

SYBASE®

Installation and Administration Guide

Mainframe Connect™ Server Option

15.0

[IBM IMS and MVS]

DOCUMENT ID: DC34368-01-1500-01

LAST REVISED: July 2007

Copyright © 1989-2007 by Sybase, Inc. All rights reserved.

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

To order additional documents, U.S. and Canadian customers should call Customer Fulfillment at (800) 685-8225, fax (617) 229-9845.

Customers in other countries with a U.S. license agreement may contact Customer Fulfillment via the above fax number. All other international customers should contact their Sybase subsidiary or local distributor. Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase trademarks can be viewed at the [Sybase trademarks page](http://www.sybase.com/detail?id=1011207) at <http://www.sybase.com/detail?id=1011207>. Sybase and the marks listed are trademarks of Sybase, Inc. ® indicates registration in the United States of America.

Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names used herein may be trademarks or registered trademarks of their respective companies.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

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

Contents

About This Book	vii
CHAPTER 1 Understanding the Server Option.....	1
What is the Server Option?	1
Architecture	2
Functionality	2
Requests in a three-tier environment	2
CHAPTER 2 Planning Your Installation	5
Choosing a network driver	5
IMS and MVS environment	5
Planning the installation	6
Installation media	6
Pre-installation tasks	6
Task list	7
CHAPTER 3 Installation and Configuration	11
Installing and configuring the Server Option	11
Post-installation steps for IMS.....	18
Libraries and samples	19
CHAPTER 4 Security	21
Sybase components and security	21
Client workstation	21
ASE	22
TRS	22
IMS TM	23
Vendor SNA support software	24
Security for APPC/IMS	24
LU security	24
Conversational security	24
Implementing conversational security	25

	APPC/MVS definitions.....	25
	RACF definitions	25
	SNA definitions.....	26
	Example of conversation level security	26
	External security systems	27
CHAPTER 5	Tracing and Accounting.....	29
	Tracing	29
	Server Option trace functions.....	29
	Trace log	30
	Using the tracing facility	32
	Accounting	37
	Server Option accounting functions	37
	Accounting log.....	38
APPENDIX A	Customization Options	41
	Overview	41
	Customizing global options (SYGWM CST).....	42
	Using the IBM z/OS conversion environment and services	45
	Customizing mainframe character set conversion options (SYGWMCXL)	45
	Overriding the supplied SBCS translation tables	46
	Defining new SBCS translation tables.....	47
	Defining new character set entries.....	50
	Customizing dynamic network drivers (SYGWDRIV).....	51
	CICS network drivers	51
	Customizing the TCP/IP driver (SYGWHOST)	53
	Macro formats	53
	Macro parameters	53
	Defining license keys (SYGWLKEY)	54
	Building a global customization module (SYGWXCPH).....	55
APPENDIX B	Translation Tables	57
	Understanding the ASCII-EBCDIC and EBCDIC-ASCII translation tables	57
	Default ASCII_8 translation tables	59
	ASCII_8, ASCII-to-EBCDIC translation table	60
	ASCII_8, EBCDIC-to-ASCII translation table	61
	Default ISO_1 translation tables	62
	ISO_1 ASCII-to-EBCDIC translation table	62
	ISO_1 EBCDIC-to-ASCII translation table	63
	Default cp437 (code page 437) translation tables.....	64

	cp437 ASCII-to-EBCDIC translation table.....	65
	cp437 EBCDIC-to-ASCII translation table.....	66
	Default cp850 (code page 850) translation tables.....	67
	cp850 ASCII-to-EBCDIC translation table.....	67
	cp850 EBCDIC-to-ASCII translation table.....	68
APPENDIX C	Troubleshooting.....	69
	Coordinating troubleshooting efforts	69
	Processing flow and requirements	69
	Process flow during attention sequences.....	72
	Browse applications	74
	Where to start troubleshooting	74
	Common problems and suggested solutions.....	75
	Configuration errors.....	75
	Mainframe network operational failure	77
	Network session or line failures.....	78
	Troubleshooting at each component.....	78
	Server Option support	78
	TRS support	79
	DirectConnect for z/OS Option communications with the mainframe	
	79	
	Gateway-Library support	80
	Glossary	83
	Index	99

About This Book

The Mainframe Connect™ Server Option for IMS and MVS *Installation and Administration Guide* describes how to install and configure the Server Option for IMS and MVS. It also addresses system administration.

Note If you want to go directly to the installation instructions, skip to Chapter 2, “Planning Your Installation.”

Audience

The guidelines and instructions in this book are intended for those who install, configure, and maintain Sybase® mainframe components on an IBM z/Series mainframe computer. This book refers to anyone performing these tasks as the Server Option administrator.

To use this book, you should have a working knowledge of system administration for your environment.

How to use this book

This table shows how this book is organized:

To	See
Understand the Server Option	Chapter 1, “Understanding the Server Option”
Plan the Server Option installation	Chapter 2, “Planning Your Installation”
Install the Server Option	Chapter 3, “Installation and Configuration”
Understand Server Option security	Chapter 4, “Security”
Set up tracing and accounting	Chapter 5, “Tracing and Accounting”
Customize the Server Option	Appendix A, “Customization Options”
Reference translation tables	Appendix B, “Translation Tables”
Troubleshoot problems with client access to data	Appendix C, “Troubleshooting”

Related documents

To install and use the Server Option, you may need to refer to the following documentation:

- Mainframe Connect Server Option *Programmers Reference for PL/I*
- Mainframe Connect Server Option *Programmers Reference for COBOL*

-
- Mainframe Connect Server Option *Programmers Reference for Remote Stored Procedures*
 - Mainframe Connect Client Option *Programmers Reference for PL/I*
 - Mainframe Connect Client Option *Programmers Reference for COBOL*
 - Mainframe Connect Client Option *Programmers Reference for C*
 - Mainframe Connect Client Option *Programmers Reference for Client Services Applications*
 - Mainframe Connect Client Option and Server Option *Messages and Codes*
 - Mainframe Connect DirectConnect for z/OS Option *Installation Guide*
 - Mainframe Connect DirectConnect for z/OS Option *Users Guide for DB2 Access Services*
 - Mainframe Connect DirectConnect for z/OS Option *Users Guide for Transaction Router Services*
 - Enterprise Connect Data Access and Mainframe Connect *Server Administration Guide for DirectConnect*

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.

Conventions

The Server Option uses 8-character function names; other versions of Server-Library use longer names. This book uses the long version of Server-Library names with this exception: the 8-character version is used in syntax statements. For example, in a syntax statement, "CTBCMDPROPS" is written "CTBCMDPR." You can use either version in your code.

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

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

Table 1 explains the syntax conventions used in this guide.

Table 1: Syntax conventions

Symbol	
()	When you see parentheses, include them 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.
	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.

Table 2 shows the style conventions used in this guide:

Table 2: Style conventions

This type of information	Looks like this
Gateway-Library function names	TDINIT, TDRESULT
Client-Library™ function names	CTBINIT, CTBRESULTS
Other executables (DB-Library™ routines, SQL commands) in text	the dbrpcparam routine, a select statement
Directory names, path names, and file names	<i>/usr/bin directory, interfaces file</i>
Variables	<i>n bytes</i>
Datatypes	<i>datetime, float</i>
Sample code	<i>01 BUFFER PIC S9(9) COMP SYNC</i>
User input	<i>01 BUFFER PIC X(n)</i>
Client-Library and Gateway-Library function argument names	<i>BUFFER, RETCODE</i>
Names of objects stored on the mainframe	<i>SYCTSAA5</i>
Symbolic values used with function arguments, properties, and structure fields	<i>CS-UNUSED, FMT-NAME, CS-SV-FATAL</i>
Client-Library property names	<i>CS-PASSWORD, CS-USERNAME</i>
Client-Library and Gateway-Library datatypes	<i>CS-CHAR, TDSCHAR</i>

All other names and terms are in regular typeface.

Accessibility features

This document is available in an HTML version that is specialized for accessibility. You can navigate the HTML with an adaptive technology such as a screen reader, or view it with a screen enlarger.

The HTML documentation has been tested for compliance with U.S. government Section 508 Accessibility requirements. Documents that comply with Section 508 generally also meet non-U.S. accessibility guidelines, such as the World Wide Web Consortium (W3C) guidelines for Web sites.

Note You might need to configure your accessibility tool for optimal use. Some screen readers pronounce text based on its case; for example, they pronounce ALL UPPERCASE TEXT as initials, and MixedCase Text as words. You might find it helpful to configure your tool to announce syntax conventions. Consult the documentation for your tool.

For information about how Sybase supports accessibility, see Sybase Accessibility at <http://www.sybase.com/accessibility>. The Sybase Accessibility site includes links to information on Section 508 and W3C standards.

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.

Topic	Page
What is the Server Option?	1
Architecture	2
Functionality	2

What is the Server Option?

The Server Option is an application programming environment (API) that allows you to create mainframe applications that Sybase client applications can use. Server Option applications can retrieve and update data stored in mainframe resources such as:

- DB2 UDB and other relational database management systems (RDBMSs)
- DL/1 databases
- Temporary Storage (TS) queues
- Transient Data (TD) queues
- VSAM files

The Server Option is available for CICS, and for IMS TM and MVS.

Note For information on how the Server Option functions in the CICS environment, see the *Mainframe Connect Server Option for CICS Installation and Administration Guide*.

Architecture

The Server Option runs on an IBM z/Series or plug-compatible mainframe computer. The Server Option uses LU 6.2, CPI-C, or IBM TCP/IP communications with a host transaction processor, such as MVS, as a communications front end.

Note The Server Option for IMS and MVS functions only in a three-tier (gateway-enabled) environment.

When installing and using the Server Option, follow the instructions in this book for your environment.

Functionality

Server Option applications can receive requests from LAN clients and Client Option applications in a three-tier environment using the DirectConnect for z/OS Option DB2 access service or Transaction Router Service (TRS).

Requests in a three-tier environment

In the Server Option network configuration using a three-tier (gateway-enabled) SNA environment, the DirectConnect for z/OS Option accepts requests from LAN-based clients and routes them to the appropriate server.

Server Option applications receive requests from LAN clients through either of the following DirectConnect for z/OS Option components:

- DB2 access service
- TRS

DB2 access service

An access service is a logical server application, used with an access service library, that allows a LAN client to communicate with Server Option applications. Each DirectConnect for z/OS Option server can have multiple DB2 access services.

For more information about access services, see the Mainframe Connect DirectConnect for z/OS Option *Users Guide for DB2 Access Services*.

TRS

TRS allows Sybase clients running on workstations and sharing a local area network (LAN) to access mainframe data and applications. The TRS listener waits for and accepts client requests and routes them to the mainframe, using transaction and connection information the DirectConnect for z/OS Option administrator provides during configuration.

TRS treats all client requests like remote procedure calls (RPCs), mapping each request to a specific mainframe transaction. When it receives a client request, TRS invokes the corresponding mainframe transaction. The transaction processor runs the transaction and returns results to TRS, which forwards the results to the requesting client.

For details, see the Mainframe Connect DirectConnect for z/OS Option *Users Guide for Transaction Router Services*.

TRS configuration in a three-tier architecture

The mainframe and TRS configuration parameters must be coordinated to permit communication with one another. When configuring a mainframe region to communicate with TRS, coordinate the following mainframe configuration values with TRS:

- For LU 6.2:
 - MVS connection and session definitions
 - Virtual Telecommunications Access Method (VTAM)
 - Network Control Program (NCP)
 - SNA, using your TRS platform SNA support program
- For TCP/IP:
 - TCP/IP for z/OS port definitions
 - Sybase listener configuration values

Note The configuration values are provided in the Mainframe Connect DirectConnect for z/OS Option *Users Guide for Transaction Router Services*.

A Sybase network can also connect multiple clients and multiple servers, running in one or more MVS regions on one or more mainframes. MVS accepts simultaneous client requests through TRS and assigns them to host resources based on availability. MVS also provides transaction management services, including the handling of synchronization points, rollbacks, commits, and recovery.

Note Client applications can access the Server Option with TCP/IP in a three-tier environment by using IBM IMS Connect.

Topic	Page
Choosing a network driver	5
Planning the installation	6

Choosing a network driver

The Server Option provides added flexibility and easy installation for sites configured to run SNA, CPI-C, and TCP/IP network protocols by supporting the concurrent use of multiple network drivers. Programs can invoke network drivers from the same Server Option and Client Option common code base, and the appropriate network driver loads dynamically during program execution.

The choice of a network driver depends on your network type and operating environment.

IMS and MVS environment

The following drivers are supported in the IMS or MVS environment:

- CPI-C for an SNA network
- LU 6.2 for an SNA network

Table 2-1 indicates which drivers can be used by the Server Option for IMS and MVS in a three-tier environment:

Table 2-1: Drivers to use in a three-tier environment

Driver	IMS	MVS
CPICIMS	X	
CPICMVS		X
LU62IMS	X ¹	

1. Client applications may access the Server Option for IMS if IBM IMS Connect listens at the mainframe end. In this case, the implicit mode driver LU62IMS is used.

Note The Server Option supports TCP/IP access to IMS using the IBM IMS Connect TCP/IP listener and the Mainframe Connect DirectConnect for z/OS Option. For information on enabling TCP/IP access to IMS, see “Post-installation steps for IMS” in Chapter 3, “Installation and Configuration.”

Planning the installation

This section includes the following topics:

- Installation media
- Pre-installation tasks

Installation media

The Server Option is distributed on CD or in downloadable form.

Note For information on obtaining the latest EBFs for the Server Option, see the *Release Bulletin* for your product.

Pre-installation tasks

Installation requires completing these pre-installation tasks, which are explained in the following subsections. You should skip those tasks that do not pertain to the options you have chosen to install.

1. Verify the platforms, components, and distributed software
2. Verify the space requirements
3. Determine JCL and system information
4. Determine CICS and DB2 UDB information
5. Determine compiler information
6. Determine Server Option information
7. Determine ftp information
8. Plan the security requirements
9. Identify the change control requirements
10. Back up the release libraries (upgrades only)
11. Determine the library names
12. Verify the connectivity

Task list

Perform these tasks prior to installation.

1. Verify the platforms, components, and distributed software

See the Mainframe Connect Server Option for IMS and MVS *Release Bulletin*.

2. Verify the space requirements

See the current release bulletin to verify the space required install the Server Option.

Note You can duplicate various libraries to support the configuration of your environment. However, duplication requires additional space.

3. Determine JCL and system information

Determine the following information to be used in the installation procedure:

- *JCL jobcard values* – used in the final installation jobs run in TSO.

- *High-level qualifier* – used as a prefix for data sets generated during installation.
- *Volume serial number* – indicates where generated data sets are cataloged.
- *Unit parameter value* – indicates the device requirements for cataloging generated data sets.
- *Work unit* – for the use of temporary work data sets.
- *Customer CICS, IMS, and MVS LOADLIBs* – pre-cataloged partitioned data sets (PDSs) or partitioned data sets extended (PDSE), into which configuration modules and sample programs are to be linked.

4. Determine CICS and DB2 UDB information

If you intend to install a component that uses CICS or DB2 UDB, determine this information:

- *High-level qualifier for CICS system data sets.*
- *RDO data set name* – the name of the CICS RDO (DFHCSD) containing the application resource definitions used by your CICS region.
- *RDO group list* – the RDO group list used by your CICS region when executing an initial start.
- *CICS region APPLID* – the VTAM APPLID for your CICS region.
- *DB2 system data sets high-level qualifier* – the high-level qualifier used for DB2 system data sets.
- *DB2 exit data set name (DSN).*
- *DB2 DSN.*

5. Determine compiler information

If you intend to install an API component, determine this information:

- *LE370 high-level qualifier* – used for the Language Environment 370.
- *COBOL compiler name* – the module used to execute COBOL in your environment.
- *COBOL compiler loadlib* – the system LOADLIB where your COBOL compiler module resides.
- *PLI compiler name* – the module used to execute PLI in your environment.

- *PLI compiler loadlib* – the system LOADLIB where your PLI compiler module resides.
- *C compiler data sets high-level qualifier*.
- *TCP/IP data sets high-level qualifier*.

6. Determine Server Option information

Determine this information for use in installing the Server Option:

- *TCP address space name*.
- *Remote server name* – the name by which your Server Option applications will refer to the remote server.
- *Remote TCP host name* – the DNS name for the remote server.
- *Remote server TCP host port* – the TCP/IP port used by the remote server.

7. Determine ftp information

To establish an FTP connection to your mainframe, determine this information:

- *User ID*.
- *Password*.
- *Mainframe host name*.
- *Control port number* – the listener port used by your mainframe FTP server, usually 21.
- *TCP address space name*.
- *Volume serial number or unit* – specify either a volume serial number (VOL=SER) and unit assignment for FTP to use, or allow FTP to use default values.
- *Log path name* – indicates where FTP log information will be written.

8. Plan the security requirements

Review your security requirements with your security administrator. You may also need to consult with your network administrator.

9. Identify the change control requirements

Create a change control plan that includes:

- All the tasks that need to be considered for installation
- The different groups that need to be aware of the environment change, for example, field personnel and groups involved in administering applications, zSeries, security, change control, and scheduling
- A schedule, including cut-off dates for specific tasks

10. Back up the release libraries (upgrades only)

If you are upgrading an existing release of the Server Option, Sybase strongly recommends that you back up the entire set of release libraries before beginning this installation.

