processing region at the z/OS mainframe.

You can use snaping to collectively test all LU 6.2 connections defined to TRS or to test a single LU 6.2 connection.

When to use snaping

To check connectivity between various parts of your network, use the snaping utility at the following times:

- After you set up LU 6.2 communications between the DirectConnect Server and the mainframe
- After Open ServerConnect is installed (it loads SYI1)
- Before you start or configure DirectConnect TRS or ACSLIB

What snaping does

The snaping utility opens an LU 6.2 conversation to a mainframe transaction processing region. Then, it starts the SYI1 transaction on the z/OS mainframe and sends a message to SYI1. Finally, snaping receives an "echo" response from SYI1 when it executes successfully.

If your network is configured correctly, snaping returns a short message. The last line should read:

```
Done.
```

```
Test complete.
```

Installing the SYI1 utility

The SYI1 utility ships with Open ServerConnect. Instructions for defining these programs to the mainframe are listed in the following guides:

- Mainframe Connect Client Option *Installation and Administration Guide* for CICS
- MainframeConnect for DB2 UDB *Installation and Administration Guide*

Defining SYI1 to the z/OS mainframe

For you to run snaping at the server, the z/OS mainframe system programmer defines the snaping transaction and program to the transaction processing region. Verify that this is done before you use the snaping utility.

The snaping transaction ID is SYI1, and the snaping program name is one of the following:

- SYGWCAI1 for CICS
- SYGWIAI1 for IMS

Syntax for snaping

This section shows the syntax for the snaping utility and provides details about each snaping parameter. Parameters enclosed in square brackets [ ] are optional. The default parameter is *-LConnection _file* for the specific platform.

For HP 9000/8xx, Sun Solaris 2.x, Windows:

snaping [-C*connection* -M*modename* -R*remote_lu*]
[-U*user_name* [-P*password*]]
[-L*connection_file*]
[-H*host_transaction_id*] [-v]

For RS/6000 AIX:

snaping [-C*connection* -M*Modename*]
[-U*user_name* [-P*password*]]
[-L*connection_file*]
[-T*host_transaction_id*] [-v]

Examples of using snaping

This section provides the following examples of how to use snaping:

- Testing the connectivity between the z/OS mainframe and the server
- Testing all defined connections
- Using a new transaction ID

Note Since you are setting up only the LU 6.2 communication environment, you test only the server to mainframe connectivity using this section.

Testing the connectivity

Use snaping after you configure the z/OS mainframe and the local SNA subsystem, and before you configure TRS or the ACSLIB. Use the -C, -M, and -R parameters to test each connection, one at a time, that you will use when you do configure TRS or the ACSLIB. The connections are defined in the server configuration and at the z/OS mainframe.

The following command tests the connection called “SYBLU01”. The mode is “SYBMODE,” and the remote LU is called “TESTCICS”.

Replace the italicized example parameter values with the correct connection, mode and remote_LU values for your site, in the following example:

- For HP-UX, Sun Solaris, Windows:

```
snaping -CSYBLU01 -MSYBMODE -RTESTCICS
```

- For IBM RISC System/6000:

```
snaping -CSYBLU01 -MSYBMODE
```

Testing all connections defined to TRS

If you configured TRS and you need to run snaping again, execute the command without the parameters. This tests all connections defined to TRS.

To test all defined connections, enter the following command:

- For HP-UX, AIX, and Sun Solaris:

```
snaping -L $SYBASE/$SYBASE_ECON/server_name/cfg/ngcid.trslu62
```

- For Windows:

```
snaping -L %SYBASE%\$SYBASE_ECON\%server_name\cfg\trslu62.cid
```

where *server_name* is the name of your DirectConnect Server.

Using a new transaction ID

The snaping utility requires the SYGWCAI1 program for CICS or the SYGWIAI1 program for IMS, which is accessed through the default SYI1 transaction.

Note The snaping utility requires the *SYI1* transaction. You can rename the transaction; however, this transaction must be used.

If your systems programmer changes the name of the snaping mainframe transaction, run snaping using the -T parameter to specify the new transaction ID.

In the following examples, snaping runs against an z/OS mainframe transaction that has been renamed "SYT2." Replace "SYT2" in the example with your own transaction ID.

