## Contents

### About This Book ........................................................................................................................ vii

### CHAPTER 1 Understanding the Client Option ...................................................  1
- What is the Client Option? ........................................................................................................ 1
- Adaptive Server Enterprise (ASE) ..................................................................................... 2
- Server Option .................................................................................................................... 2
- Architecture ........................................................................................................................ 2
- Choosing a network driver ................................................................................................... 2
- General criteria for choosing a driver ................................................................................ 3
- Client Option communications ............................................................................................ 3
- Client Option environment .................................................................................................. 4
- Communication flow .......................................................................................................... 6
- Understanding TCP/IP SYGWXCPH definitions ............................................................... 7
- Client Option security ........................................................................................................ 7

### CHAPTER 2 Planning Your Installation ..............................................................  9
- Installation media ................................................................................................................ 9
- Pre-installation tasks ............................................................................................................ 9
  1. Verify the platforms, components, and distributed software ................................. 10
  2. Determine JCL and system information ...................................................................... 10
  3. Determine CICS and DB2 UDB information ................................................................. 10
  4. Determine compiler information .................................................................................. 11
  5. Determine Client Option information .......................................................................... 11
  6. Determine FTP information .......................................................................................... 12
  7. Plan the security requirements ..................................................................................... 12
  8. Identify the change control requirements ..................................................................... 12
  9. Back up the release libraries (upgrades only) ............................................................... 12
 10. Determine the library names ......................................................................................... 13
 11. Verify the connectivity .................................................................................................. 13

### CHAPTER 3 Installation and Configuration .....................................................  15
- Installation and configuration ............................................................................................. 15
- Libraries and samples ........................................................................................................ 21
### Contents

**CHAPTER 4**

*Mainframe Connect Client Option for CICS*

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introducing isql</td>
<td>23</td>
</tr>
<tr>
<td>Invoking isql and initializing the environment</td>
<td>23</td>
</tr>
<tr>
<td>Establishing the connection</td>
<td>24</td>
</tr>
<tr>
<td>Entering SQL commands</td>
<td>25</td>
</tr>
<tr>
<td>Viewing data results</td>
<td>26</td>
</tr>
<tr>
<td>Viewing trace results</td>
<td>27</td>
</tr>
</tbody>
</table>

**APPENDIX A**

*Customization Options*

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>29</td>
</tr>
<tr>
<td>Customizing global options (SYGWMCST)</td>
<td>30</td>
</tr>
<tr>
<td>Using the IBM z/OS conversion environment and services</td>
<td>33</td>
</tr>
<tr>
<td>Customizing mainframe character set conversion options (SYGWMCXL)</td>
<td>33</td>
</tr>
<tr>
<td>Overriding the supplied SBCS translation tables</td>
<td>34</td>
</tr>
<tr>
<td>Defining new SBCS translation tables</td>
<td>35</td>
</tr>
<tr>
<td>Defining new character set entries</td>
<td>38</td>
</tr>
<tr>
<td>Customizing dynamic network drivers (SYGWDRIV)</td>
<td>39</td>
</tr>
<tr>
<td>CICS network drivers</td>
<td>39</td>
</tr>
<tr>
<td>Customizing the TCP/IP driver (SYGWHOST)</td>
<td>41</td>
</tr>
<tr>
<td>Macro formats</td>
<td>41</td>
</tr>
<tr>
<td>Macro parameters</td>
<td>41</td>
</tr>
<tr>
<td>Defining license keys (SYGWLKEY)</td>
<td>42</td>
</tr>
<tr>
<td>Building a global customization module (SYGWXCPH)</td>
<td>43</td>
</tr>
</tbody>
</table>

**APPENDIX B**

*Translation Tables*

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understanding the ASCII-EBCDIC and EBCDIC-ASCII translation tables</td>
<td>45</td>
</tr>
<tr>
<td>Default ASCII_8 translation tables</td>
<td>47</td>
</tr>
<tr>
<td>ASCII_8, ASCII-to-EBCDIC translation table</td>
<td>48</td>
</tr>
<tr>
<td>ASCII_8, EBCDIC-to-ASCII translation table</td>
<td>48</td>
</tr>
<tr>
<td>Default ISO_1 translation tables</td>
<td>50</td>
</tr>
<tr>
<td>ISO_1 ASCII-to-EBCDIC translation table</td>
<td>51</td>
</tr>
<tr>
<td>ISO_1 EBCDIC-to-ASCII translation table</td>
<td>51</td>
</tr>
<tr>
<td>Default cp437 (code page 437) translation tables</td>
<td>53</td>
</tr>
<tr>
<td>cp437 ASCII-to-EBCDIC translation table</td>
<td>54</td>
</tr>
<tr>
<td>cp437 EBCDIC-to-ASCII translation table</td>
<td>55</td>
</tr>
<tr>
<td>Default cp850 (code page 850) translation tables</td>
<td>55</td>
</tr>
<tr>
<td>cp850 ASCII-to-EBCDIC translation table</td>
<td>57</td>
</tr>
<tr>
<td>cp850 EBCDIC-to-ASCII translation table</td>
<td>58</td>
</tr>
</tbody>
</table>

**APPENDIX C**

*Setting Up the CICS Sockets Interface*

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mainframe Connect Client Option for CICS</td>
<td>59</td>
</tr>
</tbody>
</table>
Contents

Understanding the CICS sockets interface ................................. 59
Installing and configuring the Client Option of CICS sockets interface 60

APPENDIX D

Setting Up Secure Sockets Layer Protocol ................................. 61
Understanding Secure Sockets Layer ........................................ 61
Description of SSL features .................................................. 61
How SSL provides security .................................................... 62
Setting up SSL in Sybase products ........................................... 63
  Setting up SSL in ASE and Open Client ............................... 63
Setting up SSL in IBM z/OS .................................................... 65
  Using System SSL on z/OS ................................................. 66
  Configuring AT-TLS .......................................................... 67
AT-TLS support in Client and Server Options for CICS ............... 70
  Configuring a z/OS client or server system .......................... 71
For more information ......................................................... 78

Glossary ...................................................................................... 79

Index .......................................................................................... 95
About This Book

The Mainframe Connect™ Client Option for CICS Installation and Administration Guide describes how to install and configure the Client Option for CICS. It includes planning considerations, installation instructions, configuration instructions, and troubleshooting information.

**Note** If you want to go directly to the installation planning tasks, 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/OS mainframe computer. This book refers to anyone performing these tasks as the Client 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 where to find the information you need in this book:

<table>
<thead>
<tr>
<th>To</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Understand</strong> the Client Option</td>
<td>Chapter 1, “Understanding the Client Option”</td>
</tr>
<tr>
<td><strong>Plan for installation and configuration</strong></td>
<td>Chapter 2, “Planning Your Installation.”</td>
</tr>
<tr>
<td><strong>Install</strong> the Client Option</td>
<td>Chapter 3, “Installation and Configuration”</td>
</tr>
<tr>
<td><strong>Access</strong> a server or host-based database using the Interactive SQL utility (isql)</td>
<td>Chapter 4, “Using isql”</td>
</tr>
<tr>
<td><strong>Customize</strong> the Client Option</td>
<td>Appendix A, “Customization Options”</td>
</tr>
<tr>
<td><strong>Reference</strong> default translation tables to customize SBCSs</td>
<td>Appendix B, “Translation Tables”</td>
</tr>
<tr>
<td><strong>Install and configure</strong> the CICS sockets interface</td>
<td>Appendix C, “Setting Up the CICS Sockets Interface”</td>
</tr>
<tr>
<td><strong>Install and configure</strong> the Secure Sockets Layer (SSL) protocol</td>
<td>Appendix D, “Setting Up Secure Sockets Layer Protocol”</td>
</tr>
</tbody>
</table>
Related documents
You may also need to refer to this related documentation:

- Mainframe Connect Client Option and Server Option *Messages and Codes*
- Mainframe Connect Client Option *Programmers Reference for C*
- Mainframe Connect Client Option *Programmers Reference for COBOL*
- Mainframe Connect Client Option *Programmers Reference for PL/1*
- Mainframe Connect Client Option *Programmers Reference for Client Services Applications*

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
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.
2. Click MySybase and create a MySybase profile.

Finding the latest information on EBFs and software maintenance
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.
The Client Option uses 8-character function names, while other versions of Client-Library™ use longer names. This book uses the long version of Client-Library names with one exception: the 8-character version is used in syntax statements. For example, CTBCMDPROPS has 11 letters. In the syntax statement, it is written CTBCMDPR, using 8 characters. You can use either version in your code.

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

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( )</td>
<td>Parentheses indicate that parentheses are included as part of the command.</td>
</tr>
<tr>
<td>{ }</td>
<td>Braces indicate that you must choose at least one of the enclosed options. Do not type the braces when you type the option.</td>
</tr>
<tr>
<td>[ ]</td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>The vertical bar indicates that you can select only one of the options shown. Do not type the bar in your command.</td>
</tr>
<tr>
<td>,</td>
<td>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.</td>
</tr>
</tbody>
</table>

The following table explains style conventions used in this book.

<table>
<thead>
<tr>
<th>This type of information</th>
<th>Looks like this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gateway-Library function names</td>
<td>TDINIT, TDRESULT</td>
</tr>
<tr>
<td>Client-Library function names</td>
<td>CTBINIT, CTBRESULTS</td>
</tr>
<tr>
<td>Other executables (DB-Library™ routines, SQL commands) in text</td>
<td>the dbrpcparam routine, a select statement</td>
</tr>
<tr>
<td>Directory names, path names, and file names</td>
<td>/usr/bin directory, interfaces file</td>
</tr>
<tr>
<td>Variables</td>
<td>n bytes</td>
</tr>
<tr>
<td>Adaptive Server® Enterprise datatypes</td>
<td>datetime, float</td>
</tr>
<tr>
<td>Sample code</td>
<td>01 BUFFER PIC S9(9) COMP SYNC.</td>
</tr>
<tr>
<td>User input</td>
<td>01 BUFFER PIC X(n)</td>
</tr>
<tr>
<td>Client-Library and Gateway-Library function argument names</td>
<td>BUFFER, RETCODE</td>
</tr>
</tbody>
</table>
About This Book

This type of information | Looks like this
--- | ---
Client-Library function arguments that are input (I) or output (O) | COMMAND – (I) RETCODE – (O)
Names of objects stored on the mainframe | SYCTSAA5
Symbolic values used with function arguments, properties, and structure fields | CS-UNUSED, FMT-NAME, CS-SV-FATAL
Client-Library property names | CS-PASSWORD, CS-USERNAME
Client-Library and Gateway-Library datatypes | CS-CHAR, TDSCHAR

All other names and terms appear in this typeface.

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.
CHAPTER 1
Understanding the Client Option

What is the Client Option?

The Client Option is a programming environment that provides Open Client™ Client-Library routines you can use to build mainframe client applications.

The Client Option runs on an IBM z/Series or plug-compatible mainframe computer. It uses the LU 6.2 or TCP/IP communications protocols and is available for CICS.

Note The Client Option is also available for IMS TM and native MVS. This document pertains to the Client Option for CICS only.

Client Option applications can communicate with two kinds of servers:

- Adaptive Server® Enterprise (ASE) and Open Server™ on PCs and several mid-range UNIX platforms
- Server Option applications running in a CICS region on the mainframe

Client Option applications can send requests to ASE and Server Option applications.
Adaptive Server Enterprise (ASE)

Client Option applications can send requests to an ASE in a two-tier (gateway-less) environment using TCP.

Server Option

Client Option applications can send requests directly to the Server Option running in a CICS region. With TCP/IP, the Client Option can send requests to the Server Option running in the same CICS region.

Architecture

The Client Option works in a two-tier, gateway-less environment.

A Client Option network configuration consists of:

- A host-based client, which is a Client Option program running under CICS. The client program selects a server and sends requests to that server.
- A server, which can be any server that Client Option applications can access, including servers on the LAN—Sybase Open Servers and Adaptive Servers—as well as Server Option running in a separate CICS region. If using TCP/IP, the Client Option may access servers running in the same CICS region.

For more information on using Client-Library functions, see the appropriate Mainframe Connect Client Option Programmers Reference. COBOL, PL/1, and C versions of this reference guide are available.