11. Determine the library names

The shipped library names are unique for this release. If you are upgrading, decide whether you want to use your current library names. If this is a new release, you still might want to consider how to name the files.

You do not need to remove previous releases from your Sybase libraries because default names shipped with this release create an entirely unique set of release libraries. However, you can change them, based on naming standards at your site.

Note When the upgrade is complete and tested, be sure to replace the old LOADLIB name or add the new LOADLIB name to the DFHRPL concatenation for the selected CICS regions, as described in the installation instructions.

If you plan to continue using the old Sybase library names, delete all members before installing the new ones with the new version.

12. Verify the connectivity

Use the standard LAN ping utility to ensure connectivity between z/OS and the workstation running Adaptive Server® Enterprise (ASE).

Topic	Page
Installing and configuring the Server Option	11
Post-installation steps for IMS	18
Libraries and samples	19

Installing and configuring the Server Option

Note Be sure you completed the tasks in Chapter 2, “Planning Your Installation.”

Licensing information

The Server Option for IMS and MVS requires a permanent authorization key. However, Sybase includes a temporary key, which is valid for 30 days, within the order at installation time. To avoid interrupting your operations, call Customer Service at 1-800-8Sybase (1-800-879-2273), select Option 3, then select Option 3 again, and request a permanent key.

When speaking with Customer Service, please have the following information ready:

- Product name
- Order number
- For the machine you are using:
 - Serial number
 - Machine type
 - Model number

- A valid e-mail address

Note Please allow seven business days for the key to be generated and sent to you.

The two procedures in this section describe the installation steps you need to install all Mainframe Connect options from the InstallShield wizard and to complete the installation for the Server Option for IMS and MVS. Skip those installation steps that do *not* pertain to the options you have chosen to install.

Note The InstallShield wizard runs only on Windows.

❖ **To install from the InstallShield wizard**

- 1 Start the InstallShield wizard from CD by executing *setupwin.exe*, which is in the root directory.

The dialog box that appears displays the options available for installation. Click Next and Back to navigate through the wizard. To cancel the installation, click Cancel. Click next to proceed.

- 2 Accept the terms of the user-license agreement by selecting your country in the drop-down list and selecting the option to indicate that you agree with the terms. Click Next.
- 3 Select the components you want to install and click Next.

Note If you are installing the Server Option for CICS or the DB2 UDB Option for CICS, the Server Option for CICS runtime component will be automatically selected as you go to the next window.

- 4 Provide the following JCL and system information:

- *JCL Line 1-3* – a valid jobcard used to run the final installation jobs in TSO.
- *High Level Qualifier* – used as a prefix for all data sets generated during installation.
- *Volume* – the volume serial number that indicates where generated data sets are cataloged.
- *Unit* – unit parameter value that indicates the device requirements for cataloging generated data sets.

- *Work Unit* – for the use of temporary work data sets.
- *Customer CICS, IMS and MVS Loadlibs* – pre-cataloged partitioned data sets (PDSs) or partitioned data sets extended (PDSE) into which configuration modules and sample programs are to be linked. For CICS, this data set should be in the DFHRPL configuration ahead of other Sybase libraries.

Click Next.

- 5 If you are installing an option that uses CICS, DB2 UDB, or IMS, provide the following information where it applies. Otherwise, skip to the next step.
 - *CICS system datasets hlq* – the high-level qualifier for CICS system data sets that is used to locate *SDFHLOAD* and other CICS libraries.
 - *RDO Dataset* – the name of the CICS RDO (DFHCSD) containing the application resource definitions used by your CICS region.
 - *RDO Group List* – the RDO group list used by your CICS region when executing an initial start.
 - *CICS Region Applid* – the VTAM APPLID for your CICS region.
 - *DB2 system datasets hlq* – the high-level qualifier that is used for DB2 system data sets.
 - *DB2 Exit Dataset* – the name of the DB2 exit data set used by your DB2 region.
 - *DB2 DSN Name:* – the data set name (DSN) of your DB2 region.
 - *IMS datasets hlq* – the high-level qualifier for IMS system data sets that is used to locate IMS libraries.

Click Next.

- 6 If you are installing an API component, provide the following compiler information, which is used to configure JCL for compiling sample programs. Otherwise, skip to the next step.
 - *LE/370 datasets hlq* – used for the Language Environment 370 and is used here to locate data sets like *CEELKED*.
 - *COBOL Compiler Name* – the module used to execute COBOL in your environment.
 - *COBOL Compiler Loadlib* – the system LOADLIB in which your COBOL compiler module resides.

- *PLI Compiler Name*: The PLI compiler name is the module used to execute PLI in your environment.
- *PLI Compiler Loadlib* – the system LOADLIB in which your PLI compiler module resides.
- *C compiler datasets hlq* – the high-level qualifier used for C that is used to locate data sets like *SBCCMP*.
- *TCP/IP datasets hlq* – used to locate data sets like *SEZATCP*.

Click Next.

- 7 If you are installing the Client Option for CICS, provide the following information for configuring a host connection definition for the Client Option. Otherwise, skip to the next step.

- *TCP Address Space Name* – the name of your TCP/IP region.
- *Server Name* – the name by which your Client Option applications refers to the remote server.
- *Server TCP Host Name* – the DNS name for the remote server.
- *Server TCP Host Port* – the TCP/IP port used by the remote server.

Click Next.

- 8 If you are installing the Server Option for CICS or the DB2 UDB Option for CICS, provide the following information for configuring a TCP/IP listener for these options. Otherwise, skip to the next step.

- *TCP Address Space Name* – the name of your TCP/IP region.
- *Listener Port* – the port on which the option listens.

Note The Server Option for CICS and the DB2 UDB Option for CICS share the same TCP/IP listener.

- 9 Click Next until the wizard displays the information you entered in steps 5 through 8. Review this information and, if necessary, click Back to return to previous screens and make corrections.
- 10 Click Next until the wizard displays a dialog box for FTP information. Provide the following data for establishing an FTP session to your mainframe:
 - *Userid* – the mainframe user ID for the FTP SESSION.
 - *Password* – the password for the FTP SESSION.

- *Mainframe Host Name* – the mainframe DNS name.
- *FTP Port* – the control port used by your mainframe FTP SERVER, usually 21.
- *VOL/UNIT Assignment* – specify either a volume serial number and unit assignment for FTP, or allow FTP to use default values.

Note If you specify a volume serial number that does not exist, FTP suspends operation until the mainframe responds to a message requesting that the volume be mounted.

- *Log FTP Commands* – indicates where FTP log information will be written. This log information may be useful in troubleshooting FTP problems.

When you click Next, the InstallShield wizard will create JCL and upload the selected components to your mainframe.

11 Close the InstallShield wizard.

To complete the installation of your Mainframe Connect components, review and submit JCL from TSO. If you are installing multiple components, Sybase strongly suggests you install in the following sequence:

- 1 Client Option for CICS
- 2 Server Option for CICS
- 3 DB2 UDB Option for CICS
- 4 Any other options

Use the following procedure to complete your installation for the Server Option for IMS. If you are installing the Server Option for MVFS, skip to the next procedure.

❖ **To complete the installation for IMS**

- 1 Locate the installation JCL for the Server Option for IMS in *hlq.OSC126.IMS.JCL*, where *hlq* is the high-level qualifier you specified in step 5.
- 2 Run the following jobs in the order they are described here, where *x* is an integer that denotes the order in which the job is to be run in the overall sequence of jobs. Ignore jobs that are not present or relevant to the option you are installing.

- *IxRECV* – runs IKJEFT01 to use the TSO RECEIVE command to build and populate the product libraries.
 - *IxHOST* – assembles and links the Server Option customization module, character sets, licensing, and remote host definitions. You may rerun this job at any time to change configuration and character sets or to add, remove, or modify remote host definitions.
 - *IxPSB* – generates PSBs.
 - *IxCAPPC* – defines APPC VSAM datasets.
 - *IxUAPPC* – inserts entries into the APPC VSAM datasets.
 - *IxDELETE* – (optional) deletes the data sets in the TSO XMIT format used for the installation.
- 3 Run the following jobs if you want to compile and link-edit the sample applications provided with the Server Option for IMS:
- *PIRDO* – creates sample CICS LU 6.2 RDO entries to be used with IMS.
 - *SAPBIND* – contains the DB2 binds for sample programs that use DB2.
 - *SAPBMPC* – contains the BMP region JCL for the SYICSAV2 sample application.
 - *SAPBMPP* – contains the BMP region JCL for the SYIPSAV1 sample application.
 - *SAPCOB1* – compiles and links COBOL samples that do not use DB2.
 - *SAPCOB2* – compiles and links COBOL samples that use DB2.
 - *SAPDB* – generates IMS data for sample applications.
 - *SAPIMSA* – compiles and links the IMSASYCH sample application.
 - *SAPPLI1* – compiles and links PL/1 samples that do not use DB2.
 - *SAPPLI2* – compiles and links PL/1 samples that use DB2.
 - *SAPTPPRF* – contains TP profiles for implicit and explicit sample applications.

❖ To complete the installation for MVS

- 1 Locate the installation JCL for the Server Option for MVS in *hlq.OSC126.MVS.JCL*, where *hlq* is the high-level qualifier you specified in step 5.
- 2 Run the following jobs in the order they are described here, where *x* is an integer that denotes the order in which the job is to be run in the overall sequence of jobs. Ignore jobs that are not present or relevant to the option you are installing.
 - *IxRECV* – runs IKJEFT01 to use the TSO RECEIVE command to build and populate the product libraries.
 - *IxHOST* – assembles and links the Server Option customization module, character sets, licensing, and remote host definitions. You may rerun this job at any time to change configuration and character sets or to add, remove, or modify remote host definitions.
 - *IxCAPPC* – defines APPC VSAM datasets.
 - *IxDELETE* – (optional) deletes the data sets in the TSO XMIT format used for the installation.
- 3 Run the following jobs if you want to compile and link-edit the sample applications provided with the Server Option for MVS:
 - *PIRDO* – creates sample CICS LU 6.2 RDO entries to be used with MVS.
 - *SAPAPPC*: contains APPC TP profiles for the sample applications.
 - *SAPBIND* – contains the DB2 binds for the sample applications that use DB2.
 - *SAPCOB1* – compiles and links non-VSAM COBOL sample applications.
 - *SAPCOB2*: – compiles and links COBOL sample applications that use DB2.
 - *SAPPLI1* – compiles and links non-VSAM PL/1 sample applications.
 - *SAPPLI2* – compiles and links PL/1 sample applications that use DB2.

Post-installation steps for IMS

The Server Option supports TCP/IP access to IMS using the IBM IMS Connect TCP/IP listener and the Mainframe Connect DirectConnect for z/OS Option. To enable TCP/IP access to IMS, use the following procedure.

❖ To set up TCP access to IMS

- 1 Copy the supplied IMS Connect exit, named SYBTRSEX, from *hlq.OSC126.IMS.LOADLIB* to an APF-authorized library.
- 2 Add the name of the APF-authorized library from step 1 to the STEPLIB of the IMS Connect start-up procedure, for example:

```
//IMS7HW01 PROC
//*
//STEP1 EXEC PGM=HWSHWS00,REGION=7M,TIME=1440,
// PARM='BPECFG=BPECFG01,HWSCFG=HWSCFG01'
//STEPLIB DD DSN=IMS710.HWS110.SHWSRESL,DISP=SHR
// DD DSN=IMS710B.SDFSRESL,DISP=SHR
//PROCLIB DD DSN=IMS710B.PROCLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//HWSRCORD DD DSN=IMS710B.HWSRCDR,DISP=SHR
```

Here, the APF-authorized library is named “IMS710B.SDFSRESL.”

- 3 Add “SYBTRSEX” to the IMS configuration member specified in the start-up procedure, for example:

```
HWSCFG01
```

```
HWS (ID=IMS71B01,RACF=Y,XIBAREA=20)
TCPIP (HOSTNAME=TCPIP,ECB=Y,RACFID=DEFAULT,PORTID=(8880),MAXSOC=500,
TIMEOUT=30000,EXIT=(SYBTRSEX))
DATASTORE (ID=IMSB,GROUP=IMXCF,MEMBER=IMSHWS1,TMEMBER=IMS7B)
```

Here, “SYBTRSEX” is added to the HWSCFG01 member of IMS710B.PROCLIB.

- 4 Have your DirectConnect for z/OS TRS administrator add an IMS region with the same listener port as the PORTID specified in the IMS Connect configuration member, for example:

```
1> exec sgw_addregion IMSB,ophelia,'8880',IMS
```

2> go

Note The DirectConnect for z/OS Option and IBM IMS Connect support only *implicit* IMS transactions for the Server Option. Explicit IMS transactions are not supported.

Libraries and samples

For a list and description of the libraries, sample programs, JCL, and transactions for your product, see the CONTENTS member of the JCL data set.

Topic	Page
Sybase components and security	21
Security for APPC/IMS	24
Implementing conversational security	27
External security systems	27

The information in this chapter applies to the set of DB2 UDB Option for CICS components that make up your LAN-to-mainframe configuration.

Sybase components and security

Sybase components can provide their own security at the following levels:

- Client workstation
- ASE
- TRS
- IMS TM
- Vendor SNA support software

Note Coordinate efforts to set up and maintain security between these components and communicate changes when they occur.

Client workstation

Most workstations have a secure login that verifies the identity and authorization of the user by requiring a unique user ID and password. Client user ID, password, and profile information can be passed to ASE and to the DirectConnect for z/OS Option.

ASE

Adaptive Server® Enterprise (ASE) can grant or deny a user the permission to call a particular remote procedure. Requests routed through ASE undergo security checks. The TRS administrator can apply this security to all requests by specifying the -D (indirect access) parameter when starting TRS. This parameter requires all client requests to go through the ASE. For details, refer to the Mainframe Connect DirectConnect for z/OS Option *Users Guide for Transaction Router Services*.

Note Routing transactions in a server-to-server mode (for example, through an ASE to the mainframe) *requires* a three-tier processing environment.

TRS

This section addresses:

- Defining security
- Overriding security
- Conversation-level security

Defining security

Under TRS security, every client login must be defined to TRS. For each client, this login definition:

- Specifies the client login ID and password
- Optionally specifies a host login ID and password
- Includes lists of the connections and host transactions available to clients using the login

By working with mainframe systems programmers and security administrators, and by carefully defining user IDs, host IDs, transactions, and connections, the TRS administrator can restrict client access to particular host resources.

When the TRS administrator defines a remote procedure call (RPC) to TRS, these security options are available:

- None

- User ID
- Both (user ID and password)

Each option represents the type of login information that is passed to the mainframe when a client calls the RPC. The TRS administrator sets security parameters for a transaction when defining the associated RPC to TRS.

When security is enforced at TRS, the mainframe verifies that the caller is authorized to access the requested transaction. If proper authorization exists, the transaction executes; otherwise, the mainframe returns an error message to the user.

Overriding security

The TRS administrator can override TRS security by setting `Security=no` in the TRS configuration file so you can map users to transaction groups that allow specific RPCs. For more information about the security parameter, see the Mainframe Connect DirectConnect for z/OS Option *Users Guide for Transaction Router Services*.

Conversation-level security

You can set up conversation-level security, a process by which TRS passes client login information to the mainframe when it allocates a conversation. Under conversation-level security, the following can be passed to the host:

- A pre-defined host ID and password, which can be set up in the login definition
- A separate ID and password attached to the transaction group of the client

IMS™

You can use RACF to check whether the inbound request is authorized to access the local LU. However, if you are using the Adapter, it sends only the password, not the user ID. You can use the Gateway-Library function TDGETUSR to retrieve the user ID from the client login packet. For details on TDGETUSR, see the appropriate Mainframe Connect Server Option *Programmers Reference*. PL/1 and COBOL versions of this guide are available.

For further details, refer to the appropriate IBM documentation.

Vendor SNA support software

The SNA support software of the vendor may send login information to the host in FMH-5 fields with client requests. This allows you to use external security products that require client login information.

Security for APPC/IMS

APPC/MVS provides two categories of APPC/IMS TM security:

- LU security – to protect logical units
- Conversational security – to protect transaction programs

LU security

LU security includes:

- Session level security, which uses the RACF resource class APPCLU to set up session keys for LUs. If two LUs try to initiate a session and the session keys do not match, SNA and RACF does not allow the session to occur.
- Control of user access to LUs through the RACF class APPL.
- Control of user access from an LU through the RACF class APPCPORT.

For further details, refer to the appropriate IBM documentation.

Conversational security

Conversational security uses the RACF resource class APPCTP to determine the user IDs that can access APPC/MVS TP profiles and their associated transaction program names (TPNs).

Conversational security depends on the level of security defined in the SNA APPL statement for the APPC LU. Based on the level of security you specify, MVS checks every transaction initiated from an LU 6.2 workstation.

IMS TM adapter The Adapter sends only the password, not the user ID. To do a security check, you can use the Gateway-Library function TDGETUSR to retrieve the user ID from the client login packet. For details on TDGETUSR, see the appropriate Mainframe Connect Server Option *Programmers Reference*. PL/1 and COBOL versions of this guide are available.

Implementing conversational security

Conversational security (CONV) requires changes to APPC/MVS, RACF, and SNA. This section addresses Server Option-specific requirements. For details on IMS TM and APPC/MVS security, see the appropriate IBM documentation.