- For HP-UX, AIX, Sun Solaris, and Windows:

```
snaping -TSYT2
load snaping -TSYT2
```

If you receive errors

Execute snaping successfully and without errors before you configure TRS or ACSLIB. An example of successful snaping output is:

```
Done.
Test complete.
```

If you do not receive this message, use the following sections to check for errors.

If snaping stops prematurely or fails to operate correctly, check for one or more of the following conditions:

- A mainframe configuration error
- An SNA network support configuration error
- A line outage problem

You can determine where the configuration error or line outage occurred based on the type of error message snaping returns or by the point at which it stops prematurely.

- If the error occurred at the z/OS mainframe, the system programmer or operator must make the necessary changes in network configuration.
- If the error occurred at the server, the communications specialist administrator must make the necessary changes to the SNA network support configuration.

APPENDIX B

DirectConnect Utilities

To simplify the execution of DirectConnect on multiple platforms, Sybase provides utilities to start a server, create and start a new server, and migrate files from one server to another.

These utilities are c shell scripts (on UNIX) and batch files (on Windows) that can be found and kept in the *$SYBASE/$SYBASE_ECON/bin* (UNIX) or *%SYBASE%\%SYBASE_ECON%\bin* (Window) directory. To run properly, the scripts must be kept in their original directory. It is from this directory that the utilities can find the paths to the other files they need to perform their tasks.

This appendix contains the following topics:

Creating and starting a DCDirector server

DirectConnect version 12.6 provides a new feature that allows DirectConnect Manager to connect to a “directing” server, called a DCDirector, that is capable of creating, starting, and stopping DirectConnect servers. To create a new DCDirector for an installation, use the DCDirector script (on UNIX) or batch file (on Windows).

This script creates a new server with the default name “DCDirector” and sets the port number it listens on to a default port of 7711. The script then starts the server. No parameters are required or allowed, just enter:

```
DCDirector
```

You may change the default server name and default port number. If you need to use different values to identify the directors, you can use a text editor to modify the DCDirector batch file by assigning different values.

DCDirector utility

This utility creates a default DCDirector server in the installation area. This script or batch file does not accept any parameters and uses the server name "DCDirector," and the port 7711 by default. You can change these values with a text editor if you prefer to use different values.

Usage DCDirector

Example DCDirector

Creating and starting a new server

Creating a new DirectConnect server is not particularly difficult but it often happens that one will forget to add the necessary entries in the *interfaces* (UNIX) or *sql.ini* (Windows) file. AddServer is a utility that creates the necessary entries in the *interfaces* or *sql.ini* file before starting the DirectConnect server. AddServer requires two parameters:

- The name of the new server
- The port number for the server to listen on

AddServer utility

This utility is a simple way to create a server entry in the *interfaces* (UNIX) or *sql.ini* (Windows) files, and then run the "direct" executable using the -N option. This utility makes changes to the *interfaces* or *sql.ini* file but it does not make any checks to see if the *servername* or the *port number* are already being used.

Usage AddServer <servername> <port number>

Example

```
AddServer srvrname 1133
```

Starting a server

In the past, you started a DirectConnect server using a command that used the direct executable itself. For example, the command, direct –Ssrvname, was the standard way to start an existing DirectConnect server. This method is not recommended because it has some limitations: For this command to work properly, all of the appropriate environment variables need to be set properly before attempting to execute the command. Also, if multiple installs of DirectConnect exist on a single machine, each installation will need its own environment.

To provide some help with this, a shell script or a batch file is provided with DirectConnect version 12.6 that sets the installation-specific variables before executing the direct –Ssrvname command. This script, called DCStart, requires that all the non-Sybase variables be set properly, and ensures that the environment variables that are specific to a DirectConnect installation are all correctly set.

To stop a server, see "Stopping the DirectConnect server" section in Chapter 5, "Installing DirectConnect Manager."

DCStart utility

This utility is similar to using the direct executable. DCStart will automatically "source" the appropriate *DC_SYBASE.csh* (UNIX) file or run the appropriate *DC_SYBASE.bat* (Windows) file to ensure that all the appropriate Sybase-specific environment variables are set properly. This is the required method to start a DirectConnect server, for one or multiple versions, since DirectConnect no longer sets the system environment.

Usage

DCStart <Parameters>

Example

```
DCStart -Ssrvname
```

Glossary

access service — The named set of properties, used with a DirectConnect Access Service Library, to which clients connect. Each DirectConnect Server can have multiple services.

access service library — A component of DirectConnect. A service library that provides access to non-Sybase data contained in a database management system or other type of repository. Each such repository is called a "target." Each access service library interacts with exactly one target and is named accordingly. See also **service library**.