Choosing a network driver

The Client Option supports concurrent use of multiple network drivers that can be invoked from the same Client Options and Server Options common code base. The appropriate network driver is loaded dynamically at the time the program executes.
You must use the SYGWDRIV macro to define the network drivers to be used with the Client Option and the Server Option. The default SYGWXCIPH member provided contains the SYGWDRIV macro definitions for all the supported network drivers. When installing the Client Option, be sure to edit the appropriate $HOST member to comment-out the drivers that your site does not intend to use.

This section provides an overview of network communication, and lists and describes general criteria for choosing a driver.

**General criteria for choosing a driver**

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

**CICS environment**

These drivers are supported in the CICS environment:

- TCP/IP for an IBM network
- LU 6.2 for an SNA network

The following table indicates which drivers can be used by the Client Option for CICS in a two-tier environment to communicate with the Server Option for CICS or for IMS and MVS.

<table>
<thead>
<tr>
<th>Driver</th>
<th>Gateway-less</th>
<th>To Server Option for CICS</th>
<th>To Server Option for IMS and MVS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LU62CICS</td>
<td>X1</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>TCP/CICS</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. The Client Option for CICS and the Server Option for CICS run in separate CICS regions.

**Client Option communications**

This section describes Client Option communications. It also explains the communication flow.
Client Option environment

The Client Option routes requests to servers on the LAN or on the mainframe. When using a TCP/IP network driver, Client Option requires the following additional information to establish a TCP/IP connection to a given server:

- The name of the IBM TCP/IP address space
- IP address or host name of the host on which the server runs
- Server name

When an application requests a connection to a server, Client Option:

- Looks up the server name in the TCP/IP Server-Host Mapping Table (SYGWHOST)
- Determines the type of listener the server is using. If the listener is for a Sybase LAN or mainframe, the Client Option routes the request directly to the server

After the server has received and processed the request, Client Option reads the results directly from a LAN-based server or mainframe Open Server.

Sample network

Figure 1-1 provides a sample two-tier TCP/IP environment. The Client Option is running in CICS region CICS1, and the Server Option is running in CICS2.

**Note** Using TCP/IP, the Client Option has the capability to connect to an Open Server running in the same CICS region.
In Figure 1-1, CICS1 has connections to:

- CICS2, the region where Server Option is installed
- A local area network (LAN) that has an ASE (myserve1) and an Open Server (myserve5)

Client Option programs can send requests to Server Option and to any LAN server. Figure 1-1 shows Client Option prog11a and prog11b sending requests to Server Option prog12a and prog12b.
Communication flow

This section describes what happens at the mainframe and at the server during Client Option processing.

At the mainframe

A Client Option application calls a pre-written procedure, such as a stored procedure or a Server Option application. All calls from the Client Option to remote nodes are processed using the TCP/IP communications protocol. For requests to an Open Server (LAN), the client can access any data available to the Open Server application. If the request is to the Server Option, the client can access any data storage system accessible through CICS.

The called procedure or transaction executes and returns results to the calling Client Option application, which can use the results for local processing. If the client has permission, the client transaction can update data at remote sites by inserting, modifying, and deleting entries in database tables or other data storage systems.

isql utility

The Client Option includes isql, a utility that allows users to send SQL language commands interactively. Users specify the server and whether to enable tracing, and then type SQL commands in a 3270 panel. For more information, see Chapter 4, “Using isql.”

Server-Host Mapping Table

For TCP/IP, the Client Option includes a Server-Host Mapping Table that allows you to define servers. For more information, see “Customizing the TCP/IP driver (SYGWHOST),” in Appendix A, “Customization Options.”

At the server

Typically, a server accepts requests from a client and returns results. In a Client Option environment, the server can be an ASE, an Open Server on the LAN, or a Server Option on the mainframe.

From the server standpoint, a request from an IBM host is no different than a request from a Sybase client. The Client Option participates in ASCII-EBCDIC and EBCDIC-ASCII translations and datatype conversions.
Understanding TCP/IP SYGWXCPH definitions

If you are using a TCP/IP driver, you must use the SYGWHOST macro to define the TCP/IP Server-Host Mapping Table.

For a description of the macro parameters and formats, see Appendix A, “Customization Options.”

Client Option security

Security for Client Option processing can be configured to require permission to:

- Log in to the target server or desired CICS region
- Use specific commands, stored procedures or transactions, and data objects at the target server

For more information about:

- **ASE security**: Refer to the chapter called “Security Administration,” in the Adaptive Server Enterprise System Administration Guide.
- **Mainframe security**: Refer to documentation provided with CICS and z/OS, or with the appropriate mainframe security system.
- **Secure Sockets Layer (SSL) implementation and configuration**: Refer to Appendix D in this guide.
CHAPTER 2
Planning Your Installation

Installation media

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

Note The Client Option is distributed on CD or in downloadable form. For information on obtaining the latest EBFs for the Client Option, see the Mainframe Connect Client Option for CICS Release Bulletin.

Pre-installation tasks

Installation requires completing the following pre-installation tasks, which are explained in the following subsections.

Note Skip any steps that do not pertain to the options you have chosen to install.

1. Verify the platforms, components, and distributed software
2. Determine JCL and system information
3. Determine CICS and DB2 UDB information
4. Determine compiler information
5. Determine Client Option information
Pre-installation tasks

6. Determine FTP information
7. Plan the security requirements
8. Identify the change control requirements
9. Back up the release libraries (upgrades only)
10. Determine the library names
11. Verify the connectivity

1. Verify the platforms, components, and distributed software

   See the Mainframe Connect Client Option for CICS Release Bulletin.

2. 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* – used for 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.

3. Determine CICS and DB2 UDB information

   Determine the following information if you intend to install a component that uses CICS or DB2 UDB:
   
   - *High-level qualifier for CICS system data sets*.
   - *RDO data set name (DSN)* – name of the CICS RDO (DFHCSD) containing the application resource definitions used by your CICS region.
4. Determine compiler information

Determine the following information if you intend to install an API component:

- *LE370 high-level qualifier* – used for the Language Environment 370.
- *COBOL compiler name* – module used to execute COBOL in your environment.
- *COBOL compiler LOADLIB* – system LOADLIB where your COBOL compiler module resides.
- *PL/1 compiler name* – module used to execute PL/1 in your environment.
- *PL/1 compiler LOADLIB* – system LOADLIB where your PL/1 compiler module resides.
- *C compiler data sets high-level qualifier* – high-level qualifier used for C.
- *TCP/IP data sets high-level qualifier*.

5. Determine Client Option information

Determine the following information for use in installing the Client Option:

- *TCP address space name*.
- *Remote server name* – name by which your Client Option applications will refer to the remote server.
- *Remote TCP host name* – DSN name for the remote server.
- *Remote server TCP host port* – TCP/IP port used by the remote server.
6. Determine FTP information

Determine the following information needed to establish an FTP connection to your mainframe:

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

7. Plan the security requirements

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

8. 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, such as field personnel and groups involved in administering applications, z/OS, security, change control, and scheduling
- A schedule, including cut-off dates for specific tasks

9. Back up the release libraries (upgrades only)

If you are upgrading an existing release of the Client Option, Sybase strongly recommends that you back up the entire set of release libraries before beginning this installation.
10. 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 are going to continue to use the old Sybase library names, delete all members before installing the new ones with the new version.

11. Verify the connectivity

Use the standard LAN ping utility to ensure connectivity between z/OS and the workstation running ASE.
Pre-installation tasks
CHAPTER 3

Installation and Configuration

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installation and configuration</td>
<td>15</td>
</tr>
<tr>
<td>Libraries and samples</td>
<td>21</td>
</tr>
</tbody>
</table>

Installation and configuration

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

Licensing information

The Client Option for CICS 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, be sure to 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 Client Option for CICS. You can 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.

2. The dialog box displays the options available for installation. To continue the installation, click Next.

3. 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.

4. Select the components you want to install.

**Note** If you are installing the Server Option for CICS or the DB2 UDB Option for CICS, the Server Option for CICS runtime component is automatically selected as you proceed to the next window.

Click Next.

5. 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** – indicates where generated data sets are cataloged.
   - **Unit** – (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. For CICS, this data set should be in the DFHRPL configuration ahead of other Sybase libraries.
Click Next.

6 If you are installing an option that uses CICS, DB2 UDB, or IMS, provide this information where it applies. Otherwise, skip to the next step.

• **CICS system datasets hlq** – high-level qualifier 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** – high-level qualifier 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** – high level qualifier used to locate IMS libraries.

Click Next.

7 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** – high-level qualifier used for the Language Environment 370 that 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 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** – high-level qualifier used for C to locate data sets like SBCCMP.
Installation and configuration

- **TCP/IP datasets hlq** – high-level qualifier used to locate data sets like `SEZATCP`.

  Click Next.

8 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.

9 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** – name of your TCP/IP region.
- **Listener Port** – 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.

  Click Next.

10 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.

11 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** – mainframe user ID for the FTP session.
  - **Password** – password for the FTP SESSION.
  - **Mainframe Host Name** – mainframe DSN name.
  - **FTP Port** – 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 hangs until the mainframe responds to a message requesting that the volume be mounted.

• **Log FTP Commands** – indicates where FTP log information is to be written. This log information may be useful in troubleshooting FTP PROBLEMS.

12 Click Next. The InstallShield wizard will create JCL and upload the selected components to your mainframe.

13 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 that 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

❖ **To complete the installation**

1 Locate the installation JCL for the Client Option for CICS in *hlq.OCC150.CICS.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.

  • *IxRDO* – runs the CICS Resource Definition Utility, *DFHCSDup*, to define the transaction, program, and file entries for the Client Option for CICS. If your CICS region has had a previous version of the Client Option, you may need to un-comment or change the DELETE and REMOVE entries at the top of the RDO input.
Installation and configuration

- **IxVSAM** – allocates the VSAM data sets used for error and trace logging. Because these data sets are shared by the Client Option for CICS, the Server Option for CICS, and the DB2 UDB Option for CICS, they may already have been created.

- **IxHOST** – assembles and links the Client Option for CICS customization module, character sets, remote host definitions, and license keys. You may rerun this job at any time to change configuration and character sets or to add, remove, or modify remote host definitions and license keys.

  **Note** The Server Option for CICS installation has its own IxHOST job. If you are installing both the Client and Server Options for CICS, you should run the IxHOST job contained in hlq.OSC150.CICS.JCL.

- **IxDELETE** – (optional) deletes the data sets in the TSO XMIT form used for the installation.

3 Run these jobs if you want to compile and link-edit the sample applications provided with the Client Option for CICS:

- **SCTC1** – compiles and links the sample C applications that use the Client Option interface.

- **SCTPL1** – compiles and links the sample PL/1 applications that use the Client Option interface.

- **SCTPL2** – compiles and links the sample PL/1 applications that act as Server Option for CICS applications that make Client Option calls.

  **Note** This job can be run only if the Server Option for CICS has been installed.

- **SCSACOB** – compiles and links the sample COBOL applications that use the CSA interface.

  **Note** This job can be run only if the Server Option for CICS has been installed.
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.
Libraries and samples
CHAPTER 4  Using isql

Introducing isql

Using the Interactive SQL (isql) utility, you can type requests directly onto a 3270-type window. The Client Option routes SQL requests to the requested server and displays the results in a window at your terminal. No programming is required.

From the mainframe, you can also use isql to access ASE, Open Server, or host-based databases through the Server Option. The server is specified in the login window.