APPC/MVS definitions

❖ To set up APPC/MVS for the Server Option

- Define a token for the TP profile data set that corresponds to the APPC/IMS TM LU, using the DBMODIFY command of the ATBSDFMU utility:

```
//STEP1EXEC PGM=ATBSDFMU
//STEPLIBDD DSN=IMS41.RESLIB,DISP=SHR
//SYSPRINTDDSYSOUT=A
//SYSSDLIBDDDSN=SYS1.APPCTP,DISP=SHR
//SYSSDOUTDDSYSOUT=*
//SYSINDD*
DBMODIFY
                                DBTOKEN(SYBAPPC)

/*
//
```

RACF definitions

❖ To set up RACF for the Server Option

- 1 Define a resource profile DBTOKEN.X.TPNAME in the APPCTP class for each TPNAME defined in the TP profile data set.
- 2 Add the user IDs or groups to the access list.

- 3 Activate and refresh the APPCTP class.

SNA definitions

❖ To set up SNA for the Server Option

- 1 Define the SECACPT=CONV parameter on the APPL definition for the APPC/IMS LU.
- 2 To allow an LU to support conversation level security, you must set the PSERVIC parameter of a Logmode entry. Assign each LU a Logmode corresponding to the desired level of security.

The 10th byte of PSERVIC determines security as follows:

- x'00' – LOCAL
- x'12' – IDENTIFY
- x'10' – VERIFY

Example of conversation level security

As shown in this example, to set up a successful security system for use with the Server Option in an IMS TM LU 6.2 environment, you need to synchronize SNA and IMS TM very carefully.

❖ To set up conversational level security

- 1 At the mainframe, go to the SNA LogMode entry.
- 2 Define security as “VERIFY,” which requires a user ID and password. This example highlights security defined as VERIFY:

```
SYBTABVMODETAB
  EJECT
  TITLE 'M6P1024V'
*-----*
* LU6.2,SINGLE SESSIONS, RU_SIZE(1024),
* SYNCH_LEVEL(CONFIRM), SECURITY(VERIFY)
*-----*
M6P1024VMODEENT LOGMODE=M6S1024V,FMPROF=X'13',TSPROF=X'07',
  PRIPROT=X'B0',SECPROT=X'B0',COMPROT=X'78A5',
  RUSIZES=X'8787',TYPE=X'00',
  PSNDPAC=X'05',SRCVPAC=X'05',SSNDPAC=X'05',
  PSERVIC=X'0602000000000000102F00'
```

*MODEEND

Note The 10th byte of the PSERVIC parameter must be set to x'10' for "VERIFY" security.

- 3 In your network definition to SNA, specify the mode you defined in the LogMode entry. You can apply the LogMode entry to a specific LU statement, or apply it globally through the PU statement:

```
SYBPU1PUCUADDR=041 , DLOGMOD=M6P1024V , MAXBFRU=11 , SSCPFM=FSS , +  
    USSTAB=ISTINCDT , DELAY=0 , SECNET=YES , ISTATUS=ACTIVE
```

*

```
SYBLU02LU LOCADDR=2
```

External security systems

Most mainframe-based external security systems, such as RACF, are based on user login information. The system employs user ID and password information, restricting transaction access to authorized users.

Topic	Page
Tracing	29
Accounting	37

Tracing

The Server Option provides tracing functions for tracing program activity, either for all transactions (global tracing) or for individual transactions (specific tracing). The Server Option writes header and data information to the error log under IMS TM in the IMS TM log API traces, which trace calls from the client application to the Server Option.

This section contains the following subsections:

- Server Option trace functions
- Trace log
- Using the tracing facility

Server Option trace functions

The Server Option trace functions provides three types of tracing:

- API tracing, which traces Server Option calls
- TDS header tracing

- Tabular Data Stream™ (TDS) data tracing

Note TDS is the proprietary Sybase protocol that defines the format of data transmitted between client and server programs in an efficient, self-describing manner.

You can enable and disable any kind of transaction tracing globally or specifically, using these Server Option functions:

- TDSETLOG – turns tracing on or off and changes the name of the trace log.
- TDINFLOG – determines whether tracing is enabled and names the trace log.
- TDSETSPT, TDLSTSPT, and TDINFSPT – enable, disable, and retrieve information about specific tracing.
- TDWRTLOG – writes your own record or adds a system entry to the trace log file.

For complete descriptions and examples of these functions, see the appropriate Mainframe Connect Server Option *Programmers Reference*. PL/1 and COBOL versions of this guide are available.

Trace log

Under IMS TM, the Gateway-Library trace facility writes header, data, and API tracing information to the IMS TM log. The same IMS TM log contains errors, tracing, and accounting, so each record needs to indicate the type of record it is. This information comes from IxHost, a JCL member with configuration options for Server Options.

Trace log layout

Note The structures of the trace log have the same layout in CICS as in IMS TM, although the log headers differ slightly.

Table 5-1: Trace log layout

Position	Field name	Field type	Field description
1-2	log-length	Unsigned 2-byte integer	Record length. The total length of this record. (Maximum size is 640.)

Position	Field name	Field type	Field description
3	log-type	Unsigned byte	Type of record, either trace or accounting: <ul style="list-style-type: none"> The trace type can be an error record (TDS-ERR-REC, with a value of 1) or a trace record (TDS-TRACE, with a value of 2). The accounting log type value is 0.
4	log-direction	Unsigned byte	Communication state that shows whether the mainframe is in send or receive mode.
5-68	log-key	Unsigned byte (max. length = 30) Unsigned byte (length=8) Unsigned byte (length = 24) Unsigned 2-byte integer	Structure containing: user_id – server login ID of the client, from the login packet. trace_resid – trace resource I, which, in CICS, keeps track of who is doing the tracing. datetime – date and time the SNA conversation or TCP/IP session started (TDACCEPT was issued). uniquekey – reserved for future use to ensure record has unique key.
69-82	log-txp-name	Unsigned byte	Transaction name.
82-112	log-conn-id	Unsigned byte	Connection ID. Name by which the connection is known to TRS.
113	log-connp	Unsigned byte	Name of associated TDPROC structure.
114-117	log-error-rc	4-byte integer	Value returned to the RETURN-CODE parameter of a Server Option function (TDS-xxxx). See the Mainframe Connect Client Option and Server Option <i>Messages and Codes</i> for more information about return codes.
118-121	log-err-type	4-byte integer	Type of error detected.
122-123	log-err-reserved	2-byte integer	Reserved for future use.
124-125	log-data-length	2-byte integer	Length of the data to be logged.
126-637	log-data	Unsigned byte	Data, including the packet header and the data. For TDWRTLOG, this is the message being written to the log.
637-640	log_filler	Unsigned byte (length=36)	Filler to fill out record to 640 bytes.

Using the tracing facility

This section describes how to use the Server Option tracing facility.

❖ To use the trace facility

- 1 Call TDSETLOG and perform these steps for global or specific tracing:

For this type of tracing	Do this
Global	<ol style="list-style-type: none"> 1 Set the trace flag to the TRACE ALL RPCS option. 2 Set the flag for each kind of tracing you want to TRUE.
<p>Note To enable tracing for the entire program, TDSETLOG must precede TDACCEPT.</p>	
Specific	<ol style="list-style-type: none"> 1 Set the trace flag to the TRACE SPECIFIC RPCS option. 2 Set the flag for each desired type of tracing to TRUE.
Both global and specific	<ol style="list-style-type: none"> 1 Perform the previous steps for global and specific tracing. 2 If you are developing under CICS and want to enable API tracing, give the auxiliary trace log a CICS ID.

Note You can use TDINFLOG at any time to check the value of the settings.

- 2 For each transaction for which you want trace activity, call TDSETSPT and perform these steps:
 - 1 Identify the transaction.
 - 2 Set the transaction trace flag to TRUE.
 - 3 Set the trace options flags for the types of tracing desired.

Note You can enable tracing for up to eight transactions at a time.

Using the trace table for individual transactions

When you enable tracing for an individual transaction, TDSETSPT adds the transaction to a trace table, which can contain up to eight entries. For examples of trace tables, see “Example of specific tracing” on page 33.

When you disable tracing for a transaction, its position in the trace table becomes available for another transaction. If all eight positions are in use, you can trace more transactions *only* if you turn tracing off for one of the transactions in the list or if you set global tracing on.

❖ **To query the trace table**

- Use one of the following methods:
 - Call TDINFSPT to determine if tracing is enabled for a specific transaction. You specify the transaction ID, and TDINFSPT returns the trace flag setting, or
 - Call TDLSTSPT to get a list of all transactions for which tracing is currently enabled. TDLSTSPT returns this list as an array.

Example of specific tracing

This example shows how to enable or disable tracing for specific transactions. It also shows how TDSETSPT calls affect the contents of the trace table. TDS packet tracing is initially turned on for eight specific transactions. Tracing continues for the specified functions until a TDSETSPT call turns tracing off for those functions or until TDSETLOG disables tracing entirely.

This example does not show exact syntax or arguments; it merely indicates which flags and transactions are set. For an example of exact coding, see the sample program in the appropriate *Programmers Reference* for the Server Option. PL/1 and COBOL versions of this guide are available.

```
*-----*
* First, initialize your environment and set on specific tracing. *
*-----*
CALL 'TDINIT' ...
CALL 'TDSETLOG' ... (global flag: OFF,
                    API flag: ON,
                    header flag: OFF,
                    data flag: OFF)...

*-----*
* Enable packet tracing (option 01) for a specific transaction.*
*-----*
CALL 'TDSETSPT' ... (trace flag: ON,
                   trace option: 01,
                   tran ID: MYT1)...

*-----*
* Use the same parameter values except the transaction ID*
```

```

* in the next seven TDSETSPT calls.*
*-----*
CALL 'TDSETSPT' ... (tran ID: MYT2)...
CALL 'TDSETSPT' ... (tran ID: MYT3)...
CALL 'TDSETSPT' ... (tran ID: MYT4)...
CALL 'TDSETSPT' ... (tran ID: MYT5)...
CALL 'TDSETSPT' ... (tran ID: MYT6)...
CALL 'TDSETSPT' ... (tran ID: MYT7)...
CALL 'TDSETSPT' ... (tran ID: MYT8)...

*-----*
* With tracing on, begin to accept and process client requests.*
*-----*
CALL 'TDACCEPT'
.
.
.

```

At this point, the trace table looks like this:

Table 5-2: Sample trace table (1)

Transaction ID	Tracing flag
MYT1	TRUE
MYT2	TRUE
MYT3	TRUE
MYT4	TRUE
MYT5	TRUE
MYT6	TRUE
MYT7	TRUE
MYT8	TRUE

Later, you decide to turn on tracing for one more transaction called “MYT9”:

```

*-----*
* Try to turn on packet tracing for MYT9. *
*-----*
CALL 'TDSETSPT' ... (trace flag: ON,
                    trace option: 01,
                    tran ID: MYT9)...

*-----*
* The operation fails, and you get a return code of SOS, *
* indicating that the trace table is full. *
* The contents of the trace table do not change. *
* To make room in the table for MYT9, you decide to *
* turn tracing off for MYT0. *

```

```

*-----*
CALL 'TDSETSPT'...(trace flag: OFF,
                    trace option: 01,
                    tran ID: MYT0)...
*-----*
* The operation fails, and you get a return code
* of ENTRY NOT FOUND, indicating that there is no such
* transaction listed in the trace table.
* The contents of the trace table do not change.
*-----*
* Since you apparently do not have an up-to-date list of the
* contents of the trace table, you use TDLSTSPT to survey
* all entries.
*
* TDLSTSPT returns an array containing eight elements, each
* containing the transaction ID of an entry in the trace table
* for which tracing is TRUE.
*-----*
CALL 'TDLSTSPT' ...
*-----*
* You decide to turn tracing off for MYT3.
*-----*
CALL 'TDSETSPT' ...(trace flag: OFF,
                    trace option: 08,
                    tran ID: MYT3)...
*-----*
* The operation succeeds; the return code is OK.
*-----*

```

The trace table now looks like this:

Table 5-3: Sample trace table (2)

Transaction ID	Tracing flag
MYT1	TRUE
MYT2	TRUE
MYT3	FALSE
MYT4	TRUE
MYT5	TRUE
MYT6	TRUE
MYT7	TRUE
MYT8	TRUE

Note The third position in the trace table is now considered empty.

When you try again to turn tracing on for MYT9, TDSETSPT moves it into the open position in the trace table:

```
*-----*
* Try to enable tracing for MYT9.                *
*-----*
CALL 'TDSETSPT' ... (trace flag: ON,
                    trace option: 01,
                    tran ID: MYT9)
.
.
```

The trace table now looks like this:

Table 5-4: Sample trace table (3)

Transaction ID	Tracing flag
MYT1	TRUE
MYT2	TRUE
MYT9	TRUE
MYT4	TRUE
MYT5	TRUE
MYT6	TRUE
MYT7	TRUE
MYT8	TRUE

Still later, you decide to turn on tracing for MYT2:

```
*-----*
* Try to enable tracing for MYT2.                *
*-----*
CALL 'TDSETSPT' ... (trace flag: OFF,
                    trace option: 01,
                    tran ID: MYT2)...
*-----*
* The operation fails. You get a TDS DUPLICATE ENTRY return code, as *
* tracing is already enabled for the transaction-no action needed.  *
*-----*
```


Accounting

The Server Option allows you to record accounting information at the mainframe and at TRS. Mainframe-based accounting is independent of TRS-based accounting. For example, when the TRS accounting facility records a packet is received, it is recording the number of packets sent from the mainframe to TRS; however, when the mainframe accounting facility records a packet is received, it is recording the number of packets sent from TRS to the mainframe.

Accounting can be enabled at TRS, at the mainframe, or both. For information on TRS accounting, see the *Mainframe Connect DirectConnect for z/OS Option Users Guide for Transaction Router Services*. This section describes accounting at the mainframe.

Note The mainframe accounting facility uses elapsed time.

Under MVS, a point in time is the number of seconds and milliseconds since 00:00:00 on 1/1/70. Elapsed time is the difference between the ending time and the starting time, that is, the number of seconds and milliseconds that elapse between the start and end points.

The following subsections explain the Server Option functions and the accounting log:

- Server Option accounting functions
- Accounting log

Server Option accounting functions

To enable mainframe server accounting information, call TDSETACT in your Server Option program. TDSETACT begins recording when your program issues a TDACCEPT and continues until the program issues TDFREE. Use TDINFACT to learn whether accounting recording is enabled and the name of the accounting log file.

See the *Programmers Reference* for the Server Option for complete descriptions and examples of these functions. PL/1 and COBOL versions of this guide are available.

Accounting log

Warning! The IMS TM log must be set up during customization using IMSLOGTYPE.

Under IMS TM, mainframe-based accounting stores accounting information in the IMS TM log.

Note The structures of the accounting log have the same layout in CICS as in IMS TM, although the log headers differ slightly.

Table 5-5: Accounting log layout

Position	Field name	Field type	Field description
1-2	acct-length	Unsigned 2-byte integer	The total length of this accounting record. (Maximum size of a CICS record is 256.)
3	acct-type	Unsigned byte	Type of record. For the accounting log, this type is always TDS-ACCT-REC.
4	acct-direction	Unsigned byte	Reserved for future use.
5-68			Structure containing the following:
	acct-key	Unsigned byte (max. length = 30)	user_id: Client's server login ID, from the login packet.
		Unsigned byte (length = 24)	trace_resid: Trace resource ID. In CICS, this keeps track of who is doing the tracing.
		Unsigned byte (length=8)	datetime: Date and time the SNA conversation or TCP/IP session started. (TDACCEPT was issued.)
		Unsigned 2-byte integer	uniquekey: Reserved for future use to ensure record has unique key.
69-82	acct-tpx-name	Unsigned byte	Transaction name.
83-112	acct-server-id	Unsigned byte	TRS name. Name of the TRS sending the current request.
113-142	acct-conn-id	Unsigned byte	Connection ID. Name by which the connection is known to TRS.
143-144	Filler	Unsigned byte	Filler to allow next entries to be fullwords.
145-148	acct-tot-secs-wall	4-byte integer	Elapsed wall clock time, in seconds, during the SNA conversation or TCP/IP session.
149-152	acct-tot-fracsecs-wall	Unsigned 4-byte integer	Elapsed wall clock time, in milliseconds during the SNA conversation or TCP/IP session.
153-156	acct-tot-secs-cpu	4-byte integer	CPU time used, in seconds during the SNA conversation or TCP/IP session.

Position	Field name	Field type	Field description
157-160	acct-tot-fracsecs-cpu	Unsigned 4-byte integer	CPU time used, in milliseconds, during the SNA conversation or TCP/IP session.
161-164	acct-tot-sent-bytes	4-byte integer	Total number of TDS bytes sent during an SNA conversation or TCP/IP session.
165-168	acct-tot-sent-packets	4-byte integer	Total number of TDS packets sent during an SNA conversation or TCP/IP session.
169-172	acct-tot-sent-msgs	4-byte integer	Total number of TDS messages sent during an SNA conversation or TCP/IP session.
173-176	acct-tot-sent-rows	4-byte integer	Total number of TDS rows sent during an SNA conversation or TCP/IP session.
177-180	acct-tot-sent-requests	4-byte integer	Total number of RPCs or SQL requests sent during an SNA conversation or TCP/IP session. For the Server Option, this is always 0.
181-184	acct-tot-rcvd-bytes	4-byte integer	Total number of TDS bytes received during an SNA conversation or TCP/IP session.
185-188	acct-tot-rcvd-packets	4-byte integer	Total number of TDS packets received during an SNA conversation or TCP/IP session.
189-192	acct-tot-rcvd-msgs	4-byte integer	Total number of TDS messages received during an SNA conversation or TCP/IP session.
193-196	acct-tot-rcvd-rows	4-byte integer	Total number of TDS rows received during an SNA conversation or TCP/IP session. For the Server Option, this is always 0.
197-200	acct-tot-rcvd-requests	4-byte integer	Total number of RPCs or SQL requests received during an SNA conversation or TCP/IP session.
201-204	acct-tot-rcvd-cancels	4-byte integer	Total number of Cancels or Attentions received during an SNA conversation or TCP/IP session.
205-208	acct-reserved1	4-byte integer	Reserved for future use.
209-212	acct-reserved2	4-byte integer	Reserved for future use.
213-216	acct-reserved3	4-byte integer	Reserved for future use.
217-220	acct-reserved4	4-byte integer	Reserved for future use.
221-236	acct_fill	Unsigned byte (length=36)	Filler to fill out record to 256 bytes.