ACSLIB — The DB2 Access Service Library.

administrative service library — A service library that provides remote management capabilities and server-side support. It supports a number of remote procedures (invoked as RPC requests) that enable remote DirectConnect management. See also **remote procedure call**, **service library**.

Adaptive Server Enterprise — The server in the Sybase Client-Server architecture. It manages multiple databases and multiple users, tracks the actual location of data on disks, maintains mapping of logical data description to physical data storage, and maintains data and procedure caches in memory.

ADMLIB — See **administrative service library**.

APPC — See **advanced program-to-program communication**.

application program interface — A functional interface, supplied by an operating system or other licensed program, that allows an application program written in a high-level language to use specific data, functions of the operating system, or the licensed program.

ASCII — See **American Standard Code for Information Interchange**.

ASE — See **Adaptive Server Enterprise**.

ASE/CIS Adaptive Server Enterprise/Component Integration Services replaces OmniConnect. An add-on product for Sybase SQL Server that provides a Transact-SQL interface to external data sources, including host data files and tables in other database systems. OmniConnect replaces "OmniSQL Gateway" and "OmniSQL Server."

catalog stored procedure A stored procedure that provides information about tables, columns, and authorizations. It is used in SQL generation and application development.

CICS See **Customer Information Control System**.

client In client/server systems, the part of the system that sends requests to servers and processes the results of those requests. See also **client/server**. Compare with **server**.

client application Software that is responsible for the user interface, including menus, data entry screens, and report formats. It also is an application that sends requests to another application that acts as a server. See also **client/server**.

Client-Library A library of routines that is part of Open ClientConnect. See also **Open ClientConnect**.

client services applications A customer-written CICS program initiated on the host that uses the Sybase API to invoke MainframeConnect for DB2 UDB as a client to DirectConnect or SQL Server. Compare with **remote stored procedure**.

code page An assignment of graphic characters and control function meanings to all code points.

configuration file A file that specifies the characteristics of a system or subsystem.

configuration set A section into which service library configuration files are divided.

CSA See **client services application**.

CSP See **catalog stored procedure**.

CT-Library See **Client-Library**.

database management system A computer-based system for defining, creating, manipulating, controlling, managing, and using databases.

DBMS See **database management system**.

DirectConnect — A Sybase Open Server application that provides access management for non-Sybase databases, copy management, and remote systems management. Each DirectConnect consists of a server and one or more service libraries to provide access to a specific data source.

DirectConnect Manager — A Sybase Windows application that provides remote management capabilities for DirectConnect products. These capabilities include starting, stopping, creating, and copying services.

DirectConnect for z/OS — A Sybase LAN-based solution that communicates with mainframe host components. It incorporates the functionality of the MDI Database Gateway and the Sybase Net-Library and includes LU 6.2 and TCP/IP support.

DirectConnect Server — The component that provides general management and support functions (such as log file management) to service libraries.

DirectConnect Service — The named set of properties, used with a DirectConnect Service Library, to which clients connect. Each DirectConnect can have multiple services.

DirectConnect Service Library — The component that provides a set of functions within the DirectConnect Server environment.

dynamic link library — A file containing executable code and data bound to a program at load time or run time, rather than during linking. The code and data in a dynamic link library can be shared by several applications simultaneously.

Extended Binary-Coded Decimal Interchange Code (EBCDIC) — A coded character set of 256 8-bit characters.

globalization — The combination of internationalization and localization. See also **internationalization**, **localization**.

graphical user interface (GUI) — A type of computer interface consisting of a visual metaphor of a real-world scene, often of a desktop. Within that scene are icons, representing actual objects, that the user can access and manipulate with a pointing device.

InstallShield — The Sybase DirectConnect server installation program. An interactive program that is used to install DirectConnect products and programs that support them.

interfaces file — An operating system file that determines how the host client software connects to a Sybase product. The file must be available on each machine that connects to DirectConnect or other Sybase products.

internationalization The process of extracting locale-specific components from the source code and moving them into one or more separate modules, making the code culturally neutral so it can be localized for a specific culture. See also **globalization**. Compare with **localization**.

library A named disk area that can contain programs and related information. A library consists of different sections, called library members.

localization The process of preparing an extracted module for a target environment, in which messages are displayed and logged in the user's language; numbers, money, dates, and time are represented using the user's cultural convention; and documents are displayed in the user's language. See also **globalization**. Compare with **internationalization**.

logical unit A type of unit that enables end-users or programs to gain access to network resources and communicate with each other.

logical unit 6.2 A logical unit that supports general communication between programs in a distributed processing environment. See also **advanced program-to-program communication**.