This chapter describes the program windows and the entry choices available on them. Table 4-1 shows the keyboard commands for isql on these windows:
Invoking isql and initializing the environment

Table 4-1: isql function and action keys

<table>
<thead>
<tr>
<th>Key</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear</td>
<td>Ends the current transaction and exits from the current window to the CICS command window.</td>
</tr>
<tr>
<td>Enter</td>
<td>Performs the indicated action.</td>
</tr>
<tr>
<td>Tab</td>
<td>Moves the cursor to the next field.</td>
</tr>
<tr>
<td>PF1</td>
<td>Does one of the following:</td>
</tr>
<tr>
<td></td>
<td>• Resets the fields to blank.</td>
</tr>
<tr>
<td></td>
<td>• Resets the temporary storage browse pointer to the beginning of the queue.</td>
</tr>
<tr>
<td>PF3</td>
<td>Does one of the following:</td>
</tr>
<tr>
<td></td>
<td>• Terminates the connection to the server or, if not connected, terminates the environment.</td>
</tr>
<tr>
<td></td>
<td>• For viewing isql results, terminates the current result set. If you are viewing the last or only result set, it returns to the SQL input display.</td>
</tr>
<tr>
<td>PF4</td>
<td>Terminates all result sets and returns to the SQL input display.</td>
</tr>
<tr>
<td>PF7</td>
<td>Scrolls backward.</td>
</tr>
<tr>
<td>PF8</td>
<td>Scrolls forward.</td>
</tr>
<tr>
<td>PF9</td>
<td>Displays messages.</td>
</tr>
<tr>
<td>PF10</td>
<td>Shifts left.</td>
</tr>
<tr>
<td>PF11</td>
<td>Shifts right.</td>
</tr>
</tbody>
</table>

Invoking isql and initializing the environment

❖ To invoke isql and initialize the environment

1 Enter the isql command on a 3270 terminal window to invoke the Open Client Environment window.

When you open this window, it invokes isql.

2 To set tracing, enter:
   • Y – to enable tracing.
     During the current isql session, each time you press Enter after keying a value in a window or typing a SQL command, the Trace/Error window displays trace information results.
   • N (or blank) – to proceed without tracing.
When you press Enter after typing a value or SQL command, the SQL Result Display window displays results.

3 To set whether to save results for browsing, enter:
   • Y – to save results to temporary storage.
     This allows you to scroll backward as well as forward through returned rows. Otherwise, you can scroll forward only.
   • N (or blank) – to proceed without saving results to temporary storage.

Whether or not you save results, you can scroll left and right, if needed, to view the entire length of returned rows.

Refer to Table 4-1 for isql function and action keys.

Establishing the connection

Use the Establish Open Client Server Connection window to establish a connection with a specific server. The server can be an ASE, an Open Server application, or a separate CICS region on either the same or a different mainframe.

❖ To establish a connection to the server
1 At the initialization window, press Enter to display the Establish Open Client Server Connection window.

   Note At the bottom right of the window, the Temp Stg field indicates whether or not you chose to scroll backwards, that is, not to save results to temporary storage. (You set this parameter on the previous window.)

2 To connect to a specific server, enter the requested values:
   • Server name – name of the server you want to access. This name must already be defined in the Connection Router Table.
   • Login name – your login ID for the specified server. This login ID is passed to the server.
   • Password – password you use to log in to the specified server. This password is passed to the server.
   • Net-Driver – type of network driver you use (IBMTCPIP or LU62).
Entering SQL commands

- Login Charset – character set with which to log in to the remote server. Default value are:
  - If the USEIBMUNICODE parameter in SYGWMCST is set to Y, the default character set is utf8.
  - If the USEIBMUNICODE parameter in SYGWMCST is set to N, the default character set is iso_1.
- Display CCSID – CCSID to which query result sets should be converted.

When you press Enter, the isql utility establishes a connection with the named server and displays the Enter Open Client ISQL Commands window.

Entering SQL commands

❖ To enter SQL commands

1 Go to the Enter Open Client ISQL Commands window to submit a SQL command to the specified server. The accessed DBMS must recognize these SQL commands.

2 To send SQL commands to the server, enter a single SQL statement at the prompt, for example:

   select * from SYBASE.SAMPLETB

**Note** To get multiple results, first type “begin,” then type several lines without a joining semicolon, and then type “end,” for example:

   begin
   select firstname from sampletb
   select lastname from sampletb
   end

When you press Enter, the server processes the SQL command and returns the results. If you turned tracing on or specified that results be saved to temporary storage, the server saves the results to temporary storage.
Viewing data results

You can view the results of your SQL command on the SQL Result Display window as follows:

- If tracing is not turned on, the SQL Result Display window appears after you submit your SQL command.
- If tracing is turned on, the Trace/Error Display window displays the results of your SQL command. To display the SQL Result Display window, press Enter.

If result tables are larger than your screen, use the PF keys indicated on the display to scroll through the results.

**Note** You can scroll backward (PF7) only if when you initialized, you chose to save results to temporary storage.

Viewing trace results

The Trace/Error Display window displays after you finish typing a window. Error messages display at the top. For more information about the error messages, see the Mainframe Connect Client Option and Server Option *Messages and Codes* and the Mainframe Connect Client Option *Programmers Reference*. COBOL, C, and PL/1 versions of this guide are available.

Using the previous SQL command from step 2 in “Entering SQL commands” on page 26, the following results are shown on the Trace/Error Display window when underlying Client Option Client-Library routines are executed:

```sql
ct_con_alloc rc=-1
cd_diag (CS_INIT) rc=-1
cr_con_props (CS_USERNAME) rc=-1
cr_con_props (CS_PASSWORD) rc=-1
cr_connect rc=-1
```

For more information about individual Client-Library routines, refer to the Mainframe Connect Client Option *Programmers Reference*. COBOL, C, and PL/1 versions of this guide are available.
Entering SQL commands
Customization Options

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:

- SYGWMCST – a global customization macro.
- SYGWMCXL – 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 (SYGWMCST)

SYGWMCST, 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 SYGWMCST using the provided JCL member.

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

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

Note: The value utf8 is valid only if USEIBMUNICODE=Y.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Format</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DECPOINT</strong></td>
<td>:</td>
<td>Either a</td>
<td>Decimal point indicator, used only with the DB2 UDB option for CICS.</td>
</tr>
<tr>
<td><em>(Server Option only)</em></td>
<td></td>
<td>decimal</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>point or</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>comma</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>delimited</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>by single</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>quotation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>marks</td>
<td></td>
</tr>
<tr>
<td><strong>DEFLTPROTOCOL</strong></td>
<td>TCP</td>
<td>TCP</td>
<td>Specifies the default network driver protocol.</td>
</tr>
<tr>
<td><strong>DQUOTETRAN</strong></td>
<td>Y</td>
<td>Y or N</td>
<td>Used only with the DB2 UDB option for CICS. Make this setting</td>
</tr>
<tr>
<td><em>(Server Option only)</em></td>
<td></td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Note If you are using double-byte or multi-byte characters for DB2 metadata, set DQUOTETRAN=N.</td>
</tr>
<tr>
<td><strong>IMSLOGTYPE</strong></td>
<td>A1</td>
<td>A value</td>
<td>Specifies a log type. IMS TM reserves values less than A0.</td>
</tr>
<tr>
<td><em>(IMS TM only)</em></td>
<td></td>
<td>greater</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>than or</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>equal to A0</td>
<td></td>
</tr>
<tr>
<td><strong>LONGVARTRUNC</strong></td>
<td>N</td>
<td>Y or N</td>
<td>Indicates whether to truncate LongVarChar and VarBinary. For CICS only: 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 Users Guide for Transaction Router Services.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Customizing global options (SYGWMCST)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Format</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| MVSDNAME           | blank   | From 1 to 8 characters      | 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. MVSDNAME must match a DD name specified in the JCL for one of the following:  
  • 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      | Designates the default national language used by the Client Option or Server Option. Also see the CHARSETSRV property.                                                                                   |
| ROWLIMIT           | 0 (zero) | Used only by the DB2 UDB Option for CICS. When ROWLIMIT=0, there is no limit to the number of rows that can be sent. ROWLIMIT=n, where n indicates the global limit of rows that can be sent. |
| USEIBMUNICODE      | N       | Y or N                     | Specifies whether or not Unicode support for a particular z/OS installation is enabled through the IBM conversion environment and services.  
  • If USEIBMUNICODE=Y, IBM support is used for character set conversions.  
  • If USEIBMUNICODE=N, conversion is accomplished through the product-supplied translation tables. |

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

See “Using the IBM z/OS conversion environment and services” on page 33.
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. The CUNIMG01 member is loaded into z/OS using this command:

   SET UNI=01

   This command displays the current active image and the character set conversions defined for that image:

   DISPLAY UNI, ALL

To enable Unicode support for the Client Option and Server Option, set the USEIBMUNICODE configuration parameter to Y. The USEIBMUNICODE parameter is specified in the SYGWMCST 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.
Customizing mainframe character set conversion options (SYGWMCXL)

- 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:

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

**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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A2E</td>
<td>Optional ASCII-to-EBCDIC translate overrides</td>
</tr>
<tr>
<td>E2A</td>
<td>Optional EBCDIC-to-ASCII translate overrides</td>
</tr>
<tr>
<td>CHARSET</td>
<td>Name of the SBCS</td>
</tr>
<tr>
<td>CHARSETBYTES</td>
<td>S for SBCS</td>
</tr>
<tr>
<td>TYPE</td>
<td>Valid types:</td>
</tr>
<tr>
<td></td>
<td>• INITIAL</td>
</tr>
<tr>
<td></td>
<td>• ENTRY (default)</td>
</tr>
<tr>
<td></td>
<td>• FINAL</td>
</tr>
</tbody>
</table>

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 43.
Customizing mainframe character set conversion options (SYGWMCXL)

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.

The first example, Figure A-1, 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=iso_1,
    CHARSETBYTES=S,
    A2E=(0C-40,OA-40),
    E2A=(7F-20)
SYGWMCXL TYPE=FINAL
```

Start overrides in column 16.

Put continuation mark in column 72.

```
SYGWMCXL TYPE=INITIAL
SYGWMCXL TYPE=ENTRY
    CHARSET=iso_1,
    CHARSETBYTES=S,
    A2E=(0C-40,OA-40),
    E2A=(7F-20)
SYGWMCXL TYPE=FINAL
```

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>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARSET</td>
<td>The name of the SBCS or DBCS character set.</td>
</tr>
<tr>
<td>CHARSETBYTES</td>
<td>An S to denote SBCS, or a D to denote DBCS.</td>
</tr>
<tr>
<td>CCSID</td>
<td>The CCSID for the character set.</td>
</tr>
<tr>
<td>CHARTYPE</td>
<td>The type of character set. A denotes ASCII, and E denotes EBCDIC.</td>
</tr>
<tr>
<td>CHARSIZE</td>
<td>The maximum length of a character, from 1 to 4 bytes.</td>
</tr>
<tr>
<td>PAD</td>
<td>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.</td>
</tr>
</tbody>
</table>

**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**

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

**Example: code page 939**

```plaintext
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
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-3 shows the default drivers that are shipped with the Client Option or Server Option, depending on the environment:

Table A-4: CICS network drivers

<table>
<thead>
<tr>
<th>Driver</th>
<th>Load module name</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>LU 6.2</td>
<td>LU62CICS</td>
<td>Uses CICS LU 6.2 API</td>
</tr>
<tr>
<td>IBM TCP/IP</td>
<td>TCPCICS</td>
<td>Uses IBM EZACICAL API</td>
</tr>
<tr>
<td>CPIC</td>
<td>CPICCCICS</td>
<td>Uses CICS CPIC Support</td>
</tr>
</tbody>
</table>