Customization Options

Topic	Page
Overview	41
Customizing global options (SYGWCST)	42
Using the IBM z/OS conversion environment and services	45
Customizing mainframe character set conversion options (SYGWCXL)	45
Customizing dynamic network drivers (SYGWDRIV)	51
Customizing the TCP/IP driver (SYGWHOST)	53
Defining license keys (SYGWLKEY)	54
Building a global customization module (SYGWXCPH)	55

Overview

You can customize Sybase mainframe access components to meet the requirements at your site. The customization load module SYGWXCPH is a table created by assembling and linking five macros:

- SYGWCST – a global customization macro.
- SYGWCXL – a character set conversion macro.
- SYGWDRIV – specifies which dynamic network drivers are used at the site.
- SYGWHOST – provides mapping between Sybase Server names and TCP/IP addresses or host names.
- SYGWLKEY – a license key macro.

The SYGWXCPH table is shared by the Client Option and the Server Option.

Customizing global options (SYGWMCSST)

SYGWMCSST, one of the macros in table SYGWXCPH, provides options for customizing the Client Option and the Server Option. Some Server Option parameters are used only for customizing the DB2 UDB Option for CICS. You can customize SYGWMCSST using the provided JCL member.

Table A-1 describes SYGWMCSST parameters. Except where noted, these apply to both the Client Option for IMS and Server Option for IMS.

Table A-1: Complete list of SYGWMCSST parameters

Parameter	Default	Format	Purpose
ACCESSCODE (Server Option <i>only</i>)	blank	Up to 32 characters	<p>Defines an access code, which is then compared to the access code supplied by Server Option programs using TDGETUSR.</p> <p>If the access codes do not match, the client password is not returned to the caller of Server Option programs using TDGETUSR.</p> <p>See the appropriate Mainframe Connect Server Option <i>Programmers Reference</i> for details on TDGETUSR.</p>
ACCESSCODESW (Server Option <i>only</i>)	N	Y or N	<p>Turns on/off access code comparison (see ACCESSCODE value).</p> <p>When ACCESSCODESW=N (default), the logged-in password is always returned to the caller of Server Option programs using TDGETUSR.</p> <p>When ACCESSCODESW=Y, the logged-in password is returned only if the access code passed to TDGETUSR matches the access code specified in SYGWMCSST ACCESSCODE.</p>
CHARSETSRV	iso_1	Up to 32 characters	<p>Specifies the default character set that the Client Option or Server Option uses internally. The valid values are iso_1 and utf8.</p> <hr/> <p>Note The value utf8 is valid only if USEIBMUNICODE is set to Y.</p>
DEBUGSW	N	Y or N	<p>Specifies whether or not debugging messages, used in troubleshooting, should be displayed in the system log.</p>

Parameter	Default	Format	Purpose
DECPOINT (<i>Server Option only</i>)	'.'	Either a decimal point or comma delimited by single quotation marks	Decimal point indicator, used only with the DB2 UDB Option for CICS.
DEFLTPROTOCOL	TCP	TCP	Specifies the default network driver protocol.
DQUOTETRAN (<i>Server Option only</i>)	Y	Y or N	Used only with the DB2 UDB Option for CICS. Make this setting consistent with your DB2 configuration. When DQUOTETRAN=Y (default), double quotes are translated to single quotes in incoming SQL text. If you are using an ODBC driver, set DQUOTETRAN=N. Note If you are using double-byte or multi-byte characters for DB2 metadata, set DQUOTETRAN=N.
IMSLOGTYPE (<i>IMS TM only</i>)	A1	A value greater than or equal to A0	Specifies a log type. IMS TM reserves values less than A0.
LONGVARTRUNC	N	Y or N	Indicates whether to truncate LongVarChar and VarBinary. <i>For CICS only:</i> Coordinate this setting with the DirectConnect for z/OS Option TRS. If either this parameter or the TRS TruncateLV configuration property is set for truncation, truncation occurs. If you do not want truncation, set this parameter to N and make sure the TRS TruncateLV configuration property is set to No. See the Mainframe Connect DirectConnect for z/OS Option <i>Users Guide for Transaction Router Services</i> .

Parameter	Default	Format	Purpose
MVSDDNAME (IMS TM and MVS only)	blank	From 1 to 8 characters	<p>The DD name of the MVS Open Client and Open Server log file. If this parameter is left blank (the default), MVS transactions are not logged. If you enter a DD name of 1-8 characters, MVS transactions are logged. The name specified here must match a DD name specified in each MVS transaction profile job.</p> <p>MVSDDNAME must match a DD name specified in the JCL for one of the following:</p> <ul style="list-style-type: none"> • An MVS job • An MVS started task • The MVS transaction profile (if run in an APPC initiator as a transaction)
NATLANGUAGESRV	us_english	Up to 32 characters	<p>Designates the default national language used by the Client Option or Server Option. Also see the CHARSETSRV property.</p>
ROWLIMIT (Server Option only)	0 (zero)		<p>Used only by the DB2 UDB Option for CICS:</p> <ul style="list-style-type: none"> • ROWLIMIT=0 – there is no limit to the number of rows that can be sent. • ROWLIMIT=<i>n</i> – <i>n</i> indicates the global limit of rows that can be sent.
USEIBMUNICODE	N	Y or N	<p>Specifies whether Unicode support for a particular z/OS installation is enabled through the IBM conversion environment and services.</p> <ul style="list-style-type: none"> • If USEIBMUNICODE=Y, IBM support is used for character set conversions. • If USEIBMUNICODE=N, conversion is accomplished through the product-supplied translation tables. <hr/> <p>Note If USEIBMUNICODE=Y, all character sets that are to be used at a particular site must have entries created with the SYGWMCXL macro.</p>

See “Using the IBM z/OS conversion environment and services” on page 45.

Using the IBM z/OS conversion environment and services

Unicode support in the Client Option and Server Option is based on Unicode support provided by IBM z/OS, including the conversion environment and conversion services. When the conversion environment and services are installed and set up, the Client Option and Server Option can convert character streams from one Coded Character Set Identifier (CCSID) to another. This functionality is provided in addition to the support for language and character sets offered in previous versions.

❖ To install IBM Unicode support

- 1 Create an *IMAGE* member in SYS1.PARMLIB using the CUNMIUTL utility.
- 2 Copy the *CUNIMG01* member from WORK.IMAGE to SYS1.PARMLIB.
- 3 Using this command, load the *CUNIMG01* member into z/OS:

```
SET UNI=01
```

- 4 Use this command to display the current active image and the character set conversions defined for that image:

```
DISPLAY UNI, ALL
```

- 5 To enable Unicode support for the Client Option and Server Option, set the USEIBMUNICODE=Y. The USEIBMUNICODE parameter is specified in the SYGWMCS macro in the SYGWXCPH customization module.

For more information on installing Unicode support for IBM z/OS, see “Support for Unicode Using Conversion Services” (SA22-7649-07).

Customizing mainframe character set conversion options (SYGWMCXL)

SYGWMCXL is the character set conversion macro in the SYGWXCPH table. The following considerations apply in using the SYGWMCXL macro:

- When Unicode support is disabled (USEIBMUNICODE=N) and the original translation method is used, SYGWMCXL can be used to override supplied SBCS translation tables or to define new SBCS translation tables.

- When Unicode support is enabled (USEIBMUNICODE=Y), SYGWMCXL is used to create definition entries for the character sets to be used in the Client Option or Server Option conversions at a particular installation. These entries are created in addition to system-generated entries.

Note All EBCDIC-to-ASCII and ASCII-to-EBCDIC translation for Client Option or the Server Option occurs on the mainframe.

Overriding the supplied SBCS translation tables

For SBCS, shipped character sets are called *predefined*, and the character sets you define are called *user-definable*.

Predefined character sets

Predefined SBCSs shipped with the product include:

SBCS	Definition
ascii_8	Default used for logins and for IBM cp1027 (code page 1027) support
cp437 (code page 437)	Used by IBM PCs
cp850 (code page 850)	IBM/Microsoft Multilingual Character Set, used by IBM PCs
iso_1 (ascii 0819)	International ISO standard, 8-bit character set for many systems, and the default for Adaptive Server Enterprise on several platforms
mac (Macintosh Roman)	Default used by Macintosh systems
roman8	Default Hewlett-Packard proprietary character set

Warning! Unpredictable failures can occur if the character set names are changed from lowercase to uppercase.

User-defined character sets

You can change all attributes for user character sets. Table A-2 shows the SBCS settings of the parameters for SYGWMCXL:

Table A-2: SYGWMCXL parameters for SBCS

Parameter	Value
A2E	Optional ASCII-to-EBCDIC translate overrides
E2A	Optional EBCDIC-to-ASCII translate overrides
CHARSET	Name of the SBCS
CHARSETBYTES	S for SBCS
TYPE	Valid types: <ul style="list-style-type: none"> • INITIAL • ENTRY (default) • FINAL

If there is no override entry for a predefined character set, a default entry is generated with the appropriate translation tables and other attributes for that character set. A total of 99 character sets, including custom-generated character set entries, is supported.

The minimum translate customization entries are:

```
SYGWMCXL TYPE=INITIAL
SYGWMCXL TYPE=FINAL
```

These entries generate all of the predefined SBCSs.

Defining new SBCS translation tables

For SBCSs, you can modify the translation tables shipped with the product and create new translation tables with names you define.

Warning! Do not use the shipped table names for the tables you create.

If you create new tables for the Server Option in a three-tier environment, you must coordinate with the person responsible for the Sybase client. The client uses the names of the tables you create to issue logins to the DirectConnect for z/OS Option TRS.

When you finish customizing the SBCS translation tables, rebuild the SYGWXCPH module, and load the new module for your revisions to take effect. Instructions are provided in “Building a global customization module (SYGWXCPH)” on page 55.

Overriding defaults and creating new tables on the mainframe

The SYGWMCXL macro generates translation tables to convert between ASCII and EBCDIC character sets. Default translation tables are generated for the following ASCII character sets:

- `ascii_8`
- `cp437`
- `cp850`
- `iso_1`
- `mac`
- `roman8`

Warning! Unpredictable failures can occur if the character set names are changed from lowercase to uppercase.

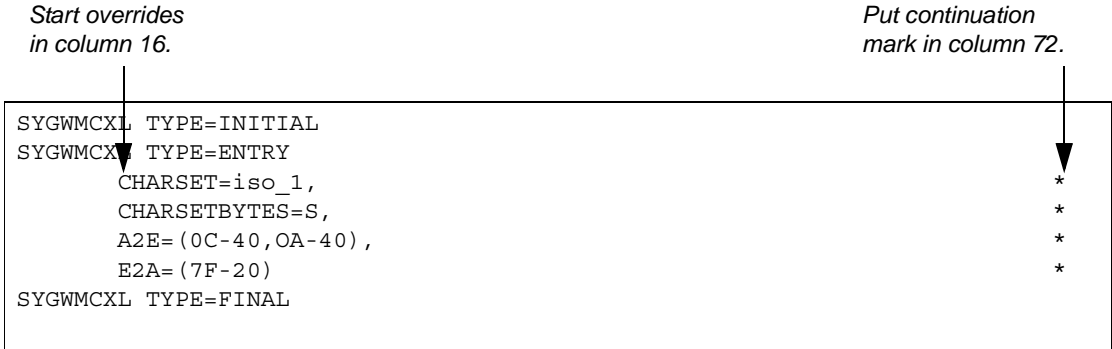
These default tables also provide the “base” for any character set changes or new tables you want to define. For details on the base translate tables, see Appendix B, “Translation Tables.”

You can change all attributes for user character sets. An entry is added to the translate table, specifying the appropriate character set attributes. Two examples follow for overriding defaults.

Overriding ASCII-to-EBCDIC defaults

The first example shows how to use A2E and E2A macro parameters to override the ASCII-to-EBCDIC defaults. You can use uppercase or lowercase to define the parameters.

When you override the ASCII-to-EBCDIC defaults, the appropriate base table is picked up as a template for the character overrides or user-defined character sets, thus generating a default table. In Figure A-1, the client is using `us_english`, which is not predefined.

Figure A-1: Using A2E and E2A example

This example converts both of the following:

- ASCII form feeds (x'0C') and line feeds (x'0A') to EBCDIC spaces (x'40')
- EBCDIC DELs (x'7F') to ASCII space (x'20')

Creating a new table

The next example shows how to modify the default character set, iso_1, for Hebrew, creating a new table:

```

* These SYGWMCXL macro calls modify the iso_1 character set
* to Hebrew.
*
SYGWMCXL          TYPE=INITIAL
SYGWMCXL TYPE=ENTRY,
          CHARSET=(unique_name),
          CHARSETBYTES=S,
          A2E=(E0-41,E1-42,E2-43,E3-44,E4-45,E5-46,E6-47,E7-48,E8-
          49,E9-51,EA-52,EB-53,EC-54,ED-55,EE-56,EF-57,F0-58,F1-59*
          ,F2-62,F3-63,F4-64,F5-65,F6-66,F7-67,F8-68,F9-69,FA-71),*
          E2A=(41-E0,42-E1,43-E2,44-E3,45-E4,46-E5,47-E6,48-E7,49-*
          E8,51-E9,52-EA,53-EB,54-EC,55-ED,56-EE,47-EF,58-F0,59-F1*
          ,62-F2,63-F3,64-F4,65-F5,66-F6,67-F7,68-F8,69-F9,71-FA)
SYGWMCXL TYPE=FINAL
*
* Assembler END is required.
*
END
  
```

For the CHARSET parameter, specify a unique name. This generates a new user-defined table. Provide the name to the appropriate person at the Sybase client site. The client login packet uses this name.

Defining new character set entries

In using the IBM Unicode conversion environment and services, the SYGWMCXL macro is used to create definition entries for all the character sets that will be used at a particular site and that are not already defined as system character sets. Table A-3 describes the parameters used in the SYGWMCXL macro to create a definition entry:

Table A-3: SYGWMCXL macro parameters

Parameter	Value
CHARSET	The name of the SBCS or DBCS character set.
CHARSET BYTES	An S to denote SBCS, or a D to denote DBCS.
CCSID	The CCSID for the character set.
CHARSETTYPE	The type of character set. A denotes ASCII, and E denotes EBCDIC.
CHARSIZE	The maximum length of a character, from 1 to 4 bytes.
PAD	The padding character. The value of this parameter depends on the character set type. For ASCII, the padding character is 20. For EBCDIC, the padding character is 40.

Note If USEIBMUNICODE=Y, all character sets that are to be used at a particular site must have entries created with the SYGWMCXL macro.

The following examples illustrate definitions for Russian and Japanese EBCDIC character sets, which are code pages 1025 and 939, respectively.

Example: code page 1025

```
SYGWMCXL TYPE=ENTRY,
          CHARSET=Russian, CHARSETBYTES=S,
          CCSID=1025, CHARTYPE=E, CHARSIZE=1, PAD=40
```

Example: code page 939

```
SYGWMCXL TYPE=ENTRY,
          CHARSET=cp939, CHARSETBYTES=D,
          CCSID=939, CHARTYPE=E, CHARSIZE=2, PAD=40
```

In addition to the default ASCII SBCS translation tables, these names are used to generate system entries for ASCII DBCS character sets:

- *sjis* – Japanese code page cp943 or cp932

- *eucjis* – Japanese code page cp33722
- *cp950* – traditional Chinese Big5 or cp950
- *cp936* – simplified Chinese GBK or cp936

If you use any of these names, you do not need to create a new definition.

Customizing dynamic network drivers (SYGWDRIV)

SYGWDRIV, a macro in the SYGWXCPH table, defines the dynamic network drivers for the Client Option or the Server Option.

Note If you are using a TCP/IP driver, you must also configure the SYGWHOST macro.

CICS network drivers

Table A-4 shows the default drivers that are shipped with the Client Option or Server Option, depending on the environment:

Table A-4: CICS network drivers

Driver	Load module name	Comments
LU 6.2	LU62CICS	Uses CICS LU 6.2 API
IBM TCP/IP	TCPCICS	Uses IBM EZACICAL API
CPIC	CPICCICS	Uses CICS CPIC Support

The CICS JCL member *IxHOST* contains these macro definitions, which set up support for all three network drivers:

```

SYGWDRIV TYPE=INITIAL
*
SYGWDRIV TYPE=ENTRY, ENV=CICS, NETD=LU62
SYGWDRIV TYPE=ENTRY, ENV=CICS, NETD=CPIC
SYGWDRIV TYPE=ENTRY, ENV=CICS, NETD=TCP
*
SYGWDRIV TYPE=FINAL

```

Using the CPI-C CICS network driver

If you use the CPI-C CICS driver, you must use CEDA to define an entry in the CICS PARTNER Table. Due to an IBM requirement, each Partner entry must be exactly 8 characters in length and use A-Z, 0-9. If your actual server name is not 8 characters, put an alias for it in your *interfaces* file.