LU See **logical unit**.

LU 6.2 See **logical unit 6.2**.

Mainframe Connect A Sybase suite of products that provide access to mainframe data.

MainframeConnect for DB2 UDB A Sybase mainframe solution that provides dynamic access to DB2 data. It replaces the OmniSQL Access Module for DB2 and the functionality of the MDI Access Server.

Net-Gateway A Sybase product that provides communication between a mainframe and a LAN server. Net-Gateway is the "ancestor" of the DirectConnect Transaction Router Service. See also **Transaction Router Service**.

Net-Library A Sybase product that lets PC applications become clients of SQL Server or Open Server. See also **client**, **Open Server**, **SQL Server**.

Network ID A customer-selected name or a registered name that uniquely identifies a specific subnetwork.

Online Transaction Processing A system that supports database maintenance through an interactive user interface. The user usually requires a response time of less than three seconds.

Open Client A Sybase product that provides customer applications, third-party products, and other Sybase products with the interfaces required to communicate with Open Client and Open Server applications.

Open ClientConnect — A Sybase product that allows mainframe clients to send requests to SQL Server, Open Server, MainframeConnect for DB2 UDB, and Open ClientConnect using Client-Library. Open ClientConnect provides capability for the mainframe to act as a client to LAN-based resources.

Open Server — A Sybase product that provides the tools and interfaces required to create a custom server.

Open ServerConnect — A Sybase product that provides capability for programmatic access to mainframe data. It allows workstation-based clients to execute customer-written mainframe transactions remotely.

protocol — A set of rules that governs the behavior of computers communicating on a network.

remote procedure call — A stored procedure executed on a different DirectConnect Server from the one onto which a user is logged or on which the initiating application resides.

remote stored procedure — A customer-written CICS program that resides on the mainframe and communicates with MainframeConnect for DB2 UDB. Compare with **client services application**.

remote systems management — A feature that allows a system administrator to manage multiple DirectConnect Servers and multiple services from a client.

RPC — See **remote procedure call**.

RSP — See **remote stored procedure**.

server — A functional unit that provides shared services to workstations over a network. See also **client/server**. Compare with **client**.

service — A functionality available to DirectConnect applications. It is the pairing of a service library and a set of specific configuration properties.

service library — A set of configuration properties that determine service functionality. Examples of service libraries include access service libraries, administrative service libraries, and transaction router service libraries. See also **access service library**, **administrative service library**, **Transaction Router Service Library**, **transfer service library**.

service name redirection — A type of service name resolution that allows a system administrator to map alternative connections to services. See also **service name resolution**. Compare with **direct resolution**.

service name redirection file — The default name of the file used for the service name redirection feature. See also **service name redirection**.

SNA See **Systems Network Architecture**.

SNRF See **service name redirection file**.

sql.ini The interfaces file containing definitions for each DirectConnect Server to which a workstation can connect. The file must be on every client machine that connects to SQL Servers. See also **interfaces file**, **libtcl file**.

Systems Network Architecture (SNA) An IBM proprietary plan for transmitting information units through networks and controlling network configuration and operation.

TCP/IP See **Transmission Control Protocol/Internet Protocol**.

Transaction Router Service The DirectConnect program that accepts requests from workstation-based clients and routes them to Open ServerConnect.

Transaction Router Service library A service library that facilitates access to remote transactions, allowing customers to execute thousands of transactions from virtually any mainframe data source. See also **service library**.

TRS See **Transaction Router Service**.

TRS Library See **Transaction Router ServiceLlibrary**.

UNIX An operating system developed by Bell Laboratories that allows for multiple concurrent programs and users.

utility program A computer program that supports computer processes, including diagnostic programs, trace programs, and sort programs.

z/OS An IBM operating system that runs on z/OS mainframes.

Index

Symbols

A

C

D

E

F

H

I

L

M

N

O

P

R

S

T

U

V

W

Z