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
```
Customizing dynamic network drivers (SYGWDRIV)

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.

For example:

**Figure A-2: CEDA window**

<table>
<thead>
<tr>
<th>OBJECT CHARACTERISTICS</th>
<th>CICS RELEASE = 0410</th>
</tr>
</thead>
<tbody>
<tr>
<td>CEDA View PARTner( MYSERVER )</td>
<td></td>
</tr>
<tr>
<td>PARTner : MYSERVER</td>
<td></td>
</tr>
<tr>
<td>Group : GROUP42</td>
<td></td>
</tr>
<tr>
<td>Description : SIDE INFO ENTRY TO GET TO mymcg</td>
<td></td>
</tr>
<tr>
<td>REMOTE LU NAME</td>
<td></td>
</tr>
<tr>
<td>NETName : U6T42P0M</td>
<td></td>
</tr>
<tr>
<td>NETWork :</td>
<td></td>
</tr>
<tr>
<td>SESSION PROPERTIES</td>
<td></td>
</tr>
<tr>
<td>Profile : SYOCPROF</td>
<td></td>
</tr>
<tr>
<td>REMOTE TP NAME</td>
<td></td>
</tr>
<tr>
<td>Tpname :</td>
<td></td>
</tr>
<tr>
<td>Xtpname : 94A8948387</td>
<td></td>
</tr>
<tr>
<td>SYSID=CICS APPLID=CICS41</td>
<td></td>
</tr>
</tbody>
</table>

Enter the PARTner and Remote TP name field values as follows:

- **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 SYGHOST 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 SYGHOST 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
   IBMTCPADRS/CNAME=&&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:
Defining license keys (SYGWLKEY)

The SYGWLKEY macro is part of the SYGWXCIPH global customization module. It is used to define the customer license key that is verified at runtime.

There are two parameters in this macro:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT</td>
<td>The product related to the license key, either the Client Option, the Server Option, or the DB2 UDB Option. Valid values are OCG, OSC, or DB2.</td>
</tr>
<tr>
<td>KEY</td>
<td>Defines the license key given for a product. The license key is a 23-character numeric value.</td>
</tr>
</tbody>
</table>
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 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 $xTCP 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 $xTCP 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
Building a global customization module (SYGWXCPH)
APPENDIX B

Translation Tables

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understanding the ASCII-EBCDIC and EBCDIC-ASCII</td>
<td>45</td>
</tr>
<tr>
<td>translation tables</td>
<td></td>
</tr>
<tr>
<td>Default ASCII_8 translation tables</td>
<td>47</td>
</tr>
<tr>
<td>Default ISO_1 translation tables</td>
<td>50</td>
</tr>
<tr>
<td>Default cp437 (code page 437) translation tables</td>
<td>53</td>
</tr>
<tr>
<td>Default cp850 (code page 850) translation tables</td>
<td>55</td>
</tr>
</tbody>
</table>

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 data conversion only if Unicode support is disabled and USEIBMUNICODE=N.

The four pairs of default tables are:

- ascii_8
- iso_1
- cp437
- cp 850

**Note**  The ascii_8 default table also provides the “base” for roman8 (HP), ibmascii, mac (Macintosh Roman), and user-definable character sets.
Understanding the ASCII-EBCDIC and EBCDIC-ASCII translation tables

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 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 “ASCII_8, ASCII-to-EBCDIC translation table” on page 48.

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

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

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 “ASCII_8, ASCII-to-EBCDIC translation table” on page 48.
Figure B-2: Example from the ASCII_8, EBCDIC-to-ASCII translation table

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

This section contains the following tables:

- ASCII_8, ASCII-to-EBCDIC translation table
- ASCII_8, EBCDIC-to-ASCII translation table

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.
Default ASCII_8 translation tables

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 | 90 | 91 | 92 | 93 | 94 | 95 |
| 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*
### APPENDIX B  Translation Tables

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>09</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>08</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>20</td>
<td>1C</td>
<td>20</td>
<td>0A</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>20</td>
<td>16</td>
<td>20</td>
<td>20</td>
<td>04</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>26</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>2D</td>
<td>2F</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>7</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>20</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
</tr>
<tr>
<td>9</td>
<td>20</td>
<td>6A</td>
<td>6B</td>
<td>6C</td>
<td>6D</td>
<td>6E</td>
</tr>
<tr>
<td>A</td>
<td>20</td>
<td>7E</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
</tr>
<tr>
<td>B</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>C</td>
<td>7B</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
</tr>
<tr>
<td>D</td>
<td>7D</td>
<td>4A</td>
<td>4B</td>
<td>4C</td>
<td>4D</td>
<td>4E</td>
</tr>
<tr>
<td>E</td>
<td>5C</td>
<td>20</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
</tr>
<tr>
<td>F</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
</tr>
</tbody>
</table>
Default ISO_1 translation tables

This section contains the following tables:

- ISO_1 ASCII-to-EBCDIC translation table
- ISO_1 EBCDIC-to-ASCII translation table

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_

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

## ISO_1 EBCDIC-to-ASCII translation table

_Figure B-6: ISO_1 EBCDIC-to-ASCII translation table_
Default ISO_1 translation tables

<table>
<thead>
<tr>
<th></th>
<th><em>0</em></th>
<th><em>1</em></th>
<th><em>2</em></th>
<th><em>3</em></th>
<th><em>4</em></th>
<th><em>5</em></th>
<th><em>6</em></th>
<th><em>7</em></th>
<th><em>8</em></th>
<th><em>9</em></th>
<th><em>A</em></th>
<th><em>B</em></th>
<th><em>C</em></th>
<th><em>D</em></th>
<th><em>E</em></th>
<th><em>F</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>0_</td>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>09</td>
<td>08</td>
<td>7F</td>
<td>97</td>
<td>8D</td>
<td>8E</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
<td></td>
</tr>
<tr>
<td>1_</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>9D</td>
<td>85</td>
<td>08</td>
<td>87</td>
<td>18</td>
<td>19</td>
<td>92</td>
<td>8F</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
<td>1F</td>
</tr>
<tr>
<td>2_</td>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>0A</td>
<td>17</td>
<td>1B</td>
<td>88</td>
<td>89</td>
<td>8A</td>
<td>8B</td>
<td>8C</td>
<td>05</td>
<td>06</td>
<td>07</td>
</tr>
<tr>
<td>3_</td>
<td>90</td>
<td>91</td>
<td>92</td>
<td>93</td>
<td>94</td>
<td>95</td>
<td>96</td>
<td>98</td>
<td>99</td>
<td>9A</td>
<td>9B</td>
<td>14</td>
<td>15</td>
<td>9E</td>
<td>1A</td>
<td></td>
</tr>
<tr>
<td>4_</td>
<td>20</td>
<td>A0</td>
<td>E2</td>
<td>E4</td>
<td>E0</td>
<td>E1</td>
<td>E3</td>
<td>E5</td>
<td>E7</td>
<td>F1</td>
<td>5B</td>
<td>2E</td>
<td>3C</td>
<td>28</td>
<td>2B</td>
<td>21</td>
</tr>
<tr>
<td>5_</td>
<td>26</td>
<td>E9</td>
<td>EA</td>
<td>EB</td>
<td>E8</td>
<td>ED</td>
<td>EE</td>
<td>EF</td>
<td>EC</td>
<td>DF</td>
<td>5D</td>
<td>24</td>
<td>2A</td>
<td>29</td>
<td>3B</td>
<td>5E</td>
</tr>
<tr>
<td>6_</td>
<td>2D</td>
<td>2F</td>
<td>C2</td>
<td>C4</td>
<td>C0</td>
<td>C1</td>
<td>C3</td>
<td>C5</td>
<td>C7</td>
<td>D1</td>
<td>A6</td>
<td>2C</td>
<td>25</td>
<td>5F</td>
<td>3E</td>
<td>3F</td>
</tr>
<tr>
<td>7_</td>
<td>F8</td>
<td>C9</td>
<td>CA</td>
<td>CB</td>
<td>C8</td>
<td>CD</td>
<td>CE</td>
<td>CF</td>
<td>CC</td>
<td>60</td>
<td>3A</td>
<td>23</td>
<td>40</td>
<td>27</td>
<td>3D</td>
<td>22</td>
</tr>
<tr>
<td>8_</td>
<td>D8</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>AB</td>
<td>BB</td>
<td>F0</td>
<td>FD</td>
<td>FE</td>
<td>B1</td>
</tr>
<tr>
<td>9_</td>
<td>B0</td>
<td>6A</td>
<td>6B</td>
<td>6C</td>
<td>6D</td>
<td>6E</td>
<td>6F</td>
<td>70</td>
<td>71</td>
<td>72</td>
<td>AA</td>
<td>BA</td>
<td>E6</td>
<td>B8</td>
<td>C6</td>
<td>A4</td>
</tr>
<tr>
<td>A_</td>
<td>B5</td>
<td>7E</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td>78</td>
<td>79</td>
<td>7A</td>
<td>A1</td>
<td>BF</td>
<td>D0</td>
<td>DD</td>
<td>DE</td>
<td>AE</td>
</tr>
<tr>
<td>B_</td>
<td>A2</td>
<td>A3</td>
<td>A5</td>
<td>B7</td>
<td>A9</td>
<td>A7</td>
<td>B6</td>
<td>BC</td>
<td>BD</td>
<td>BE</td>
<td>AC</td>
<td>7C</td>
<td>AF</td>
<td>A8</td>
<td>B4</td>
<td>D7</td>
</tr>
<tr>
<td>C_</td>
<td>7B</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>AD</td>
<td>F4</td>
<td>F6</td>
<td>F2</td>
<td>F3</td>
<td>F5</td>
</tr>
<tr>
<td>D_</td>
<td>7D</td>
<td>4A</td>
<td>4B</td>
<td>4C</td>
<td>4D</td>
<td>4E</td>
<td>4F</td>
<td>50</td>
<td>51</td>
<td>52</td>
<td>B9</td>
<td>FB</td>
<td>FC</td>
<td>F9</td>
<td>FA</td>
<td>FF</td>
</tr>
<tr>
<td>E_</td>
<td>5C</td>
<td>P7</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
<td>5A</td>
<td>B2</td>
<td>D4</td>
<td>D6</td>
<td>D2</td>
<td>D3</td>
<td>D5</td>
</tr>
<tr>
<td>F_</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>B3</td>
<td>DB</td>
<td>DC</td>
<td>D9</td>
<td>DA</td>
<td>9F</td>
</tr>
</tbody>
</table>
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**

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


**APPENDIX B   Translation Tables**

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

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

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

**Default cp850 (code page 850) translation tables**

This section contains the following tables:
Default cp850 (code page 850) translation tables

- cp850 ASCII-to-EBCDIC translation table
- cp850 EBCDIC-to-ASCII translation table

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**

<table>
<thead>
<tr>
<th><em>0</em></th>
<th><em>1</em></th>
<th><em>2</em></th>
<th><em>3</em></th>
<th><em>4</em></th>
<th><em>5</em></th>
<th><em>6</em></th>
<th><em>7</em></th>
<th><em>8</em></th>
<th><em>9</em></th>
<th><em>A</em></th>
<th><em>B</em></th>
<th><em>C</em></th>
<th><em>D</em></th>
<th><em>E</em></th>
<th><em>F</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>0_</td>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
</tr>
<tr>
<td>1_</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
</tr>
<tr>
<td>2_</td>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>2A</td>
<td>2B</td>
<td>2C</td>
<td>2D</td>
<td>2E</td>
</tr>
<tr>
<td>3_</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>3A</td>
<td>3B</td>
<td>3C</td>
<td>3D</td>
<td>3E</td>
</tr>
<tr>
<td>4_</td>
<td>40</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>4A</td>
<td>4B</td>
<td>4C</td>
<td>4D</td>
<td>4E</td>
</tr>
<tr>
<td>5_</td>
<td>50</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
<td>5A</td>
<td>5B</td>
<td>5C</td>
<td>5D</td>
<td>5E</td>
</tr>
<tr>
<td>6_</td>
<td>60</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>6A</td>
<td>6B</td>
<td>6C</td>
<td>6D</td>
<td>6E</td>
</tr>
<tr>
<td>7_</td>
<td>70</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td>78</td>
<td>79</td>
<td>7A</td>
<td>7B</td>
<td>7C</td>
<td>7D</td>
<td>7E</td>
</tr>
<tr>
<td>8_</td>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td>86</td>
<td>87</td>
<td>88</td>
<td>89</td>
<td>8A</td>
<td>8B</td>
<td>8C</td>
<td>8D</td>
<td>8E</td>
</tr>
<tr>
<td>9_</td>
<td>90</td>
<td>91</td>
<td>92</td>
<td>93</td>
<td>94</td>
<td>95</td>
<td>96</td>
<td>97</td>
<td>98</td>
<td>99</td>
<td>9A</td>
<td>9B</td>
<td>9C</td>
<td>9D</td>
<td>9E</td>
</tr>
<tr>
<td>A_</td>
<td>A0</td>
<td>A1</td>
<td>A2</td>
<td>A3</td>
<td>A4</td>
<td>A5</td>
<td>A6</td>
<td>A7</td>
<td>A8</td>
<td>A9</td>
<td>AA</td>
<td>AB</td>
<td>AC</td>
<td>AD</td>
<td>AE</td>
</tr>
<tr>
<td>B_</td>
<td>B0</td>
<td>B1</td>
<td>B2</td>
<td>B3</td>
<td>B4</td>
<td>B5</td>
<td>B6</td>
<td>B7</td>
<td>B8</td>
<td>B9</td>
<td>BA</td>
<td>BB</td>
<td>BC</td>
<td>BD</td>
<td>BE</td>
</tr>
<tr>
<td>C_</td>
<td>C0</td>
<td>C1</td>
<td>C2</td>
<td>C3</td>
<td>C4</td>
<td>C5</td>
<td>C6</td>
<td>C7</td>
<td>C8</td>
<td>C9</td>
<td>DA</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>DE</td>
</tr>
<tr>
<td>D_</td>
<td>D0</td>
<td>D1</td>
<td>D2</td>
<td>D3</td>
<td>D4</td>
<td>D5</td>
<td>D6</td>
<td>D7</td>
<td>D8</td>
<td>D9</td>
<td>EF</td>
<td>FD</td>
<td>8C</td>
<td>8D</td>
<td>AD</td>
</tr>
<tr>
<td>E_</td>
<td>EE</td>
<td>EF</td>
<td>FG</td>
<td>8E</td>
<td>8F</td>
<td>90</td>
<td>91</td>
<td>92</td>
<td>93</td>
<td>94</td>
<td>9A</td>
<td>9B</td>
<td>9C</td>
<td>9D</td>
<td>9E</td>
</tr>
<tr>
<td>F_</td>
<td>CA</td>
<td>8F</td>
<td>1B</td>
<td>B9</td>
<td>B6</td>
<td>B5</td>
<td>E1</td>
<td>9D</td>
<td>90</td>
<td>BD</td>
<td>B3</td>
<td>DA</td>
<td>FA</td>
<td>EA</td>
<td>3E</td>
</tr>
</tbody>
</table>
### cp850 EBCDIC-to-ASCII translation table

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

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

Understanding the CICS sockets interface

Mainframe Connect Client Option for CICS contains modules for implementing IBM CICS sockets interface. Before running Mainframe Connect Client Option for CICS, make sure that a CICS region has the IBM CICS sockets interface installed and configured. For details, refer to z/OS Communications Server: IP CICS Sockets Guide.

For Mainframe Connect Client Option for CICS, using the IBM CICS sockets interface requires:

- Installing IBM and Sybase RDO definitions for the required programs and transactions
- Concatenating the sockets load library
- Enabling CICS sockets interface automatically or by using the EZAO command

**Note** To run the Client Option for CICS, you do not need the SYBTPSEC configuration module.
Installing and configuring the Client Option of CICS sockets interface

Follow these procedures to install and configure the CICS sockets interface for Mainframe Connect Client Option for CICS.

❖ To install and configure the CICS sockets interface

1. Add the IBM CICS sockets RDO entries to your CICS region. You can find input for DFHCSDUP in tcpqlq.SEZAINST(EZACICCT).

   **Note** tcpqlq refers to the high-level qualifier of your system’s IBM TCP/IP configuration.

2. Define the EZACONFG data set to contain the configuration data for CICS sockets and load the system configuration parameters.

   You can find IBM-supplied JCL for this procedural step in tcpqlq.SEZAINST(EZACICFG).

3. Add tcpqlq.SEZATCP into the DFHRPL concatenation for the CICS region.

   To control the CICS sockets interface, you can use the EZAC and the EZAO commands. To start or stop the entire sockets interface, use EZAO.

   **Note** If you want the CICS sockets interface to start automatically when CICS is initialized, you must add the EZACIC20 program to the second stage of the start-up PLT and to the first stage of the shutdown PLT. Any sockets defined with IMMEDIATE=YES are automatically started.
Setting Up Secure Sockets Layer Protocol

Understanding Secure Sockets Layer

Mainframe Connect Client Option and Server Option for CICS supports Secure Sockets Layer (SSL) session-based security. SSL is the standard for securing the transmission of sensitive information (such as credit card numbers, stock trades, and banking transactions) over the Internet. These sections describe the Secure Sockets Layer (SSL) protocol and how it works.

Description of SSL features

SSL provides these features:

- Authentication for clients and servers, with practical emphasis on the server
- Data confidentiality (encryption)
- Verification that a transaction was sent by the client and that the identical transaction was received by the server

To provide efficient authentication and encryption, SSL combines private-key and public-key technologies.
Public-key (asymmetric) cryptography is based on the Public Key Infrastructure (PKI) method of encryption, in which two different keys are used for encrypting and decrypting operations: one is public, the other is private. This means that an operation encrypted by one key can only be decrypted by the other key. The result is that public keys can be seen by all, yet privacy is still possible.

The highly-used RSA algorithm works for both encryption and decryption operations. This method solves critical key exchange issue, but the algorithms it uses require large key sizes and often result in slow CPU-bound operations.

In private-key (symmetric) cryptography, the sender and receiver use the same key for both encryption and decryption operations on the same data. Key size is very important: the longer the key, the stronger it is. Currently, 1024, 2048 is the recommended length. The private-key (symmetric) method is 1000 times faster than the public-key (asymmetric) method.

How SSL provides security

This section describes how SSL provides and signs certificates to provide security.

SSL starts with a “handshake,” during which the client authenticates the server, and the server optionally authenticates the client. Handshake negotiations are based on the public-key cryptography: The client and server agree on how to encrypt and decrypt data, such as using cipher suites and session keys. Also, the format to transmit encrypted data is defined in the handshake.

When securing communications, both client and server use X.509 certificates. The client must verify the server’s certificate based on the certificate of the Certificate Authority (CA) that signed the certificate or based on a self-signed certificate from the server. (The client verification is optional.) Then, the client and the server use the negotiated session keys and begin encrypted communication, using private-key cryptography.

These are the main fields in a X.509 V3 certificate:

- Issuer
- Subject fields (an X.500 Distinguished Name, commonName field)
- Algorithm identifier
- Subject’s public key