Figure A-2: CEDA window

```

OBJECT CHARACTERISTICS                                CICS RELEASE = 0410

CEDA View PARTner( MYSERVER )
  PARTner      : MYSERVER
  Group        : GROUP42
  Description   : SIDE INFO ENTRY TO GET TO mymcg
REMOTE LU NAME
  NETName      : U6T42P0M
  NETWork      :
SESSION PROPERTIES
  Profile      : SYOCPROF
REMOTE TP NAME
  Tpname       :
  Xtpname      : 94A8948387

                                SYSID=CICS APPLID=CICS41

PF1 HELP 2 COM 3 END          6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
  
```

Enter the PARTner and Remote TP name field values:

- PARTner – This must be *exactly* 8 characters long. An alias for the 8-character name should be added to the *interfaces* file if necessary.
- Remote TP name – If the name of your server is in uppercase, enter it in the Tpname field. If the name of your server is in lowercase, enter the EBCDIC hexadecimal name in the Xtpname field.

Note If you enter a lowercase name in the Tpname field, CEDA changes it to uppercase and an erroneous entry is passed.

Customizing the TCP/IP driver (SYGWHOST)

The SYGWHOST macro is part of the SYGWXCPH global customization module. This macro is used only for the Client Option in connections from the mainframe to other applications. It is required only if you are using a TCP/IP driver, in which case you must configure SYGWHOST to define the mapping between Sybase server names and TCP/IP addresses or host names. Do not depend on the default shipped with the installation to work in your environment.

Macro formats

There are three macro formats: TYPE=INITIAL, TYPE=ENTRY, and TYPE=FINAL.

Note For the Server Option, only the TYPE=INITIAL and TYPE=FINAL macros are required. For the Client Option, only the TYPE=ENTRY macro is required.

TYPE=INITIAL

The format of TYPE=INITIAL is:

```
SYGWHOST TYPE=INITIAL
```

TYPE=ENTRY

The format of TYPE=ENTRY is:

```
SYGWHOST TYPE=ENTRY
      IBMTCPADRSPCNAME=&&TCP,
      LISTENER=(LAN,CICS,IMS)
      LSTNPORT=99999,
      SERVERNAME=sybase10,
      HOSTNAME=myhost
```

TYPE=FINAL

The format of TYPE=FINAL is:

```
SYGWHOST TYPE=FINAL
```

Macro parameters

There are six parameters in the SYGWHOST macro:

Parameter	Definition
HOSTNAME	The name of the host on which the Sybase server resides. The maximum length of the host name is 24 characters. If a value is provided for the IPADDR parameter, the HOSTNAME parameter is ignored, and no DNS search is performed.
IBMTCPADDRSPACE	Designates the name of the IBM TCP/IP address space. This parameter can be specified as either of the following: <ul style="list-style-type: none"> • A hard-coded value of up to 8 characters. • A system symbolic name. System symbolic names are defined in the IEASYMxx PARMLIB member and are limited to seven characters preceded by “&&”. For example, the symbolic name “SYBTCP” would be designated as follows: <pre>IBMTCPADDRSPACE=&&SYBTCP</pre> Symbolic names allow the use of a common SYGWXCPH configuration module across multiple LPARs, even if each LPAR has a different TCP address space name. The default address space name is TCPIP.
IPADDR	The IP address of the host on which the Sybase server resides. If a value is provided for this parameter, the HOSTNAME parameter is ignored.
LISTENER	One of the following: <ul style="list-style-type: none"> • <i>LAN</i> if the listen port is for a LAN-based server (default) • <i>CICS</i> if the listen port is for an CICS Server Option listener • <i>IMS</i> if the listen port is for an IMS TM Server Option listener
LSTNPORT	The listen port of the server specified by SERVERNAME.
SERVERNAME	The 1-30 byte name of a Sybase server.

Defining license keys (SYGWLKEY)

The SYGWLKEY macro is part of the SYGWXCPH global customization module. It is used to define the customer license key that is verified at runtime and has two parameters:

Parameter	Definition
PRODUCT	The product related to the license key, either the Client Option, the Server Option, or the DB2 UDB Option. Valid values are OCC, OSC, or DB2.
KEY	Defines the license key given for a product. The license key is a 23-character numeric value.

This example of SYGWLKEY defines license keys for four Mainframe Connect options in the order they are listed: Client Option for CICS, Server Option for CICS, Server Option for IMS and MVS, and DB2 UDB Option for CICS:

```
SYGWLKEY TYPE=INITIAL
SYGWLKEY TYPE=ENTRY, PRODUCT=OCC, KEY=19320-00000-10$*#-#19$B
SYGWLKEY TYPE=ENTRY, PRODUCT=OSC, KEY=19300-00000-10E2G-4K##6
SYGWLKEY TYPE=ENTRY, PRODUCT=OSC, KEY=19315-00000-2$#0$-4A#49
SYGWLKEY TYPE=ENTRY, PRODUCT=DB2, KEY=26875-00239-2$$$A-#AR#H
SYGWLKEY TYPE=FINAL
```

Building a global customization module (SYGWXCPH)

The installation process in Chapter 3, “Installation and Configuration,” creates the *IxTCP* job (where *x* is an integer that denotes the order in which the job is to be run in the overall sequence of jobs). The *IxTCP* job can be run to create a basic version of the SYGWXCPH global customization module, which contains these macros:

- SYGWMCST
- SYGWMCXL
- SYGWDRIV
- SYGWHOST
- SYGWLKEY
- TDSGLOB, a relocatable object module

Translation Tables

Topic	Page
Understanding the ASCII-EBCDIC and EBCDIC-ASCII translation tables	57
Default ASCII_8 translation tables	59
Default ISO_1 translation tables	62
Default cp437 (code page 437) translation tables	64
Default cp850 (code page 850) translation tables	67

Understanding the ASCII-EBCDIC and EBCDIC-ASCII translation tables

This appendix shows the default settings for the ASCII-EBCDIC and EBCDIC-ASCII translation tables before any user overrides.

Note The translation tables shown here are used in date conversion only if Unicode support is disabled and USEIBMUNICODE=N.

The four pairs of default, or "base," tables are:

- ASCII_8
- ISO_1
- cp437
- cp 850

Each pair includes a table for ASCII-to-EBCDIC translation, and one for EBCDIC-to-ASCII translation.

Note As supplied, all ASCII character sets translate to and from EBCDIC code page 500 (iso_1) on the mainframe by default.

For the ASCII-to-EBCDIC tables, find the leftmost hexadecimal ASCII digit to the left of the table as a digit followed by an underscore. Find the rightmost hexadecimal ASCII digit on top of the table as a digit preceded by an underscore.

Here is an example from the default table in the section called “ASCII_8, ASCII-to-EBCDIC translation table” on page 60.

Figure B-1: Example from the ASCII_8, ASCII-to-EBCDIC translation table

	0	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>E</u>	<u>F</u>
0_	00	01	02	03	37	2D	2E	2F	16	05	25	0B	0C	0C	0E	0F
1_	10	11	12	13	3C	3D	32	26	18	19	3F	27	1C	1D	1E	1F
2_	40	5A	7F	7B	5B	6C	50	7D	4D	5D	5C	4E	6B	60	4B	61

ASCII x'26' is translated to EBCDIC x'50'.

To locate ASCII x'26', find row 2_ to the left of the table, and proceed along that row to the column headed by 6. At the intersection is x'50'. Therefore, ASCII x'26' is translated to EBCDIC x'50'.

For the EBCDIC-to-ASCII tables, find the leftmost hexadecimal EBCDIC digit to the left of the table as a digit followed by an underscore. Find the rightmost hexadecimal EBCDIC digit on top of the table as a digit preceded by an underscore.

Here is an example from the default table in the section called “ASCII_8, ASCII-to-EBCDIC translation table” on page 60.

Figure B-2: Example from the ASCII_8, EBCDIC-to-ASCII translation table

	0	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>E</u>	<u>F</u>
0_	00	01	02	03	20	09	20	7F	20	20	20	0B	0C	0D	0E	0F
1_	10	11	12	13	20	20	08	20	18	19	20	20	1C	1D	1E	1F
2_	20	20	1C	20	20	0A	17	1B	20	20	20	20	20	05	06	07

EBCDIC x'26' is translated to ASCII x'17'.

To locate EBCDIC x'26', find row 2_ on the left side of the table; then, proceed along that row to the column headed by _6. At the intersection is x'17'. Therefore, EBCDIC x'26' is translated to ASCII x'17'.

Warning! If you create a new table from a default table, give the new table a unique name and coordinate with the appropriate person at the Sybase client site. The client can use the name to issue logins to TRS.

Default ASCII_8 translation tables

The ASCII-to-EBCDIC translation tables in this section are the base tables for the following predefined system SBCSs:

- `ascii_8`
- `roman8`
- `mac`
- `ibmascii`

Use these tables as the base ASCII-to-EBCDIC translation table for user-definable character sets.

ASCII_8, ASCII-to-EBCDIC translation table

Figure B-3: ASCII_8, ASCII-to-EBCDIC translation table

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	37	2D	2E	2F	16	05	25	0B	0C	0D	0E	0F
1_	10	11	12	13	3C	3D	32	26	18	19	3F	27	1C	1D	1E	1F
2_	40	5A	7F	7B	5B	6C	50	7D	4D	5D	5C	4E	6B	60	4B	61
3_	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	5E	4C	7E	6E	6F
4_	7C	C1	C2	C3	C4	C5	C6	C7	C8	C9	D1	D2	D3	D4	D5	D6
5_	D7	D8	D9	E2	E3	E4	E5	E6	E7	E8	E9	AD	E0	BD	5F	6D
6_	79	81	82	83	84	85	86	87	88	89	91	92	93	94	95	96
7_	97	98	99	A2	A3	A4	A5	A6	A7	A8	A9	8B	6A	9B	A1	07
8_	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
9_	90	91	92	93	94	95	96	97	98	99	9A	4A	9C	9D	9E	9F
A_	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	5F	AB	AC	AD	AE	AF
B_	B0	B1	B2	4F	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BC
C_	AB	C1	C2	C3	BF	8F	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF
D_	D0	D1	D2	D3	D4	D5	D6	D7	D8	BB	AC	DB	DC	DD	DE	DF
E_	E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	EF
F_	F0	9E	AE	8C	F4	F5	F6	F7	A1	AF	FA	FB	FC	FD	9F	FF

ASCII_8, EBCDIC-to-ASCII translation table*Figure B-4: ASCII_8, EBCDIC-to-ASCII translation table*

	0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	20	09	20	7F	20	20	20	0B	0C	0D	0E	0F
1_	10	11	12	13	20	20	08	20	18	19	20	20	1C	1D	1E	1F
2_	20	20	1C	20	20	0A	17	1B	20	20	20	20	20	05	06	07
3_	20	20	16	20	20	20	20	04	20	20	20	20	14	15	20	1A
4_	20	20	20	20	20	20	20	20	20	20	9B	2E	3C	28	2B	B3
5_	26	20	20	20	20	20	20	20	20	20	21	24	2A	29	3B	AA
6_	2D	2F	20	20	20	20	20	20	20	20	7C	2C	25	5F	3E	3F
7_	20	20	20	20	20	20	20	20	20	60	3A	23	40	27	3D	22
8_	20	61	62	63	64	65	66	67	68	69	20	7B	F3	20	20	C5
9_	20	6A	6B	6C	6D	6E	6F	70	71	72	20	7D	20	20	F1	FE
A_	20	7E	73	74	75	76	77	78	79	7A	20	C0	DA	5B	F2	F9
B_	20	20	20	20	20	20	20	20	20	20	20	D9	BF	5D	20	C4
C_	7B	41	42	43	44	45	46	47	48	49	20	20	20	20	20	20
D_	7D	4A	4B	4C	4D	4E	4F	50	51	52	20	20	20	20	20	20
E_	5C	20	53	54	55	56	57	58	59	5A	20	20	20	20	20	20
F_	30	31	32	33	34	35	36	37	38	39	20	20	20	20	20	20

Default ISO_1 translation tables

The ASCII-to-EBCDIC translation tables in this section are the base table for the predefined system iso_1 character set.

ISO_1 ASCII-to-EBCDIC translation table

Figure B-5: ISO_1 ASCII-to-EBCDIC translation table

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	37	2D	2E	2F	16	05	25	0B	0C	0D	0E	0F
1_	10	11	12	13	3C	3D	32	26	18	19	3F	27	1C	1D	1E	1F
2_	40	4F	7F	7B	5B	6C	50	7D	4D	5D	5C	4E	6B	60	4B	61
3_	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	5E	4C	7E	6E	6F
4_	7C	C1	C2	C3	C4	C5	C6	C7	C8	C9	D1	D2	D3	D4	D5	D6
5_	D7	D8	D9	E2	E3	E4	E5	E6	E7	E8	E9	4A	E0	5A	5F	6D
6_	79	81	82	83	84	85	86	87	88	89	91	92	93	94	95	96
7_	97	98	99	A2	A3	A4	A5	A6	A7	A8	A9	C0	BB	D0	A1	07
8_	20	21	22	23	24	15	06	17	28	29	2A	2B	2C	09	0A	1B
9_	30	31	1A	33	34	35	36	08	38	39	3A	3B	04	14	3E	FF
A_	41	AA	B0	B1	9F	B2	6A	B5	BD	B4	9A	6A	BA	CA	AF	BC
B_	90	8F	EA	FA	BE	A0	B6	B3	9A	DA	9B	8B	B7	C7	B9	AB
C_	64	65	62	66	63	67	9E	69	74	71	72	73	78	75	76	77
D_	AC	69	ED	EE	EB	EF	EC	BF	80	FD	FE	FB	FC	AD	AE	59
E_	44	45	42	46	43	47	9C	48	54	51	52	53	58	55	56	57
F_	8C	49	CD	CE	CB	CF	CC	E1	70	DD	DE	DB	DC	8D	8E	DF

ISO_1 EBCDIC-to-ASCII translation table*Figure B-6: ISO_1 EBCDIC-to-ASCII translation table*

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	9C	09	86	7F	97	8D	8E	0B	0C	0D	0E	0F
1_	10	11	12	13	9D	85	08	87	18	19	92	8F	1C	1D	1E	1F
2_	80	81	82	83	84	0A	17	1B	88	89	8A	8B	8C	05	06	07
3_	90	91	16	93	94	95	96	04	98	99	9A	9B	14	15	9E	1A
4_	20	A0	E2	E4	E0	E1	E3	E5	E7	F1	5B	2E	3C	28	2B	21
5_	26	E9	EA	EB	E8	ED	EE	EF	EC	DF	5D	24	2A	29	3B	5E
6_	2D	2F	C2	C4	C0	C1	C3	C5	C7	D1	A6	2C	25	5F	3E	3F
7_	F8	C9	CA	CB	C8	CD	CE	CF	CC	60	3A	23	40	27	3D	22
8_	D8	61	62	63	64	65	66	67	68	69	AB	BB	F0	FD	FE	B1
9_	B0	6A	6B	6C	6D	6E	6F	70	71	72	AA	BA	E6	B8	C6	A4
A_	B5	7E	73	74	75	76	77	78	79	7A	A1	BF	D0	DD	DE	AE
B_	A2	A3	A5	B7	A9	A7	B6	BC	BD	BE	AC	7C	AF	A8	B4	D7
C_	7B	41	42	43	44	45	46	47	48	49	AD	F4	F6	F2	F3	F5
D_	7D	4A	4B	4C	4D	4E	4F	50	51	52	B9	FB	FC	F9	FA	FF
E_	5C	F7	53	54	55	56	57	58	59	5A	B2	D4	D6	D2	D3	D5
F_	30	31	32	33	34	35	36	37	38	39	B3	DB	DC	D9	DA	9F

Default cp437 (code page 437) translation tables

This section contains the following tables:

- cp437 ASCII-to-EBCDIC translation table
- cp437 EBCDIC-to-ASCII translation table

The ASCII-to-EBCDIC translation tables in this section are the base tables for the predefined system cp 437 (code page 437) character set.

cp437 ASCII-to-EBCDIC translation table*Figure B-7: cp437 ASCII-to-EBCDIC translation table*