- Period of validity
Generating a certificate

The process of generating a certificate involves these basic steps:
1. Generate the public-private key pair.
2. Store the private key securely (as password-encrypted).
3. Generate a certificate signing request (CSR) in PKCS#10 format for the server certificate.
4. Present the CSR to the CA.
5. Receive the signed certificate from the CA.
6. Store the certificate.

You can use different tools to generate test (self-signed) certificates.

Setting up SSL in Sybase products

As of version 12.5, Sybase added SSL support to some Sybase products, such as ASE, Open Client, Open Server, Replication Server, and jConnect. Currently, some Sybase products use Certicom SSLPlus and Security Builder libraries.

Note
Be aware that ASE 12.5 uses SSLPlus, versions 3.1.4/5 and 5.1.4, but ASE 15.0 uses SSLPlus, version 5.1.4.

ASE can use SSL on CIS and Site-handler. Also, ASE and Replication Server require licensing to use SSL.

Some Sybase products have SSL configuration options and can be easily configured to use SSL. Others provide special APIs or require some programming. The following section provides instructions for specific setups.

Setting up SSL in ASE and Open Client

Follow these procedures to configure and enable ASE and Open Client with SSL.
To set up SSL in ASE

1. Obtain an ASE license for the SSL feature, either ASE_ASM or ASE_SECDIRS (SySAM).
2. Obtain a certificate for the server (plus any CA).
   Use the Sybase-provided utilities certreq and certauth to respectively generate and sign server certificates or CA (self-signed) certificates.
3. Install the server certificate with the private key appended. The default location and naming scheme for the server certificate is $SYBASE/$SYBASE_ASE/certificates/<server>.crt.
   **Note** The <server> name must match the server name from the interfaces file, as well as the CommonName from the server certificate.
4. Install CA certificates for the server at this location: $SYBASE/$SYBASE_ASE/certificates/<server>.txt.
5. Use the sp_ssladmin addcrt command to let the server know about the certificate location.
6. To enable SSL in the server, issue this statement:
   ```
   sp_config "enable ssl", 1
   ```
7. Modify directory services so that listening ports use SSL by adding the keyword ssl to the appropriate interfaces file entries.

To set up SSL in Open Client

1. Install client copies of CA certificates. You must concatenate your <server>.txt file to the trusted.text file, or simply create a new trusted.text file with the <server>.txt contents.
   Client default locations are:
   - For UNIX: $SYBASE/config/trusted.txt
   - For Windows: %sybase%\ini\trusted.txt
2. Enable SSL on the port in Directory Servers, for example, the interfaces file.
Setting up SSL in IBM z/OS

Note The term SSL is used to describe both SSL and TLS protocols.

SSL protocol runs above the TCP/IP protocol and below higher-level protocols such as HTTP.

IBM SSL support runs as part of the TCP/IP stack under UNIX System Services (USS).

These levels of authentication and encryption are available with TLS/SSL security:

- Server authentication only
- Client authentication level 1
- Client authentication level 2
- Client authentication level 3

Note For consistency with other Sybase products, Sybase implements server authentication only.

For server authentication to work, the server must have a private key and associated server certificate in the server key database file. To manage the keys and certificates needed for SSL support, you can use the gskkyman utility (provided by the System SSL) or RACF Common Keyring support. The server certificate and the CA certificates are stored in a key ring, also called a key database.

Some considerations when using RACF:

- References to RACF apply to any other System Authorization Facility (SAF)-compliant security products that contain the required support.
- For RACF support, all key rings and certificates are stored in the RACF database. There are no separate key database or stash files.

IBM provides the mainframe applications with two options for implementing SSL support:

- System SSL, which runs on top of the TCP/IP stack and provides interfaces to write both client and server applications.
Application Transparent - Transport Layer Security (AT-TLS), which provides application-transparent secured connections for both client and server. Internally, it uses System SSL interfaces.

The following subsections describe each of these options.

### Using System SSL on z/OS

System SSL provides APIs associated with either SSL environment layer or a secure socket connection layer.

- The SSL environment layer defines the general attributes of the environment (such as database file name, timeout, and so on).
- Secure socket connection layer defines the attributes associated with each secure connection. In addition, secure socket connection layers has read and write function calls.

First, the SSL application must create the SSL environment layer. Then, one or more secure socket connection layers can be associated with the SSL environment.

Each layer has four general function calls:

- **open**
- **attribute_set**
- **initialize**
- **close**

The `open` function calls return a handle, either an environment handle or a secure socket connection handle, that must be passed as a parameter on subsequent function calls. `Read` and `write` functions are full-duplex; however, only one `read` and one `write` call can be in progress at one time for any secure socket connection handle.

In addition to using the SSL programming interfaces in the application, a key database must be created for the SSL application. This key database, which contains certificate information, can be an HFS file built and managed by the `gskkyman` utility or a RACF key ring.

System SSL uses the Integrated Cryptographic Service Facility (ICSF), if it is available. ICSF provides hardware cryptographic support that will be used instead of the System SSL software algorithms.
For System SSL to use the hardware support, the ICSF-started task must be running, and the application user ID must be authorized to the key and certificate handling resources in the RACF CSFSERV class. RACF can also be used to control access to ICSF services.

**Note** SSL applications must call SSL APIs from a C program because they are C APIs.

### Configuring AT-TLS

AT-TLS consolidates TLS implementation in one location, reducing or eliminating application development overhead, maintenance, and parameter specification. AT-TLS is based on z/OS System SSL, and transparently implements it in the TCP layer of the stack.

Applications that are taking advantage of the AT-TLS can be separated into three different types: basic, aware, and controlling. The type is based on whether the application is aware of the service, and if so, the amount of control that the application is given over the security functions. The `SIOCTTLSCTL.ioctl` function call provides the interface for the application to query or control AT-TLS.

- **Basic application**
  - A basic application is unaware that AT-TLS is encrypting or decrypting data.

- **Aware application**
  - An aware application is aware of AT-TLS and can query information such as AT-TLS status, partner certificate, and derived RACF user ID without any advanced setting in AT-TLS policy.

- **Controlling application**
  - A controlling application is aware of AT-TLS and needs to control the secure session. It must have the `ApplicationControlled` parameter in the AT-TLS policy set to ON.

  All of these types of applications send and receive unencrypted text data while encrypted data flows over the network.

  Follow this procedure to configure AT-TLS policies.

- **To configure AT-TLS**
  1. Provide the TCP/IP stack with the AT-TLS policies required to negotiate secure connections.
AT-TLS policies are configured in the Policy Agent (described in the next section) using a set of configuration statements and parameters coded into a flat file. You can create the flat file using one of two methods:

- Using manual configuration, coding all the required statements in an HFS file or MVS data set, or
- Using z/OS Network Security Configuration Assistant, which is a standalone Windows application that requires no network connectivity or setup. You can download the GUI from the Web site for the Communication Server Family downloadable tool.

2 Enable AT-TLS through the TTLS parameter on the TCPCONFIG statement in PROFILE.TCPIP.

When AT-TLS is enabled and a newly established connection is first used, the TCP layer of the stack searches for a matching AT-TLS policy installed from the Policy Agent. If no policy is found, the connection is made without AT-TLS involvement.

### Configuring AT-TLS policies in the Policy Agent

The Policy Agent component is responsible for implementing policy decisions that control network security and traffic prioritization for the z/OS environment. When initiated, the Policy Agent reads the configuration files, parses the policies, and stores the policy definitions in the TCP/IP stack, which then operates based on the policies. When the policy rule is true, one set of actions is initiated; when it is false, a different set of actions is initiated.

The Policy Agent main configuration file points to other policy files that contain specific policies for TCP/IP images. It can contain a TcpImage statement that identifies the z/OS UNIX file or MVS data set that contains the policy to be received by a stack. On its end, the TCP/IP image policy file can contain a TTLSConfig statement that identifies the z/OS UNIX file or MVS data set that contains the AT-TLS policy.

### Types of configuration files

There are several types of configuration files:

- Main configuration file, determined by using a standard search order
- Common IPSec configuration files
- Common AT-TLS configuration files
- Image configuration files
- Image AT-TLS configuration files
Within the AT-TLS policy file, a TTLSRule statement defines a set of conditions that are compared against the connection being checked. When a match is found, policy look-up stops, and the connection is assigned the actions associated with the rule.

The rule conditions apply to connect parameters as follows:

- LocalAddr
- RemoteAddr
- LocalPortrange
- RemotePortrange
- Jobname
- Userid

Direction and at least one other condition must be specified. The TTLSRule statement can reference up to three action statements. In a simple implementation for AT-TLS, these configuration statements should be defined:

- TTLSGroupAction, which must specify TTLSEnabled=ON. The AT-TLS group action represents a single Language Environment process and enclave, and initializes one instance of the System SSL DLL.
- TTLSEnvironmentAction, which must specify a key ring and the handshake role. The AT-TLS environment action initializes a System SSL environment within the Language Environment process that was created to represent an AT-TLS group action.
- TTLSConnectionAction, which specifies attributes for a subset of connections. It is not required for a simple implementation.

❖ To start the Policy Agent

- You can start the Policy Agent, which runs as a UNIX process, using one of two methods:
  - From the z/OS shell, where its executable resides in /usr/lpp/tcpip/sbin, or
  - As a started task using the PAGENT command on an MVS console. You can find a sample started task procedure for PAGENT in TCPIP.SEZAINST(EZAPAGSP).

Note To start Policy Agent from z/OS, you need security product authorization definition (for RACF or any other product).
The Policy Agent search order for accessing the main configuration file (PAGENT.CONF information) is:

1. File or data set specified with the -c startup option
2. File or data set specified with the PAGENT_CONFIG_FILE environment variable
3. The etc/pagent.conf file

These environment variables are used to tailor the Policy Agent to a particular installation:

- PAGENT_CONFIG_FILE, which points to the main configuration file or data set
- PAGENT_LOG_FILE, which points to the log file
- PAGENT_LOG_FILE_CONTROL, which controls the number and size of log files

You might also need to define these:

- TZ, which defines the local time zone, even if it is defined in /etc/profile
- LIBPATH, which points to the dynamic link libraries (DLLs) needed to act as an LDAP client

AT-TLS support in Client and Server Options for CICS

In the CICS sockets implementation, transaction security environments are not visible to AT-TLS support. The CICS job and all its transactions appear to the stack as a single server application. As a result, all AT-TLS policy look-up, System SSL key ring authorization checks, and ICSF private key authorization checks are processed using the identity of the CICS job.

The connection that is established, whether active or passive, can perform SSL handshake processing as either the client or the server. All of the connections established by a single CICS job can share the session ID cache in the SSL environment. The CICS job should use a private key ring with a Server certificate, and the key ring used must contain the chain of the root certificates it needs to validate the Server certificate it presents to the client.

Mainframe Connect Client and Server Options for CICS take advantage of the AT-TLS security support, provided that the following conditions are true:
The TCP/IP stack supports AT-TLS.

An AT-TLS Policy configuration matches identifiers of the CICS application that will use it, for example, the status of the application as a listener or a client, the IP addresses, and the ports that will be used for communication.

Digital certificates and key rings are created for these applications.

Configuring a z/OS client or server system

On each z/OS system where a server or client application is to implement AT-TLS security, you need to perform these basic tasks:

1. Create a key ring
2. Create Policy Agent files
3. Add AT-TLS configuration
4. Add statements to the TTLSCFG policy file
5. Set up InitStack access control
6. Enable AT-TLS.

The following subsections present an example of configuration tasks performed to ensure SSL secure communication for the following network participants, all of which use self-signed digital certificates:

- A z/OS CICS server named “CICSDEV1”
- A z/OS CICS client
- A Windows ASE server named “ase1”
- A Windows Open Client client

1. Create a key ring

The task of creating a key ring involves all steps for generating and managing digital certificates. To do so, you use RACF commands. Be sure that you have RACF authority.

Note: You need the SPECIAL attribute to issue the RACDCERT command. (GROUP-SPECIAL is not sufficient.)
To create a key ring

1. To activate certificate and key ring classes, use these commands:
   - `SETROPTS CLASSACT(DIGTRING)
   - `SETROPTS CLASSACT(DIGTCERT)

2. To refresh after you make changes, use these commands:
   - `SETROPTS RACLIST(DIGTRING) REFRESH
   - `SETROPTS RACLIST(DIGTCERT) REFRESH

3. To give access to the appropriate resources, use these commands:
   - `RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
   - `RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
   - `PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY)ID(KGUEOR) ACCESS(READ)
   - `PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY)ID(KGUEOR) ACCESS(READ)
   - `PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY)ID(DFHCICS) ACCESS(CONTROL)

4. To define a key ring, generate a self-signed certificate, and connect it to the key ring, use these commands:
   - `RACDCERT ID(DFHCICS) ADDRING(CICSDEV1)
   - `RACDCERT ID(DFHCICS) GENCERT SUBJECTSDN(CN('CICSDEV1') OU('MFI') O('MFI') C('US')) WITHLABEL ('CICSDEV1CERT') TRUST SIZE(1024)
   - `RACDCERT ID(DFHCICS) CONNECT (ID(DFHCICS) RING(CICSDEV1) LABEL('CICSDEV1CERT') DEFAULT)
   - `SETROPTS RACLIST (DIGTRING DIGTCERT) REFRESH

5. To ensure the Windows SSL client connection to the “CICSDEV1” server, export the newly-created, self-signed certificate to a data set using this command:
   - `RACDCERT EXPORT(LABEL('CICSDEV1CERT')) ID(DFHCICS) DSN(CICSDEV1.CERT)

This puts the certificate contents into the PS data set called KGUEOR.CICSDEV1.CERT using the default FORMAT(CERTB64) and code page 1047.
6 FTP the contents of that file to the Windows client's host with ASCII conversion.

7 Paste the certificate to the client's CA list in the \%sybase\%\in\trusted.txt file.

8 Add ".,ssl" to the CICSDEV1 server entry in the client's sql.ini file.

9 To ensure that the CICS client (Client Option for CICS) connects to the SSL server named "ase1," FTP the "ase1" certificate file from the $SYBASE/$SYBASE_ASE/certificates/ase1.txt file to the KGUEOR.ASE1CERT data set, with ASCII conversion.

10 Add the sent FTP "ase1" CA certificate to the RACF database with a TRUST status using this command:

   RACDCERT ID(DFHCICS) ADD('KGUEOR.ASE1CERT')
   WITHLABEL('ASE1CERT') TRUST

11 Connect the newly added certificate to the key ring and refresh:

   RACDCERT ID(DFHCICS) CONNECT (ID(DFHCICS) RING(CICSDEV1)
   LABEL('ASE1CERT'))
   SETROPTS RACLIST (DIGTRING DIGTCERT) REFRESH

**Note** Be aware of certificates encoding, which is generally in binary or text formats. Binary formats must be transported in their exact binary format, without any conversion. In contrast, text formats (such as Base64) must be transported as text. When transporting for an ASCII system, be sure that the ASCII-to-EBCDIC translation is performed.

---

**2. Create Policy Agent files**

**Note** For all Policy Agent configuration file examples, only the content relevant to the AT-TLS configuring is presented here.

The main TCP/IP configuration file is in /etc/pagent.conf3. It contains two TcpImage statements for defining policies on stacks TCPIP and TCPIPT:

```
# Path: /etc/pagent.conf3

# This is a sample PAGENT.CONF Policy Agent main configuration file
# applied to stacks TCPIP and TCPIPT
LogLevel 15
```

---

**Installation and Administration Guide**
### AT-TLS support in Client and Server Options for CICS

# AT-TLS config needed for TCPIPT stack
TcpImage TCPIPT /etc/TCPIPT.policy # FLUSH PURGE 600 params can be used
TcpImage TCPIP /etc/TCPIP.policy # No AT-TLS policy applied to this stack

#### 3. Add the AT-TLS configuration

The policy file for the TCPIPT stack, /etc/TCPIPT.policy, contains a TTLSConfig statement pointing to the AT-TLS configuration file for that stack:

```
#   This is a sample TCP/IP image configuration file
# Path: /etc/TCPIPT.policy
#   This is a sample TCP/IP image configuration file
#   applied to stack TCPIPT
LogLevel 15
TTLSConfig /etc/pagent_TTLS.conf3
```

#### 4. Add statements to the TTLSConfig policy file

The TCPIPT stack configuration file /etc/pagent_TTLS.conf3 has all the AT-TLS configuration statements for that stack:

```
#   Path: /etc/pagent_TTLS.conf3
#   This is a sample AT-TLS configuration file for
# stack TCPIPT, server CICSDEV1.
# Server port #3042 is defined as SSL and port #3042 - as non-SSL
# Client is to connect to a remote SSL port #6005.
#### Inbound definitions ####

TTLSRule                  DEV1SY02    # Listener on port #3042
{ LocalPortRange           3042
  Userid                   DFHCICS
  Direction                Inbound
  TTLSGroupActionRef       TTLSGRP1
  TTLSEnvironmentActionRef TTLSENV1 }

TTLSGroupAction TTLSGRP1
{
  TLSEnabled On
  Trace 1             # Log Errors to TCP/IP job log
}

TTLSEnvironmentAction TTLSENV1
{
  HandshakeRole       Server
```
EnvironmentUserInstance 1
TTLSSKeyRingParms
{
Keyring CICSDEV1 # Refers to RACF database
}

# Port #3043 is not AT-TLS configured
TTLSSRule DEV1SY03 # Listener on port #3043
{
LocalPortRange 3043
Userid DFHCICS
Direction Inbound
TTLSSGroupActionRef TTLSGRP2
}

TTLSSGroupAction TTLSGRP2
{
TTLSEnabled Off
Trace 1 # Log Errors to TCP/IP job log
}

TTLSEnvironmentAction TTLSENV2
{
HandshakeRole Server
EnvironmentUserInstance 3
TTLSSKeyRingParms
{
Keyring CICSDEV1 # Refers to RACF database
}
}

### Outbound definitions ###
#
TTLSSRule DEV1Client # Client
{
RemotePortRange 6005 # Remote port to connect to
Userid DFHCICS
Direction Outbound
TTLSSGroupActionRef TTLSGRP3
TTLSEnvironmentActionRef TTLSENV3
}

TTLSSGroupAction TTLSGRP3
{
TTLSEnabled On
Trace 1 # Log Errors to TCP/IP job log

AT-TLS support in Client and Server Options for CICS

} TTLSEnvironmentAction TTLSENV3 {
    HandshakeRole  Client
    EnvironmentUserInstance  1
    TTLSKeyRingParms {
        Keyring  CICSDEV1  # Refers to RACF database
    }
}

5. Set up InitStack access control

You need to define the security product authorization for PAGENT.

When using AT-TLS, z/OS will not allow any socket-based applications to start before PAGENT is up and running. This restriction is needed to verify that all the security policies are enforced. However, some essential applications need to start before PAGENT. For these applications, you need to define a resource INITSTACK profile in the SERVAUTH class. The resource name consists of the following parts:

- **EZB** – the constant.
- **INITSTACK** – the constant for this resource type.
- **sysname** – the system name.
- **tcpprocname** – the TCP/IP proc name.

When TCPCONFIG TTLS is defined in the initial TCPIP.PROFILE, the INITSTACK profile must be defined. Policy Agent—and any socket-based programs it requires—must be given permission to this resource.

**Note** Be sure that the program name is the name used to invoke the program—*not* the module name.

Most TCP/IP applications are invoked by ALIAS name. This example lists both names:
APPENDIX D Setting Up Secure Sockets Layer Protocol

SETROPTS CLASSACT(SERVAUTH)
SETROPTS RACLIST (SERVAUTH)
SETROPTS GENERIC (SERVAUTH)
RDEFINE SERVAUTH EZB.INITSTACK.*.TCP* UACC(NONE) PERMIT
EZB.INITSTACK.*.TCP* -
   CLASS(SERVAUTH) ID(*) ACCESS(READ) -
   WHEN(PROGRAM(PAGENT,EZAPAGEN)
SETROPTS GENERIC(SERVAUTH) REFRESH
SETROPTS RACLIST(SERVAUTH) REFRESH
SETROPTS WHEN(PROGRAM) REFRESH

6. Enable AT-TLS

To enable AT-TLS functionality for a stack, define the TCPCONFIG TTLS parameter in the TCPIP.PROFILE.

7. Run SSL secure connections

After you complete all previous configuration tasks, you are ready to start the Policy Agent. Here is the example of the PAGENT procedure used:

```
//PAGENT   PROC PARMS='-c /etc/pagent.conf3 -l /tmp/pagent.log4'
//PAGENT   EXEC PGM=PAGENT,REGION=0K,TIME=NOLIMIT,
// PARM=('POSIX(ON) ALL31(ON)
// 'ENVAR("LIBPATH=/usr/lib",
// '"TZ=MST7MDT6")/&PARMS')
//STDENV DD PATH='/etc/pagent.env',PATHOPTS=(ORDONLY)
//SYSPRINT DD SYSOUT=* 
//SYSPRINT DD SYSOUT=* 
//CEEDUMP DD SYSOUT=*,DCB=(RECFM=FB,LRECL=132,BLKSIZE=132)
```

where `/etc/pagent.env` contains these definitions:

- `LIBPATH=/usr/lib`
- `TZ=MST7MDT6`

After PAGENT starts successfully, start the stack TCPIPT and the CICS region called “CICSDEV1” on that stack. If the server called “ase1” is running, you are ready to test your connections.
For more information

Refer to the IBM z/OS Communication Server: IP Configuration Guide for more information about these topics:

- For an overview of the policy-based networking and components, see the chapter called “Policy-based Networking.”
- For information on the AT-TLS security application and configuration, see the chapter called “AT-TLS Data Protection.”
- For an overview of using digital certificates, generating them, and handling them, see the appendix called “TLS/SSL Security.”

For detailed information on Policy Agent statements and policy applications, see the chapter called “Policy Agent Statements and Policy Applications” in the IBM z/OS Communication Server: IP Configuration Reference.

For more information about using digital certificates, generating them, and handling them, refer to the chapter called “RACF and Digital Certificates” in the z/OS Security Server RACF Security Administrator Guide.

For detailed information about using the RACDCERT (RACF Digital Certificate) command, refer to the z/OS Security Server RACF Command Language Reference.

For diagnosing AT-TLS or Policy Agent problems, refer to the appropriate chapters in the z/OS Communication Server: IP Diagnosis Guide.

For information on using digital certificates with other security products, refer to related documentation.

For more information on how to obtain digital certificates, configure and use SSL connections with the Sybase products, see the appropriate Sybase product documentation.
## Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept</td>
<td>Establishment of a SNA or TCP/IP connection between Mainframe Connect Server Option and Mainframe Connect DirectConnect for z/OS Option.</td>
</tr>
<tr>
<td>access service</td>
<td>The named set of properties, used with an access service library, to which clients connect. Each DirectConnect server can have multiple services.</td>
</tr>
<tr>
<td>access code</td>
<td>A number or binary code assigned to programs, documents, or folders that allows authorized users to access them.</td>
</tr>
<tr>
<td>access service library</td>
<td>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.</td>
</tr>
<tr>
<td>ACSLIB</td>
<td>See access service library.</td>
</tr>
<tr>
<td>Adaptive Server Enterprise</td>
<td>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.</td>
</tr>
<tr>
<td>Adaptive Server Enterprise/Component Integration Services</td>
<td>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.</td>
</tr>
<tr>
<td>administrative service library</td>
<td>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.</td>
</tr>
<tr>
<td>ADMLIB</td>
<td>See administrative service library.</td>
</tr>
<tr>
<td>Advanced Interactive Executive</td>
<td>The IBM implementation of the UNIX operating system. The RISC System/6000, among other workstations, runs the AIX operating system.</td>
</tr>
<tr>
<td>advanced program-to-program communication</td>
<td>Hardware and software that characterize the LU 6.2 architecture and its implementations in products. See also logical unit 6.2.</td>
</tr>
</tbody>
</table>

*Installation and Administration Guide*
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX</td>
<td>See <strong>Advanced Interactive Executive</strong>.</td>
</tr>
<tr>
<td>AMD2</td>
<td>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.</td>
</tr>
<tr>
<td>American Standard Code for Information Interchange</td>
<td>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).</td>
</tr>
<tr>
<td>API</td>
<td>See <strong>application program interface</strong>.</td>
</tr>
<tr>
<td>APPC</td>
<td>See <strong>advanced program-to-program communication</strong>.</td>
</tr>
<tr>
<td>application program interface</td>
<td>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.</td>
</tr>
<tr>
<td>ASCII</td>
<td>See <strong>American Standard Code for Information Interchange</strong>.</td>
</tr>
<tr>
<td>ASE</td>
<td>See <strong>Adaptive Server Enterprise</strong>.</td>
</tr>
<tr>
<td>ASE/CIS</td>
<td>See <strong>Adaptive Server Enterprise/Component Integration Services</strong>.</td>
</tr>
<tr>
<td>batch</td>
<td>A group of records or data processing jobs brought together for processing or transmission.</td>
</tr>
</tbody>
</table>
| bind                    | In the Sybase environment, this term has different meanings depending on the context:  
  - 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. |
**Glossary**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>bulk copy transfer</td>
<td>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.</td>
</tr>
<tr>
<td>call level interface</td>
<td>A programming style that calls database functions directly from the top level of the code. Contrast with embedded SQL.</td>
</tr>
<tr>
<td>catalog</td>
<td>A system table that contains information about objects in a database, such as tables, views, columns, and authorizations.</td>
</tr>
<tr>
<td>catalog RPC</td>
<td>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.</td>
</tr>
<tr>
<td>catalog stored procedure</td>
<td>A procedure used in SQL generation and application development that provides information about tables, columns, and authorizations.</td>
</tr>
<tr>
<td>character set</td>
<td>A set of specific (usually standardized) characters with an encoding scheme that uniquely defines each character. ASCII is a common character set.</td>
</tr>
<tr>
<td>CICS</td>
<td>See Customer Information Control System.</td>
</tr>
<tr>
<td>CICS region</td>
<td>The instance of CICS.</td>
</tr>
<tr>
<td>client</td>
<td>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.</td>
</tr>
<tr>
<td>client application</td>
<td>Software responsible for the user interface that sends requests to applications acting as servers. See also client/server.</td>
</tr>
<tr>
<td>Client-Library</td>
<td>A library of routines that is part of Mainframe Connect Client Option.</td>
</tr>
<tr>
<td>client request</td>
<td>An RPC or language request sent by a client to a server.</td>
</tr>
<tr>
<td>client/server</td>
<td>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.</td>
</tr>
<tr>
<td>Client Services Application</td>
<td>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 Client Option server or to ASE. See also application program interface, Client Services for CICS.</td>
</tr>
</tbody>
</table>
**Glossary**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Client Services for CICS</strong></td>
<td>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 <a href="#">application program interface</a>, <a href="#">Customer Information Control System</a>, <a href="#">Client Services Application</a>, <a href="#">Mainframe Connect Server Option</a>.</td>
</tr>
<tr>
<td><strong>clustered index</strong></td>
<td>An index in which the physical order and the logical (indexed) order is the same. Compare with <a href="#">nonclustered index</a>.</td>
</tr>
<tr>
<td><strong>code page</strong></td>
<td>An assignment of graphic characters and control function meanings to all code points.</td>
</tr>
<tr>
<td><strong>commit</strong></td>
<td>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 <a href="#">rollback</a>.</td>
</tr>
<tr>
<td><strong>Common Programming Interface</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>configuration file</strong></td>
<td>A file that specifies the characteristics of a system or subsystem.</td>
</tr>
<tr>
<td><strong>configuration set</strong></td>
<td>A section into which service library configuration files are divided.</td>
</tr>
<tr>
<td><strong>conversion</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>connection</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>connection router</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Connection Router Table</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>control section</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>control statement</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>conversation-level security</strong></td>
<td>The passing of client login information to the mainframe by TRS when it allocates a conversation.</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>CSA</strong></td>
<td>See <strong>Client Services Application</strong>.</td>
</tr>
<tr>
<td><strong>CSP</strong></td>
<td>See <strong>catalog stored procedure</strong>.</td>
</tr>
<tr>
<td><strong>cursor</strong></td>
<td>In SQL, a named control structure used by an application program to point to a row of data.</td>
</tr>
<tr>
<td><strong>Customer Information Control System</strong></td>
<td>An IBM licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs.</td>
</tr>
<tr>
<td><strong>DASD</strong></td>
<td>See <strong>direct access storage device</strong>.</td>
</tr>
<tr>
<td><strong>data definition statement</strong></td>
<td>An IBM mainframe statement used to relate a name with a file.</td>
</tr>
<tr>
<td><strong>data definition language</strong></td>
<td>A language for describing data and data relationships in a database.</td>
</tr>
<tr>
<td><strong>data set name</strong></td>
<td>The term or phrase used to identify a data set.</td>
</tr>
<tr>
<td><strong>database management system</strong></td>
<td>The term or phrase to identify a data set. A computer-based system for defining, creating, manipulating, controlling, managing, and using databases.</td>
</tr>
<tr>
<td><strong>database operation</strong></td>
<td>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 <strong>request</strong>.</td>
</tr>
<tr>
<td><strong>Database 2</strong></td>
<td>An IBM relational database management system.</td>
</tr>
<tr>
<td><strong>datatype</strong></td>
<td>A keyword that identifies the characteristics of stored information on a computer.</td>
</tr>
<tr>
<td><strong>DB-Library</strong></td>
<td>A Sybase and Microsoft API that allows client applications to interact with ODS applications. See also <strong>application program interface</strong>.</td>
</tr>
<tr>
<td><strong>DBMS</strong></td>
<td>See <strong>database management system</strong>.</td>
</tr>
<tr>
<td><strong>DB2 UDB</strong></td>
<td>See <strong>Database 2</strong>.</td>
</tr>
<tr>
<td><strong>DDL</strong></td>
<td>See <strong>data definition language</strong>.</td>
</tr>
<tr>
<td><strong>DD statement</strong></td>
<td>See <strong>data definition statement</strong>.</td>
</tr>
<tr>
<td><strong>default language</strong></td>
<td>The language that displays a user’s prompts and messages.</td>
</tr>
</tbody>
</table>
### Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>destination-template transfer</td>
<td>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.</td>
</tr>
<tr>
<td>direct access storage device</td>
<td>A device in which access time is effectively independent of the location of the data.</td>
</tr>
<tr>
<td>direct request</td>
<td>A request sent directly from a client workstation through Transaction Router Service to the DirectConnect server without going through ASE. Contract with indirect request.</td>
</tr>
<tr>
<td>direct resolution</td>
<td>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.</td>
</tr>
<tr>
<td>DirectConnect Manager</td>
<td>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.</td>
</tr>
<tr>
<td>Client Option server</td>
<td>The component of Mainframe Connect DirectConnect for z/OS Option that provides general management and support functions to service libraries.</td>
</tr>
<tr>
<td>dll</td>
<td>See dynamic link library.</td>
</tr>
<tr>
<td>DSN</td>
<td>See data set name.</td>
</tr>
<tr>
<td>dynamic link library</td>
<td>A file containing executable code and data bound to a program at load time or runtime, rather than during linking.</td>
</tr>
<tr>
<td>dynamic SQL</td>