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	37	2D	2E	2F	16	05	25	0B	0C	0D	0E	0F
1_	10	11	12	13	B6	B5	32	26	18	19	1C	27	07	1D	1E	1F
2_	40	4F	7F	7B	5B	6C	50	7D	4D	5D	5C	4E	6B	60	4B	61
3_	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	5E	4C	7E	6E	6F
4_	7C	C1	C2	C3	C4	C5	C6	C7	C8	C9	D1	D2	D3	D4	D5	D6
5_	D7	D8	D9	E2	E3	E4	E5	E6	E7	E8	E9	4A	E0	5A	5F	6D
6_	79	81	82	83	84	85	86	87	88	89	91	92	93	94	95	96
7_	97	98	99	A2	A3	A4	A5	A6	A7	A8	A9	C0	BB	D0	A1	3F
8_	68	DC	51	42	43	44	47	48	52	53	54	57	56	58	63	67
9_	71	9C	9E	CB	CC	CD	DB	DD	DF	EC	FC	B0	B1	B2	3E	B4
A_	45	55	CE	DE	49	69	9A	9B	AB	9F	BA	B8	B7	AA	8A	8B
B_	3C	3D	62	6A	64	65	66	20	21	22	70	23	72	73	74	BE
C_	76	77	78	80	24	15	8C	8D	8E	FF	06	17	28	29	9D	2A
D_	2B	2C	09	0A	AC	AD	AE	AF	1B	30	31	FA	1A	33	34	35
E_	36	59	08	38	BC	39	A0	BF	CA	3A	FE	3B	04	CF	DA	14
F_	EE	8F	46	75	FD	EB	E1	ED	90	EF	B3	FB	B9	EA	BD	41

cp437 EBCDIC-to-ASCII translation table

Figure B-8: cp437 EBCDIC-to-ASCII translation table

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	EC	09	CA	1C	E2	D2	D3	0B	0C	0D	0E	0F
1_	10	11	12	13	EF	C5	08	CB	18	19	DC	D8	1A	1D	1E	1F
2_	B7	B8	B9	BB	C4	0A	17	1B	CC	CD	CF	D0	D1	05	06	07
3_	D9	DA	16	DD	DE	DF	E0	04	E3	E5	E9	EB	B0	B1	9E	7F
4_	20	FF	83	84	85	A0	F2	86	87	A4	5B	2E	3C	28	2B	21
5_	26	82	88	89	8A	A1	8C	8B	8D	E1	5D	24	2A	29	3B	5E
6_	2D	2F	B2	8E	B4	B5	B6	8F	80	A5	B3	2C	25	5F	3E	3F
7_	BA	90	BC	BD	BE	F3	C0	C1	C2	60	3A	23	40	27	3D	22
8_	C3	61	62	63	64	65	66	67	68	69	AE	AF	C6	C7	C8	F1
9_	F8	6A	6B	6C	6D	6E	6F	70	71	72	A6	A7	91	CE	92	A9
A_	E6	7E	73	74	75	76	77	78	79	7A	AD	A8	D4	D5	D6	D7
B_	9B	9C	9D	FA	9F	15	14	AC	AB	FC	AA	7C	E4	FE	BF	E7
C_	7B	41	42	43	44	45	46	47	48	49	E8	93	94	95	A2	ED
D_	7D	4A	4B	4C	4D	4E	4F	50	51	52	EE	96	81	97	A3	98
E_	5C	F6	53	54	55	56	57	58	59	5A	FD	F5	99	F7	F0	F9
F_	30	31	32	33	34	35	36	37	38	39	DB	FB	9A	F4	EA	C9

Default cp850 (code page 850) translation tables

The EBCDIC-to-ASCII translation tables in this section are the base tables for the predefined system cp 850 (code page 850) character set.

cp850 ASCII-to-EBCDIC translation table

Figure B-9: cp850 ASCII-to-EBCDIC translation table

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	37	2D	2E	2F	16	05	25	0B	0C	0D	0E	0F
1_	10	11	12	13	3C	3D	32	26	18	19	1C	27	07	1D	1E	1F
2_	40	4F	7F	7B	5B	6C	50	7D	4D	5D	5C	4E	6B	60	4B	61
3_	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	5E	4C	7E	6E	6F
4_	7C	C1	C2	C3	C4	C5	C6	C7	C8	C9	D1	D2	D3	D4	D5	D6
5_	D7	D8	D9	E2	E3	E4	E5	E6	E7	E8	E9	4A	E0	5A	5F	6D
6_	79	81	82	83	84	85	86	87	88	89	91	92	93	94	95	96
7_	97	98	99	A2	A3	A4	A5	A6	A7	A8	A9	C0	BB	D0	A1	3F
8_	68	DC	51	42	43	44	47	48	52	53	54	57	56	58	63	67
9_	71	9C	9E	CB	CC	CD	DB	DD	DF	EC	FC	70	B1	80	BF	FF
A_	45	55	CE	DE	49	69	9A	9B	AB	AF	BA	B8	B7	AA	8A	8B
B_	2B	2C	09	21	28	65	62	64	B4	38	31	34	33	B0	B2	24
C_	22	17	29	06	20	2A	46	66	1A	35	08	39	36	30	3A	9F
D_	8C	AC	72	73	74	0A	75	76	77	23	15	14	04	6A	78	3B
E_	EE	59	EB	ED	CF	EF	A0	8E	AE	FE	FB	FD	8D	AD	BC	BE
F_	CA	8F	1B	B9	B6	B5	E1	9D	90	BD	B3	DA	FA	EA	3E	41

cp850 EBCDIC-to-ASCII translation table

Figure B-10: cp850 EBCDIC-to-ASCII translation table

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	00	01	02	03	DC	09	C3	1C	CA	B2	D5	0B	0C	0D	0E	0F
1_	10	11	12	13	DB	DA	08	C1	18	19	C8	F2	1A	1D	1E	1F
2_	C4	B3	C0	D9	BF	0A	17	1B	B4	C2	C5	B0	B1	05	06	07
3_	CD	BA	16	BC	BB	C9	CC	04	B9	CB	CE	DF	14	15	FE	7F
4_	20	FF	83	84	85	A0	C6	86	87	A4	5B	2E	3C	28	2B	21
5_	26	82	88	89	8A	A1	8C	8B	8D	E1	5D	24	2A	29	3B	5E
6_	2D	2F	B6	8E	B7	B5	C7	8F	80	A5	DD	2C	25	5F	3E	3F
7_	9B	90	D2	D3	D4	D6	D7	D8	DE	60	3A	23	40	27	3D	22
8_	9D	61	62	63	64	65	66	67	68	69	AE	AF	D0	EC	E7	F1
9_	F8	6A	6B	6C	6D	6E	6F	70	71	72	A6	A7	91	F7	92	CF
A_	E6	7E	73	74	75	76	77	78	79	7A	AD	A8	D1	ED	E8	A9
B_	BD	9C	BE	FA	B8	F5	F4	AC	AB	F3	AA	7C	EE	F9	EF	9E
C_	7B	41	42	43	44	45	46	47	48	49	F0	93	94	95	A2	E4
D_	7D	4A	4B	4C	4D	4E	4F	50	51	52	FB	96	81	97	A3	98
E_	5C	F6	53	54	55	56	57	58	59	5A	FD	E2	99	E3	E0	E5
F_	30	31	32	33	34	35	36	37	38	39	FC	EA	9A	EB	E9	9F

Troubleshooting

Topic	Page
Coordinating troubleshooting efforts	69
Where to start troubleshooting	74
Common problems and suggested solutions	75
Troubleshooting at each component	78

Coordinating troubleshooting efforts

This section contains the following subsections:

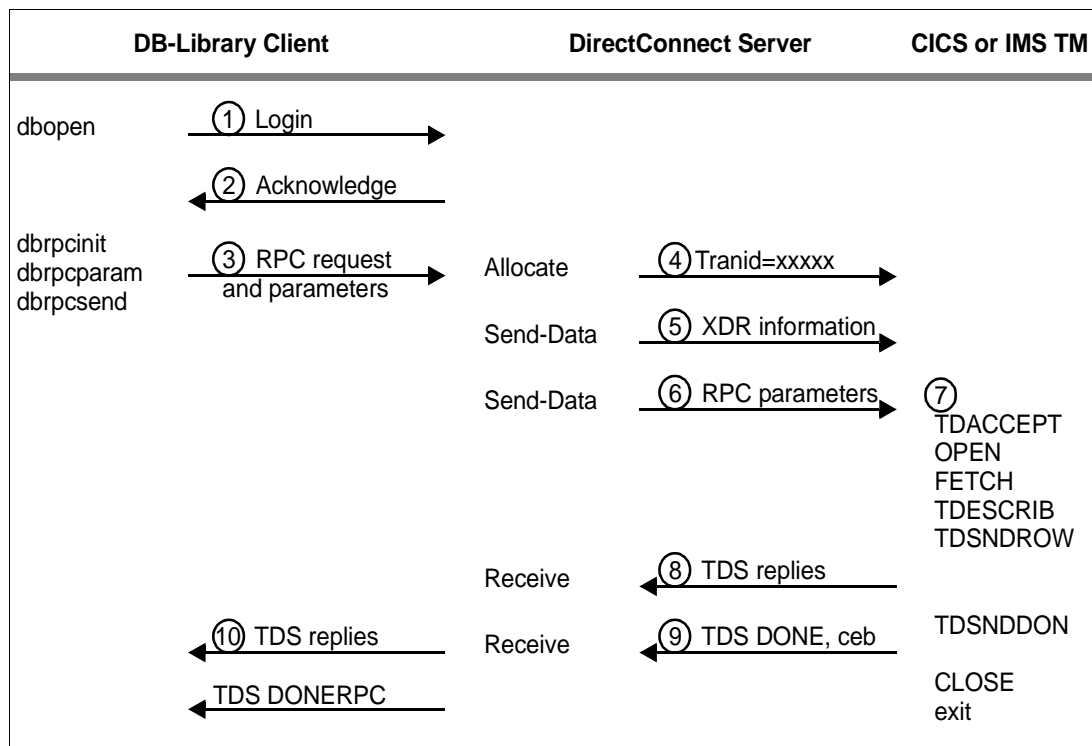
- Processing flow and requirements
- Process flow during attention sequences
- Browse applications

System administrators at the mainframe, TRS, and client need to coordinate troubleshooting efforts. To help you with your analysis, this section describes the processing flow from the client through TRS to the mainframe.

Processing flow and requirements

Figure C-1 shows the processing flow:

Figure C-1: Client-to-TRS-to-mainframe processing flow



These steps describe the sequence shown above and highlight the requirements:

- 1 With TRS started, the client opens a LAN connection to a designated DirectConnect for z/OS Option server and logs in. This message may appear:


```
Server name not found in interface file
```

 If so, make sure that:
 - The client interfaces file is set up correctly.
 - The client Sybase path variable (SYBASE) is defined correctly.
 - The DirectConnect for z/OS Option server is specified in the DSQUERY variable.
- 2 When it receives the client login information, the DirectConnect for z/OS Option checks security as follows:

- If security is enabled, the DirectConnect for z/OS Option ensures that the client is authorized. If the client is not authorized, this error appears:


```
Security Violation: Login denied (no login entry)
```
 - If the client is authorized or security is disabled, the DirectConnect for z/OS Option acknowledges the login.
- 3 When the client application needs to invoke an RPC or language request on the mainframe, the client sends a request to TRS over the logged-in LAN connection.
 - 4 TRS receives the request and performs a table look-up to find the mainframe session and the Server Option transaction ID to use. The RPC and connection must be in the table. If security is enabled, the client must be authorized to use the RPC and connection to the mainframe. If the table lookup and security check are successful, the line is up, and the session is active, TRS allocates a conversation with the named transaction.

If a failure occurs during this process, SNA Services writes one of the following error messages to both the TRS log and the client:

```
Security Violation: Access to RPC 'xxxx' denied.
```

The client is not authorized or is not listed correctly.

```
Request Rejected: No host connections are
available.
```

Connections to the mainframe are unavailable.

```
Request Rejected: Remote procedure 'xxxx' not
found.
```

The RPC name was entered incorrectly or the name is not in the lookup table.

- 5 TRS sends the client External Data Representation (XDR) information to the mainframe.
- 6 TRS sends the client RPC parameters to the mainframe, and then waits for a reply from the transaction.
- 7 On the mainframe, the transaction processor initiates the named transaction, and the transaction issues the Server Option Gateway-Library calls. These calls read the client XDR information and RPC parameters. The transaction also performs associated processing, such as issuing static SQL DB2 requests or reading VSAM or other database data.

- 8 The transaction issues Gateway-Library calls that send results back to the client. These calls perform required data conversions, generate the TDS reply data stream, and send out reply data.
- 9 TRS receives the TDS reply packet and forwards it to the client, which continues until the Server Option transaction issues a TDSNDDON call.

If a failure occurs during this process, the LAN SNA software writes an error message to the DirectConnect for z/OS Option server log. It also writes an “Unexpected EOF from Adaptive Server Enterprise” error message to the client. (The mainframe is acting as a Adaptive Server Enterprise.) Gateway-Library tracing functions, if in use, also record errors in this process.
- 10 When the request is complete, the transaction exits and the conversation terminates. A long-running transaction (also called a user-defined transaction) can remain active through multiple requests before the conversation ends. If a long-running transaction terminates before it should, determine whether appropriate client support is set up. For example:
 - The client may be set up to disconnect after invoking the transaction and before the transaction ends.
 - Adaptive Server Enterprise logs out after sending a client request and, therefore, does not support long-running transactions.

For more information on identifying problems, see “Common problems and suggested solutions” on page 75.

Process flow during attention sequences

Any of the following actions results in an attention sequence:

- Database-Library issues a `dbcancel()` command.
- An `isql` user cancels processing while the server is sending results.
- An APT program or form issues a `closesql` command.
- A Data Workbench user exits a form while the server is sending results.

When an attention sequence is issued, the process flow is as follows:

- 1 Database-Library issues an attention packet to TRS, then discards anything else received until it receives a TDS DONE packet with the attention Ack bit on.

- 2 TRS converts the attention packet into a SNA SIGNAL command, issuing an LU 6.2 request-to-send verb. TRS then discards any results received from the mainframe until it receives a TDS DONE packet with the attention Ack bit on.
- 3 At the mainframe, IMS TM receives the SIGNAL and informs Server Option support.
- 4 Gateway-Library passes back a return code, indicating TDS_CANCEL_RECEIVED, on all subsequent TDSNDROW, TDSNDMSG, and TDSETPRM calls from an application. Any data associated with TDSNDROW or TDSNDMSG calls is discarded until the application issues a TDSNDDON call.

For details on these calls, see the appropriate Mainframe Connect Server Option *Programmers Reference*. PL/1 and COBOL versions of this guide are available.

- 5 When the application issues a TDSNDDON call, the Server Option support sends a TDS DONE packet with the attention Ack bit on. This ends the attention sequence.

When an attention sequence is issued, the process flow is as follows:

- 1 Database-Library issues an attention packet to TRS, then discards anything else received until it receives a TDS DONE packet with the attention Ack bit on.
- 2 TRS converts the attention packet into a SNA SIGNAL command, issuing an LU 6.2 request-to-send verb. TRS then discards any results received from the mainframe until it receives a TDS DONE packet with the attention Ack bit on.
- 3 At the mainframe, IMS TM receives the SIGNAL and informs the Server Option support.
- 4 Gateway-Library passes back a return code, indicating TDS_CANCEL_RECEIVED, on all subsequent TDSNDROW, TDSNDMSG, and TDSETPRM calls from an application. Any data associated with TDSNDROW or TDSNDMSG calls is discarded until the application issues a TDSNDDON call.

For details on these calls, see the appropriate Mainframe Connect Server Option *Programmers Reference*. PL/1 and COBOL versions of this guide are available.

- 5 When the application issues a TDSNDDON call, the Server Option support sends a TDS DONE packet with the attention Ack bit on. This ends the attention sequence.

Browse applications

Sybase architecture uses a “streaming mode” of data transfer. Rather than sending a short block of data and waiting for a reply, the mainframe continuously sends data until the client stops accepting it. When the client stops accepting data, normal SNA pacing functions suspend data transfer.

For applications that select a small set of data, process it, then request the next block of data, it is best to use RPC parameters to specify the ID of a set of records. If the client RPC parameters are set up as return parameters, and the Gateway-Library TDSETPRM specifies the ID of the desired set of records, Server Option support returns the updated RPC value to the client. The client can use this value to invoke the next set of records.

Where to start troubleshooting

Note For troubleshooting information about TRS, see the Mainframe Connect DirectConnect for z/OS Option *Users Guide for Transaction Router Services*. For explanations of specific error messages, see the Mainframe Connect Client Option and Server Option *Messages and Codes*.

At the client, the DirectConnect for z/OS Option workstation, and the mainframe levels, check components systematically to locate the problem. Depending on your setup, you may want to check for problems in this sequence:

- 1 Connectivity
- 2 Client application
- 3 Client LAN
- 4 Client network setup
- 5 Major outage

- 6 DirectConnect for z/OS Option workstation
- 7 Connection from the client to the DirectConnect for z/OS Option workstation
- 8 Connection from the DirectConnect for z/OS Option workstation to the mainframe
- 9 Configuration between the transaction processor and the Server Option or Mainframe Connect DB2 UDB Option for IMS TM and MVS

For any of these problems, the appropriate system administrator should use normal troubleshooting procedures. For example:

- Record specific information on the error message(s), including:
 - Error message number
 - Associated SNA sense codes or SNA Services error codes
 - Time the error occurred
 - The client or user affected
- Perform the recommended action.
- Continue the process until the problem is resolved.

Common problems and suggested solutions

Problems can often be traced to configuration errors or to network, line, modem, or adaptor outages.

This section contains the following subsections:

- Configuration errors
- Mainframe network operational failure
- Network session or line failures

Configuration errors

This section contains the following sub-sections:

- Cannot establish session

- Session established, but the transaction does not run
- SDLC line or Token-Ring not up
- SDLC Link and PU are active, but the LU is not active

Configuration errors are often the cause of communications failure. To resolve these errors, you need the following information, which was created when the network was installed and successfully implemented:

- For IMS TM, definitions for transaction name and APPLID
- SNA /NCP definitions for the LU and associated logmode
- SDLC or token-ring connection charts to the mainframe
- Sybase interface files for clients and TRS
- Sybase security definitions, including client logins, connection groups, and transaction groups

Verify that this information is the same as it was before the error occurred. If it is not, determine whether a recent change is contributing to the problem.

Cannot establish session

Cause

Any of the following:

- Mismatched LU definitions between SNA and workstation
- Mismatched modenames
- Incorrect SNA MODETAB and APPLID macros

Suggested action

Check the MVS system log on the mainframe for messages.

Correct the spelling.

Coordinate with the TRS administrator to check connection and modename profiles, using the utility shipped with the product.

Session established, but the transaction does not run

Cause

Any of the following:

- RACF security error
- Incorrect transaction ID in the TRS RPC table

Suggested action

Check the MVS system log on the mainframe for messages.

Verify definitions.

Coordinate with the TRS administrator for correct security and transaction ID setups.

SDLC line or Token-Ring not up

Cause	Address incorrectly configured with NCP (assumes correct line or modem setup)
Suggested action	Check both ends of the SDLC station or Token-Ring address configuration.

SDLC Link and PU are active, but the LU is not active

Cause	One of the following: <ul style="list-style-type: none">• SNA and DirectConnect for z/OS Option LU definition errors• SSCPID value in the local LU profile set incorrectly
Suggested action	Use the SDLC trace and error log facilities to find the error.

Mainframe network operational failure

On the mainframe, there are two frequent causes of operational errors:

- The IMS TM or SNA operator put the resource out of service with the vary command.
- SNA placed the line, physical unit (PU), or LU into a non-operating (INOP) state because of a network outage.

In these cases, either:

- The TRS administrator sees SNA Services timeout and connection failure messages when he or she tries to start the DirectConnect for z/OS Option, or
- The requesting client sees an SNA Services message indicating that the system could not start the RPC.

When you are contacted about such messages, reactivate the necessary mainframe resources.

Network session or line failures

This section explains what happens when line, adapter, or modem outages occur and how to help prevent them.

When these errors occur

Line, adapter, or modem outages result in error messages at the SNA console and at the DirectConnect for z/OS Option. The DirectConnect for z/OS Option records the message and, when possible, sends a similar error message to any affected clients.

Preventing these errors

Intermittent hardware errors and line degradation problems disrupt processing and may be difficult to find. It helps to check periodically for these problems. For example:

- To check for hardware errors, use the SNA error logs. Report errors to IBM Service.
- To check for line degradation, use SNA to periodically report the SDLC line statistics. Examine the statistics for a significant number of re-transmissions or idle detect timeouts. Line degradation results in random SDLC line failures or very slow response to the client, even during a moderate processing load.

Troubleshooting at each component

This section explains troubleshooting at each of the Sybase components. It includes the following subsections:

- Server Option support
- TRS support
- DirectConnect for z/OS Option communications with the mainframe
- Gateway-Library support

Server Option support

Server Option support consists of several components on the IBM S/390 mainframe and the DirectConnect for z/OS Option platform. These components provide tracing and logging, which you can use to locate errors.

TRS support

TRS does the following:

- Receives requests from client applications
- Converts the requests to the appropriate communications protocol call
- Sends the requests to the mainframe

Each instance of TRS has a unique service name, which clients use to select a service for communication. Each instance of TRS has its own set of configuration information, defined globally in the Transaction Router Service Library (TRSL) configuration file.

TRS uses the following files:

- *<srvlibname>.tds* for tracing Sybase TDS traffic between TRS on UNIX platforms and mainframe SNA
- *ngtds.<srvlibname>* for tracing Sybase TDS traffic between TRS on PC platforms and mainframe SNA
- *svr.log* for logging TDS traffic between TRS and client workstations, and for recording errors

The Transaction Router Service Library (TRSL) *SNATraceFile* configuration property specifies the file to which the SNA side of the TRS trace file is written. Formatted TDS traces and errors are logged and written to this file.

The associated TRSL name is appended to these files so that the TRS administrator can differentiate the log for each instance of TRS. For more information, refer to the *Mainframe Connect DirectConnect for z/OS Option Users Guide for Transaction Router Services*.

The DirectConnect for z/OS Option server logs TDS traffic between TRS and client workstations and records errors. For more information, see the *Enterprise Connect Data Access and Mainframe Connect Server Administration Guide for DirectConnect*.

DirectConnect for z/OS Option communications with the mainframe

This section contains the following subsections:

- SNA LU 6.2
- IBM

TRS depends on the communications support of the server it runs on to communicate with the mainframe transaction processor. For IMS MVS, DirectConnect uses SNA LU 6.2.

SNA LU 6.2

The LAN communications server, such as SNA Services for AIX, uses the SNA trace file to record SDLC/SNA traffic between the workstation and mainframe. The vendor's trace utility extracts this file.

For AIX and OS/2 platforms, the error log file records errors that SNA Services detects or Communications Manager detects. The IBM error log report utility extracts this information.

IBM

For IBM environments, debug or trace sockets are used. Refer to your IBM documentation for more information.

Gateway-Library support

The Gateway-Library is a set of functions available for writing applications to enable mainframe environments to communicate with clients attached to TRS. These functions convert client calls into the TDS datastream needed to communicate with TRS and its clients.

Stubs provide access to the Gateway-Library functions. These stubs are a set of object libraries that application programmers can include in job steps used to link-edit programs they create.

Gateway-Library tracing functions enable you to trace program activity globally, for all transactions, or specifically, for individual transactions. Based on the transaction processor, tracing functions provide:

- API tracing for Gateway-Library calls, using the IMS TM System Log
- TDS header tracing, using the IMS TM System Log
- TDS data tracing, using the IMS TM System Log

Table C-1 shows the tracing functions:

Table C-1: Gateway-Library tracing functions

Function	Description
TDINFLOG	Determines what types of tracing are set
TDINFSPT	Indicates whether tracing is on or off for a transaction and returns the transaction ID
TDLSTSPT	Lists all transactions for which tracing is enabled
TDSETLOG	Turns system-wide tracing options on or off
TDSETSPT	Turns tracing on or off for a specific transaction
TDWRTLOG	Writes a user message or system entry

You can use standard IMS TM or MVS debugging tools or third party debugging tools to debug user-written applications.

For more information, see:

- Chapter 5, “Tracing and Accounting,” which describes the logging processes.
- The appropriate Mainframe Connect Server Option *Programmers Reference*, which describes Gateway-Library tracing functions. PL/1 and COBOL versions of this guide are available.

Remember that some TDS calls fill up internal TDS buffers before sending them out to the network. For example, a TDSNDROW or TDSNDMSG call does not cause execution of a corresponding CICS EXEC SEND call unless the TDS buffer becomes full.

Warning! To avoid losing records, periodically archive or delete the trace records on SYTDLOG1. Trace records are appended to this file until it is full; then, the records are rejected.

Glossary

accept	Establishment of a SNA or TCP/IP connection between Mainframe Connect Server Option and Mainframe Connect DirectConnect for z/OS Option.
access service	The named set of properties, used with an access service library, to which clients connect. Each DirectConnect server can have multiple services.
access code	A number or binary code assigned to programs, documents, or folders that allows authorized users to access them.
access service library	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	See access service library .
Adaptive Server Enterprise (ASE)	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.
Adaptive Server Enterprise/Component Integration Services	Includes a variation of ASE that provides a Transact-SQL interface to various sources of external data. Component Integration Services allows ASE to present a uniform view of enterprise data to client applications.
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 server management. See also remote procedure call , service library .
ADMLIB	See administrative service library .
Advanced Interactive Executive	The IBM implementation of the UNIX operating system. The RISC System/6000, among other workstations, runs the AIX operating system.
advanced program-to-program communication	Hardware and software that characterize the LU 6.2 architecture and its implementations in products. See also logical unit 6.2 .

AIX	See Advanced Interactive Executive .
AMD2	The component of the Mainframe Connect DB2 UDB Option that allows clients to submit SQL statements to DB2 UDB. It is a CICS transaction that receives SQL statements sent from Mainframe Connect DirectConnect for z/OS Option and submits them to DB2 UDB, using the DB2 UDB dynamic SQL facility. It also receives the results and messages from DB2 UDB and returns them to Mainframe Connect DirectConnect for z/OS Option.
American Standard Code for Information Interchange	The standard code used for information interchange among data processing systems, data communication systems, and associated equipment. The code uses a coded character set consisting of 7-bit coded characters (including a parity check, 8 bits).
API	See application program interface .
APPC	See advanced program-to-program communication .
application program interface	The programming language interface between the user and Mainframe Connect Client Option or Mainframe Connect Server Option. The API for Mainframe Connect Client Option is Client-Library. The API for Mainframe Connect Server Option is Gateway-Library.
ASCII	See American Standard Code for Information Interchange .
ASE	See Adaptive Server Enterprise .
ASE/CIS	See Adaptive Server Enterprise/Component Integration Services .
batch	A group of records or data processing jobs brought together for processing or transmission.
bind	In the Sybase environment, this term has different meanings depending on the context: <ul style="list-style-type: none">• In CICS, it is an SNA command used to establish a connection between LUs, or a TCP/IP call that connects an application to a port on its system.• In DB2 UDB, it compiles the Database Request Module, the precompiler product that contains SQL statements in the incoming request, and produces an access plan, a machine code version of the SQL statements that specifies the optimal access strategy for each statement.• In the mainframe access product set, it establishes a connection between a TRS port and a CICS or IMS region.

bulk copy transfer	A transfer method in which multiple rows of data are inserted into a table in the target database. Compare with destination-template transfer and express transfer .
call level interface	A programming style that calls database functions directly from the top level of the code. Contrast with embedded SQL .
catalog	A system table that contains information about objects in a database, such as tables, views, columns, and authorizations.
catalog RPC	A component of the Mainframe Connect DB2 UDB Option that allows clients to access DB2 UDB system catalogs. It uses an interface compatible with the catalog interface for the ODBC API.
catalog stored procedure	A procedure used in SQL generation and application development that provides information about tables, columns, and authorizations.
character set	A set of specific (usually standardized) characters with an encoding scheme that uniquely defines each character. ASCII is a common character set.
CICS	See Customer Information Control System .
CICS region	The instance of CICS.
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 responsible for the user interface that sends requests to applications acting as servers. See also client/server .
Client-Library	A library of routines that is part of Mainframe Connect Client Option.
client request	An RPC or language request sent by a client to a server.
client/server	An architecture in which the client is an application that handles the user interface and local data manipulation functions, and the server is an application providing data processing access and management. See also client application .
Client Services Application	A customer-written CICS program initiated on the host that uses the API to invoke the Mainframe Connect Client Option as a client to the The Server Option server or to ASE. See also application program interface, Client Services for CICS .

Client Services for CICS	A Sybase host API that invokes the Mainframe Connect Server Option as a client to an access service for DB2 UDB or ASE. See also application program interface, Customer Information Control System, Client Services Application, Mainframe Connect Server Option.
clustered index	An index in which the physical order and the logical (indexed) order is the same. Compare with nonclustered index.
code page	An assignment of graphic characters and control function meanings to all code points.
commit	A process that makes permanent all changes made to one or more database files since the initiation of the application program, the start of an interactive session, or the last commit or rollback operation. Compare with rollback.
Common Programming Interface	Specifies the languages and services used to develop applications across SAA environments. The elements of the CPI specification are divided into two parts: processing logic and services.
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.
conversion	The transformation between values that represent the same data item but which belong to different datatypes. Information can be lost due to conversion, because accuracy of data representation varies among different datatypes.
connection	A network path between two systems. For SNA, the path connects a logical unit (LU) on one machine to an LU on a separate machine. For TCP/IP, the path connects TCP modules on separate machines.
connection router	A program provided with Mainframe Connect Client Option that directs requests to particular remote servers. Mainframe system programmers use the connection router to define remote servers and server connections to Mainframe Connect Client Option.
Connection Router Table	A memory-resident table maintained by a Mainframe Connect Client Option system programmer that lists servers and the connections that a Client-Library transaction can use to access them.
control section	The part of a program specified by the programmer to be a relocatable unit, all elements of which are to be loaded into adjoining main storage locations.
control statement	In programming languages, a statement that is used to alter the continuous sequential execution of statements. A control statement can be a conditional statement or an imperative statement.

conversation-level security	The passing of client login information to the mainframe by TRS when it allocates a conversation.
CSA	See Client Services Application .
CSP	See catalog stored procedure .
cursor	In SQL, a named control structure used by an application program to point to a row of data.
Customer Information Control System	An IBM licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs.
DASD	See direct access storage device .
data definition statement	An IBM mainframe statement used to relate a name with a file.
data definition language	A language for describing data and data relationships in a database.
data set name	The term or phrase used to identify a data set.
database management system	The term or phrase to identify a data set. A computer-based system for defining, creating, manipulating, controlling, managing, and using databases.
database operation	A single action against the database. For Mainframe Connect DirectConnect for z/OS Option, a database operation is usually a single SQL statement. One or more database actions can be grouped together to form a request. See also request .
Database 2	An IBM relational database management system.
datatype	A keyword that identifies the characteristics of stored information on a computer.
DB-Library	A Sybase and Microsoft API that allows client applications to interact with ODS applications. See also application program interface .
DBMS	See database management system .
DB2 UDB	See Database 2 .
DB2 UDB Option	A Sybase mainframe solution that provides dynamic access to DB2 UDB data.
DDL	See data definition language .
DD statement	See data definition statement .

default language	The language that displays a user's prompts and messages.
destination-template transfer	A transfer method in which source data is briefly put into a template where the user can specify that some action be performed on it before execution against a target database. See also transfer . Compare with bulk copy transfer and express transfer .
direct access storage device	A device in which access time is effectively independent of the location of the data.
direct request	A request sent directly from a client workstation through Transaction Router Service to the DirectConnect server without going through ASE. Contrast with indirect request .
direct resolution	A type of service name resolution that relies upon a client application specifying the exact name of the service to be used. See also service name resolution . Compare with service name redirection .
DirectConnect for z/OS Option	A Sybase Open Server application that provides access management for non-Sybase databases, copy management (transfer), and remote systems management. Each DirectConnect for z/OS Option consists of a server and one or more service libraries to provide access to a specific data source.
DirectConnect Manager	A Java application from Sybase that can be used in Windows and UNIX environments. It provides remote management capabilities for DirectConnect products, including starting, stopping, creating, and copying services.
DirectConnect server	The component of Mainframe Connect DirectConnect for z/OS Option that provides general management and support functions to service libraries.
dll	See dynamic link library .
DSN	See data set name .
dynamic link library	A file containing executable code and data bound to a program at load time or runtime, rather than during linking.
dynamic SQL	The preparation and processing of SQL source statements within a program while the program runs. The SQL source statements are contained in host-language variables rather than being coded directly into the application program. Contrast with static SQL .
ECDA	See Enterprise Connect Data Access .

ECDA Option for ODBC	A Sybase solution that allows client applications to access ODBC data. It combines the functionality of the ECDA Option for ODBC architecture with ODBC to provide dynamic SQL access to target data, as well as the ability to support stored procedures and text and image pointers.
ECDA Option for Oracle	A Sybase solution that provides Open Client access to Oracle databases. When used in combination with ASE, it provides many of the features of a distributed database system, such as location transparency, copy transparency, and distributed joins.
embedded SQL	SQL statements that are embedded within a program and are prepared in the process before the program runs. After it is prepared, the statement itself does not change, although values of host variables specified within the statement might change.
end user	A person who connects to a DirectConnect server using an application to access databases and perform transfers. See also transfer .
Enterprise Connect Data Access	An integrated set of software applications and connectivity tools that allow access to data within a heterogeneous database environment, such as a variety of LAN-based, non-Sybase data sources, as well as mainframe data sources.
environment variable	A variable that describes how an operating system runs and the devices it recognizes.
exit routine	A user-written routine that receives control at predefined user exit points.
express transfer	A form of bulk copy transfer that uses ODBC bulk APIs to improve performance when transferring bulk data between data sources. Because it uses the same syntax as bulk copy transfer, no modification of applications is required.
external call interface	A CICS client facility that allows a program to call a CICS application as if the calling program had been linked synchronously from a previous program instead of started from a terminal.
External Security Manager	An add-on security package for the z/OS mainframe, licensed by Computer Associates.
FCT	See forms control table .
forms control table	An object that contains the special processing requirements for output data streams received from a host system by a remote session.
gateway	Connectivity software that allows two or more computer systems with different network architectures to communicate.

Gateway-Library	A library of communication, conversion, tracing, and accounting functions supplied with Mainframe Connect Server Option.
globalization	The combination of internationalization and localization. See internationalization , localization .
global variable	A variable defined in one portion of a computer program and used in at least one other portion of the computer program. Contrast with local variable .
handler	A routine that controls a program's reaction to specific external events, for example, an interrupt handler.
host	The mainframe or other machine on which a database, an application, or a program resides. In TCP/IP, this is any system that is associated with at least one Internet address. See also Transmission Control Protocol/Internet Protocol .
host ID	In Mainframe Connect Server Option, the ID that the TRS passes to the mainframe with a client request. The host ID is part of the client login definition at the TRS.
host password	In Mainframe Connect Server Option, the password that the client passes to the mainframe with a client request.
host request library	A DB2 UDB table that contains host-resident SQL statements that can be executed dynamically. See also host-resident request .
host-resident request	A SQL request that resides in a DB2 UDB table called the host request library. See also host request library .
IMS	See Information Management System .
indirect request	A client request that is routed through a stored procedure on a SQL Server, which forwards the request to TRS as an RPC. Compare with direct request .
Information Management System	A database/data communication system that can manage complex databases and networks.
interfaces file	An operating system file that determines how the host client software connects to a Sybase product. An <i>interfaces</i> file entry contains the name of any The Server Option server and a list of services provided by that server.
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 .

keyword	A word or phrase reserved for exclusive use by Transact-SQL.
language RPC	The name TRS uses to represent a client's language request. TRS treats a language request as a remote procedure call (RPC) and maps it to a language transaction at the remote server.
language transaction	The server transaction that processes client language requests. The Mainframe Connect DB2 UDB Option language transaction for CICS is AMD2, which uses the DB2 UDB dynamic SQL facilities to process incoming SQL strings. The Mainframe Connect DB2 UDB Option for IMS uses SYRT by default.
linkage	In computer security, combining data or information from one information system with data or information from another system with the intention to derive additional information; for example, the combination of computer files from two or more sources.
linkage editor	A computer program that creates load modules from one or more object modules or creates load modules by resolving cross references among the modules, and if necessary, adjusts those addresses.
link-edit	To create a loadable computer program by using a linkage editor. See also linkage editor .
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 .
local variable	A variable that is defined and used only in one specified portion of a computer program. Contrast with global variable .
logical unit	A type of network addressable unit that enables a network user to gain access to network facilities and communicate remotely. A connection between a TRS and a CICS region is a connection between logical units.
logical unit 6.2	A type of logical unit that supports general communication between programs in a distributed processing environment. See also advanced program-to-program communication .
login ID	In Mainframe Connect Server Option, the ID that a client user uses to log in to the system.
login packet	Client information made available to Mainframe Connect Server Option. The client program sets this information in a login packet and sends it to TRS, which forwards it to the mainframe.

long-running transaction	A transaction that accepts more than one client request. Whereas short transactions end the communication after returning results to a client, a long-running transaction can await and process another request. Compare with short transaction .
LU 6.2	See logical unit 6.2 .
mainframe access products	Sybase products that enable client applications to communicate with mainframes in a client/server environment. See client/server .
Mainframe Connect	The Sybase product set that provides access to mainframe data.
Mainframe Connect Client Option	A Sybase product that, using Client-Library, allows mainframe clients to send requests to SQL Server, Open Server, the Mainframe Connect DB2 UDB Option and Mainframe Connect Server Option. Mainframe Connect Client Option provides capability for the mainframe to act as a client to LAN-based resources in the CICS or the IMS and MVS environment.
Mainframe Connect DB2 UDB Option	A Sybase mainframe solution that provides dynamic access to DB2 UDB data. It is available in the CICS or IMS environment. See also Customer Information Control System, Database 2, Multiple Virtual Storage .
Mainframe Connect The Server Option for z/OS Option	A Sybase Open Server application that provides access management for non-Sybase databases, copy management (transfer), and remote systems management.
Mainframe Connect Server Option	A Sybase product that provides capability for programmatic access to mainframe data. It allows workstation-based clients to execute customer-written mainframe transactions remotely. It is available for the CICS and the IMS and MVS environments
Multiple Virtual Storage	An IBM operating system that runs on most System/370 and System/390 mainframes. It supports 24-bit addressing up to 16 megabytes.
network protocol	A set of rules governing the way computers communicate on a network.
nonclustered index	An index that stores key values and pointers to data. Compare with clustered index .
null	Having no explicitly assigned value. NULL is not equivalent to 0 or to blank.
ODBC	See Open Database Connectivity .
ODS	See Open Data Services .

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 Data Services	A product that provides a framework for creating server applications that respond to DB-Library clients.
Open Database Connectivity	A Microsoft API that allows access to both relational and non-relational databases. See also application program interface .
Open Server	A Sybase product that provides the tools and interfaces required to create a custom server. Clients can route requests to the The Server Option server through an Open Server configured to meet specific needs, such as the preprocessing of SQL statements.
parameter	A variable that is given a constant value for a specified application and can denote the application. Compare with property .
Partner Certification Reports	Sybase publications that certify third-party or Sybase products to work with other Sybase products.
Password Expiration Management	An IBM password management program with CICS Version 3.3 through an optional program temporary fix, and as an integral part of CICS with version 4.1 and higher.
PEM	See Password Expiration Management .
PL/1	See Programming Language /1 .
primary database	The database management system that the DirectConnect server is always connected to. It is implied in the transfer statement.
Programming Language/1	A programming language designed for use in a wide range of commercial and scientific computer applications.
property	A setting for a server or service that defines the characteristics of the service, such as how events are logged. Compare with parameter .
protocol	The rules for requests and responses used to manage a network, transfer data, and synchronize the states of network components.
query	A request for data from a database, based upon specified conditions.
Registry	The part of the Windows operating system that holds configuration information for a particular machine.

relational database	A database in which data is viewed as being stored in tables consisting of columns (data items) and rows (units of information).
relational operators	Operators supported in search conditions.
relops	See relational operators .
remote procedure call	A call to execute a stored procedure on a remote server. For Mainframe Connect Server Option, an RPC is a direct request from a client to TRS. For Mainframe Connect Client Option, a Client-Library transaction that calls a procedure on a remote server acts like an RPC.
remote stored procedure	A customer-written CICS program using an API that resides on the mainframe and communicates with Mainframe Connect DB2 UDB Option. See also Customer Information Control System, stored procedure . 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.
Replication Server	A Sybase SQL Server application that maintains replicated data and processes data transactions received from a data source.
request	One or more database operations an application sends as a unit to the database. Depending upon the response, the application commits or rolls back the request. See also commit, rollback, unit of work .
resource table	A main storage table that associates each resource identifier with an external logical unit (LU) or application program.
rollback	An instruction to a database to back out of changes requested in a unit of work. Compare with commit .
router	An attaching device that connects two LAN segments, which use similar or different architectures, at the Open System Interconnection (OSI) reference model network layer. Contrast with gateway .
RPC	See remote procedure call .
RSP	See remote stored procedure .
SAA	See System Application Architecture .
secondary connection	The connection specified in the transfer statement. It represents anything that can be accessed using Mainframe Connect Client Option, such as ASE or another access service.

secondary database	In transfer processing, the supported database that is specified in the transfer statement. Compare with primary database .
server	A functional unit that provides shared services to workstations over a network. See also client/server . Compare with client .
server process ID	A positive integer that uniquely identifies a client connection to the server.
service	A functionality available in Mainframe Connect DirectConnect for z/OS Option. It is the pairing of a service library and a set of specific configuration properties.
service library	In Mainframe Connect DirectConnect for z/OS Option, a set of configuration properties that determine service functionality. 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 create an alternative mechanism to map connections with 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 service name redirection .
service name resolution	The DirectConnect server mapping of an incoming service name to an actual service. See also direct resolution , service name redirection .
session	A connection between two programs or processes. In APPC communications, sessions allow transaction programs to have conversations between the partner LUs. See also advanced program-to-program communication .
short transaction	A mainframe transaction that ends the communication when it finishes returning results to the client. Compare with long-running transaction .
SNA	See Systems Network Architecture .
SNRF	See service name redirection file .
SPID	See server process ID .
SQL	See structured query language .
SQLDA	See SQL descriptor area .
sqledit	A utility for creating and editing <i>sql.ini</i> files and file entries.

sql.ini	The interfaces file containing definitions for each The Server Option server to which a workstation can connect. The file must reside on every client machine that connects to ASE.
SQL descriptor area	A set of variables used in the processing of SQL statements.
SQL stored procedure	A single SQL statement that is statically bound to the database. See also stored procedure .
static SQL	SQL statements that are embedded within a program and prepared during the program preparation process before the program runs. Compare with dynamic SQL .
stored procedure	A collection of SQL statements and optional control-of-flow statements stored under a particular name. Adaptive Server stored procedures are called “system procedures.” See also remote stored procedure , system procedures .
structured query language	An IBM industry-standard language for processing data in a relational database.
stub	A program module that transfers remote procedure calls (RPCs) and responses between a client and a server.
SYRT	The component of Mainframe Connect DB2 UDB for IMS that allows clients to submit SQL language requests to DB2 through IMS.
System Administrator	The person in charge of server system administration, including installing and maintaining DirectConnect servers and service libraries.
System Application Architecture	An IBM proprietary plan for the logical structure, formats, protocols, and operational sequences for transmitting information units through networks and controlling network configuration and operation. See also advanced program-to-program communication .
system procedures	A stored procedure that ASE supplies for use in system administration. System procedures serve as shortcuts for retrieving information from system tables, or a mechanism for accomplishing database administration. See also stored procedure .
Systems Network Architecture	An IBM proprietary plan for the structure, formats, protocols, and operational sequences for transmitting information units through networks. See also advanced program-to-program communication .
table	An array of data or a named data object that contains a specific number of unordered rows. Each item in a row can be unambiguously identified by means of one or more arguments.

Tabular Data Stream	A Sybase application-level protocol that defines the form and content of relational database requests and replies.
target	A system, program, or device that interprets, rejects, satisfies, or replies to requests received from a source.
target database	The database to which the DirectConnect server transfers data or performs operations on specific data.
TCP/IP	See Transmission Control Protocol/Internet Protocol .
TDS	See Tabular Data Stream .
transaction	A unit of processing initiated by a single request. A transaction consists of one or more application programs that, when executed, accomplish a particular action. In Mainframe Connect Server Option, a client request (RPC or language request) invokes a mainframe transaction. In Mainframe Connect Client Option, a mainframe transaction executes a stored procedure on a remote server.
transaction processing	A sequence of operations on a database that is viewed by the user as a single, individual operation.
Transaction Router Service	A Mainframe Connect DirectConnect for z/OS Option program used when the mainframe acts as a transaction server to route requests from remote clients to the Mainframe Connect Server Option and return results to the clients.
Transaction Router Service library	A service library that facilitates access to remote transactions, allowing customers to execute transactions from virtually any mainframe data source. See also service library .
Transact-SQL	A Sybase-enhanced version of the SQL database language used to communicate with ASE.
transfer	A Mainframe Connect DirectConnect for z/OS Option feature that allows users to move data or copies of data from one database to another.
transfer service library	A service library that provides copy management functionality. See also service library .
Transmission Control Protocol/Internet Protocol	A set of communication protocols that supports peer-to-peer connectivity functions for both local and wide area networks.
trigger	A form of stored procedure that automatically executes when a user issues a change statement to a specified table.

TRS	See Transaction Router Service .
TRS library	See Transaction Router Service library .
T-SQL	See Transact-SQL .
unit of work	One or more database operations grouped under a commit or rollback. A unit of work ends when the application commits or rolls back a series of requests, or when the application terminates. See also commit , rollback , transaction .
user ID	User identification. The ID number by which a user is known in a specific database or system.
variable	An entity that is assigned a value. Mainframe Connect The Server Option for z/OS Option has two kinds of variables: <i>local</i> and <i>global</i> .
view	An alternate representation of data from one or more tables. A view can include all or some of the columns contained the table or tables on which it is defined.
Virtual Storage Access Method	An IBM-licensed program that controls communication and the flow of data in an SNA network.
Virtual Telecommunications Access Method	IBM mainframe software that allows communication on an SNA network between mainframes and allows the mainframe to have multiple sessions per connection.
VSAM	See Virtual Storage Access Method .
VTAM	See Virtual Telecommunications Access Method .
wildcard	A special character that represents a range of characters in a search pattern.

Index

A

- ACCESSCODE SYGWMCSST parameter 42
- ACCESSCODESW SYGWMCSST parameter 42
- accounting 37
 - at DirectConnect for z/OS Option 37
 - at the mainframe 37
 - at the mainframe using elapsed time 37
 - Server Option functions 37
 - TDACCEPT 37
 - TDFREE 37
 - TDINFACT 37
 - TDSETACT 37
 - where enabled 37
- accounting log
 - layout 38
 - under CICS 37
- API tracing 29
- APPC/IMS
 - security 24
- ASCII_8 translation tables
 - ASCII_8 ACSII-to-EBCDIC 60
 - ASCII_8 EBCDIC-to-ASCII 61
- ATBSDFMU exec for defining DBTOKEN 25
- audience vii
- authorization key
 - ordering 11

C

- CEDA window 52
- CHARSETSRV SYGWMCSST parameter 42
- CICS
 - network driver 51
- common problems, troubleshooting 75
 - mainframe network failure 77
 - session or line failure 77
- configuration errors, troubleshooting 76
- conversational security

- implementation 25
- coordinating troubleshooting efforts 69
- cp437 translation tables
 - cp437 ASCII-to-EBCDIC 65
 - cp437 EBCDIC-to-ASCII 66
- cp850 translation tables
 - cp850 ASCII-to-EBCDIC 67
 - cp850 EBCDIC-to-ASCII 68
- CPI-C CICS network driver 51
- customization options 41
 - SYGWDRIV 51
 - SYGWHOST, TCP/IP configuration macro 53
- customization table SWGWXCPH 41
 - SYGWMCSST global macro 41
 - SYGWMCSXL,character set macro 41
- customizing
 - a network driver 51
 - global 42
 - LAN-side character sets 53

D

- DASD space required 7
- DBTOKEN.X.TPNAME
 - RACF 25
- DEBUGSW SYGWMCSST parameter 42
- DECPOINT SYGWMCSST parameter 43
- DirectConnect for z/OS Option 3
 - communication 79, 80
 - troubleshooting 79
 - troubleshooting SNA LU 6.2 80
- DQUOTETRAN SYGWMCSST parameter 43
- dynamic network driver
 - CICS 51
 - CPI-C CICS 51
 - customizing 51
 - macro 51

E

external security systems
for IMS 23

F

files
accounting log 37

G

Gateway-Library
support 80
global customization (SYGWCST) 41
list of parameters 42
global tracing 29

H

how to use this book vii

I

IMSLOGTYPE SYGWCST parameter 43
ISO_1 translation tables
ISO_1 ASCII-to-EBCDIC 62
ISO_1 EBCDIC-to-ASCII 63
IxHost
in trace log 30

L

LAN-side character sets
customizing 53
library names
using new 10
licensing key
temporary 11
LogMode entry 26
LONGVARTRUNC SYGWCST parameter 43

M

macros
SWGWHOST 53
SYGWDRIV 51
mainframe character set customization options
(SYGWCXL) 45
overriding SBCS translation tables 46
SBCS, customizing translation 47
SBCS, predefined character sets 46
SBCS, user-defined character sets 46
mainframe network failure, troubleshooting 77
MVSDDNAME SYGWCST parameter 44

N

NATLANGUAGESRV SYGWCST parameter 44
network driver
choosing 5
CICS 51
CPI-C CICS 51
customizing 51
macro 51

O

ordering a permanent authorization key 11
overriding TRS security 23

P

parameters
SYGWCST macro 42
processing flow 69
browse applications 72, 74
during attention sequences 72
requirements 69

Q

querying the trace table 33

R

- RACF
 - setup 25
- related documents vii
- ROWLIMIT SYGWMCS parameter 44

S

- sample trace table 34, 35, 36
- SBCS translation tables 57
 - ASCII_8 59
 - cp437 63
 - cp850 66
 - ISO_1 62
- SBCS translation, customizing
 - overriding defaults with SYGWMCSXL 48
- security 21
 - APPC/IMS 24
 - components 21
 - conversational implementation 25
 - external 27
- security for CICS TCP/IP
 - external security systems 27
- security responsibilities
 - at Adaptive Server Enterprise 22
 - overriding TRS security at DirectConnect for z/OS
 - Option 23
- Server Option
 - accounting functions 37
 - description 1
- session or line failure, troubleshooting 78
 - preventive measures 78
- SNA LU 6.2 80
- space requirements 7
- specific tracing 29
- style conventions x
- SYGMCST 41
- SYGWDRIV macro 51
- SYGWHOST
 - formats 53
 - macro 53
- SYGWMCS parameters 42
 - ACCESSCODE 42
 - ACCESSCODESW 42
 - CHARSETSRV 42

- DEBUGSW 42
- DECPOINT 43
- DQUOTETRAN 43
- IMSLOGTYPE 43
- LONGVARTRUNC 43
- MVSDDNAME 44
- NATLANGUAGESRV 44
- ROWLIMIT 44
- USEIBMUNICODE 44
- SYGWMCSXL 45
- syntax conventions x

T

- Tabular Data Stream (TDS) 30
 - data tracing 29
 - header tracing 29
- temporary licensing key 11
- Temporary Storage (TS) queues 1
- trace lot
 - IxHost 30
- trace table for individual transactions 32
- tracing
 - accounting 29
 - TDINFLOG trace function 30
 - TDINFSPT trace function 30
 - TDLSTSPT trace function 30
 - TDSETLOG trace function 30
 - TDSETSPT trace function 30
 - TDWRTLOG trace function 30
 - using 32
 - using trace table for individual transactions 32
- tracing facility, using 32
- tracing walkthrough for specific transactions 33
- traffic logs 79
- transaction mapping 3
- Transient Data (TD) queues 1
- translation tables, SBCS 57
- troubleshooting 69
 - common problems 75
 - components 78
 - coordination 69
 - DirectConnect for z/OS Option communication with
 - mainframe 79
 - DirectConnect for z/OS Option support 79

Index

- Gateway-Library support 80
- mainframe network failure 77
- processing flow 69
- session or line failure 78
- where to start 74

TRS 3

- overriding security 23

U

- USEIBMUNICODE SYGWMCSST parameter 44
- using the tracing facility 32
 - trace table for individual transactions 32

V

- verify in VTAM LogMode 26

W

- where to start troubleshooting 74
 - existing environment 74
- windows
 - CEDA 52