<td>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.</td>
</tr>
<tr>
<td>ECDA</td>
<td>See Enterprise Connect Data Access.</td>
</tr>
<tr>
<td>ECDA Option for ODBC</td>
<td>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.</td>
</tr>
<tr>
<td>ECDA Option for Oracle</td>
<td>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.</td>
</tr>
</tbody>
</table>
**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.
Glossary

**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 interfaces file entry contains the name of any Client 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.
<table>
<thead>
<tr>
<th><strong>mainframe access products</strong></th>
<th>sybase products that enable client applications to communicate with mainframes in a client/server environment. see <a href="#">client/server</a>.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mainframe Connect</strong></td>
<td>the sybase product set that provides access to mainframe data.</td>
</tr>
<tr>
<td><strong>Mainframe Connect Client Option</strong></td>
<td>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 environments.</td>
</tr>
<tr>
<td><strong>Mainframe Connect DB2 UDB Option</strong></td>
<td>a sybase mainframe solution that provides dynamic access to db2 udb data. it is available in the cics or ims environment. see also <a href="#">customer information control system</a>, <a href="#">database 2</a>, <a href="#">multiple virtual storage</a>.</td>
</tr>
<tr>
<td><strong>Mainframe Connect Client Option for z/OS Option</strong></td>
<td>a sybase open server application that provides access management for non-sybase databases, copy management (transfer), and remote systems management.</td>
</tr>
<tr>
<td><strong>Mainframe Connect Server Option</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Multiple Virtual Storage</strong></td>
<td>an ibm operating system that runs on most system/370 and system/390 mainframes. it supports 24-bit addressing up to 16 megabytes.</td>
</tr>
<tr>
<td><strong>network protocol</strong></td>
<td>a set of rules governing the way computers communicate on a network.</td>
</tr>
<tr>
<td><strong>nonclustered index</strong></td>
<td>an index that stores key values and pointers to data. compare with <a href="#">clustered index</a>.</td>
</tr>
<tr>
<td><strong>null</strong></td>
<td>having no explicitly assigned value. null is not equivalent to 0 or to blank.</td>
</tr>
<tr>
<td><strong>ODBC</strong></td>
<td>see <a href="#">open database connectivity</a>.</td>
</tr>
<tr>
<td><strong>ODS</strong></td>
<td>see <a href="#">open data services</a>.</td>
</tr>
<tr>
<td><strong>Open Client</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Open Data Services</strong></td>
<td>a product that provides a framework for creating server applications that respond to db-library clients.</td>
</tr>
<tr>
<td><strong>Open Database Connectivity</strong></td>
<td>a microsoft api that allows access to both relational and non-relational databases. see also <a href="#">application program interface</a>.</td>
</tr>
<tr>
<td><strong>Glossary</strong></td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td></td>
</tr>
</tbody>
</table>

**Open Server**
A Sybase product that provides the tools and interfaces required to create a custom server. Clients can route requests to the Client 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.
<table>
<thead>
<tr>
<th><strong>Glossary</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>remote stored procedure</strong></td>
</tr>
<tr>
<td><strong>remote systems management</strong></td>
</tr>
<tr>
<td><strong>Replication Server</strong></td>
</tr>
<tr>
<td><strong>request</strong></td>
</tr>
<tr>
<td><strong>resource table</strong></td>
</tr>
<tr>
<td><strong>rollback</strong></td>
</tr>
<tr>
<td><strong>router</strong></td>
</tr>
<tr>
<td><strong>RPC</strong></td>
</tr>
<tr>
<td><strong>RSP</strong></td>
</tr>
<tr>
<td><strong>SAA</strong></td>
</tr>
<tr>
<td><strong>secondary connection</strong></td>
</tr>
<tr>
<td><strong>secondary database</strong></td>
</tr>
<tr>
<td><strong>server</strong></td>
</tr>
<tr>
<td><strong>server process ID</strong></td>
</tr>
<tr>
<td><strong>service</strong></td>
</tr>
</tbody>
</table>
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 sql.ini files and file entries.

sql.ini

The interfaces file containing definitions for each Client 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.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>stored procedure</strong></td>
<td>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 <a href="#">remote stored procedure</a>, <a href="#">system procedures</a>.</td>
</tr>
<tr>
<td><strong>structured query language</strong></td>
<td>An IBM industry-standard language for processing data in a relational database.</td>
</tr>
<tr>
<td><strong>stub</strong></td>
<td>A program module that transfers remote procedure calls (RPCs) and responses between a client and a server.</td>
</tr>
<tr>
<td><strong>SYRT</strong></td>
<td>The component of Mainframe Connect DB2 UDB for IMS that allows clients to submit SQL language requests to DB2 through IMS.</td>
</tr>
<tr>
<td><strong>System Administrator</strong></td>
<td>The person in charge of server system administration, including installing and maintaining DirectConnect servers and service libraries.</td>
</tr>
<tr>
<td><strong>System Application Architecture</strong></td>
<td>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 <a href="#">advanced program-to-program communication</a>.</td>
</tr>
<tr>
<td><strong>system procedures</strong></td>
<td>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 <a href="#">stored procedure</a>.</td>
</tr>
<tr>
<td><strong>Systems Network Architecture</strong></td>
<td>An IBM proprietary plan for the structure, formats, protocols, and operational sequences for transmitting information units through networks. See also <a href="#">advanced program-to-program communication</a>.</td>
</tr>
<tr>
<td><strong>table</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Tabular Data Stream</strong></td>
<td>A Sybase application-level protocol that defines the form and content of relational database requests and replies.</td>
</tr>
<tr>
<td><strong>target</strong></td>
<td>A system, program, or device that interprets, rejects, satisfies, or replies to requests received from a source.</td>
</tr>
<tr>
<td><strong>target database</strong></td>
<td>The database to which the DirectConnect server transfers data or performs operations on specific data.</td>
</tr>
<tr>
<td><strong>TCP/IP</strong></td>
<td>See <a href="#">Transmission Control Protocol/Internet Protocol</a>.</td>
</tr>
<tr>
<td><strong>TDS</strong></td>
<td>See <a href="#">Tabular Data Stream</a>.</td>
</tr>
</tbody>
</table>
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.
### Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>variable</td>
<td>An entity that is assigned a value. Mainframe Connect Client Option for z/OS Option has two kinds of variables: <em>local</em> and <em>global</em>.</td>
</tr>
<tr>
<td>view</td>
<td>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.</td>
</tr>
<tr>
<td>Virtual Storage Access Method</td>
<td>An IBM-licensed program that controls communication and the flow of data in an SNA network.</td>
</tr>
<tr>
<td>Virtual Telecommunications Access Method</td>
<td>IBM mainframe software that allows communication on an SNA network between mainframes and allows the mainframe to have multiple sessions per connection.</td>
</tr>
<tr>
<td>VSAM</td>
<td>See Virtual Storage Access Method.</td>
</tr>
<tr>
<td>VTAM</td>
<td>See Virtual Telecommunications Access Method.</td>
</tr>
<tr>
<td>wildcard</td>
<td>A special character that represents a range of characters in a search pattern.</td>
</tr>
</tbody>
</table>
Index

A
ACCESSCODE SYGWMCST parameter 30
ACCESSCODESW SYGWMCST parameter 30
ASCII_8 translation tables
  ASCII_8 ACSII-to-EBCDIC 48
  ASCII_8 EBCDIC-to-ASCII 48
ASE
  setting up SSL 64
asymmetric cryptography 62
AT-TLS
  adding statements to the TTLSConfig policy file 74
  adding the configuration file 74
  configuring 67
  configuring with z/OS client or server 71
  creating a key ring 71
  enabling 77
  rule conditions for a connection 69
  secure connections 66
  setting up InitStack access control 76
  support in Client and Server Options for CICS 70
types of configuration files 68
authentication levels in SSL 65
authorization key
  ordering 15

B
binary formats
  in certificate encoding 73
browsing
  setting with isql 24

C
CEDA window 40
certificate encoding 73
certificates
  activating for z/OS 72
  generating 63
  installing client copies 64
  key database 66
CHARSETSRY SYGWMCST parameter 30
choosing
  dynamic network drivers 3, 7
  network drivers 3, 7
CICS
  network driver 39
CICS region
  installing CICS sockets interface 60
CICS sockets interface 59
description 59, 61
EZACIC20 program 60
EZACONFG data set 60
installing and configuring 59, 60, 61
RDO entries 60
setting up 59, 61
Client Option
  communication at the mainframe 6
  communication at the server 6
  network configuration 2
  security 7
commands
  exporting new certificate to a data set 72
  EZAO 60
  open 66
  TCPCONFIG 68
  to connect a certificate to a key ring 72
  to define a key ring 72
  to generate a self-signed certificate 72
  to refresh after changes to key ring 72
commands to activate certificate and key ring classes 72
concatenating
  sockets load library 60
configuration
  creating Policy Agent files 73
Index

configuration files  
AT-TLS 68

configuring  
AT-TLS 67
CICS sockets interface 59, 60, 61
tasks for AT-TLS with z/OS 71
z/OS client or server for AT-TLS 71

connections  
AT-TLS security 66
AT-TLS security rules 69

connectivity  
verifying gateway-less 13
verifying two-tier 13

cp437 translation tables  

cp437 ASCII-to-EBCDIC 54
cp437 EBCDIC-to-ASCII 55

cp850 translation tables  
cp850 ASCII-to-EBCDIC 57
cp850 EBCDIC-to-ASCII 58

CPI-C CICS network driver 39

creating  
key ring for AT-TLS 71
SSL environment layer 66
customization options  
SYGWDRIV 39
SYGWHOST, TCP/IP configuration macro 41
customization table SWGWXCPH 29
SYGWMCST global macro 29
SYGWMCXL character set macro 29
customizing  
a network driver 39
global 30
LAN-side character sets 41
options 29

D
DEBUGSW SYGWMCST parameter 30
DECPNT SYGWMCST parameter 31

DQUOTETRN SYGWMCST parameter 31
dynamic network driver  
choosing 3, 7
CICS 39
CPI-C CICS 39
customizing 39
defining 2
invoking 2
loading 2
macro 39
network type and environment 3

E
enabling tracing 24
encryption  
levels in SSL 65
SSL 62
entering SQL commands 26
environment  
gateway-less 2
two-tier 2
environment variables  
Policy Agent 70
Establish Open Client Server Connection window 25
EZACIC20 program  
CICS sockets interface 60
EZACONFG data set 60
EZAO command 60

F
function key listing  
isql 25

G
gateway-less  
description 2
verifying connectivity 13
generating  
certificates 63
self-signed certificate 72

Mainframe Connect Client Option for CICS
global customization (SYGWMCST) 29
list of parameters 30

I
IMSLOGTYPE SYGWMCST parameter 31
Initstack
setting up for access control in AT-TLS 76
installing
CICS sockets interface 59, 60, 61
Integrated Cryptographic Service Facility (ICSF) 66
invoking
dynamic network drivers 2
network drivers 2
ISO_1 translation tables
ISO_1 ASCII-to-EBCDIC 51
ISO_1 EBCDIC-to-ASCII 51
isql
about 6
browsing 24
tracing 24
viewing results 27

K
key database
certificate information 66
key ring
defining AT-TLS for z/OS 72
keyboard definitions
isql functions 25

L
LAN-side character sets
customizing 41
LIBPATH 70
library names
using new 13
licensing key
temporary 15
loading
dynamic network drivers 2
network drivers 2
LONGVARTRUNC SYGWMCST parameter 31

M
macros
SWGWHOST 41
SYGWDRIV 3, 39
mainframe character set customization options
(SYGWMCXL) 33
overriding SBCS translation tables 34
SBCS, customizing translation 35
SBCS, predefined character sets 34
SBCS, user-defined character sets 34
MVSDDDNAME SYGWMCST parameter 32

N
NATLANGUAGESRV SYGWMCST parameter 32
network communication definitions
choosing a network driver 3
overview 3
network driver
choosing 3, 7
CICS 39
CPI-C CICS 39
customizing 39
defining 2
invoking 2
loading 2
macro 39
network type and environment 3
network type and environment
dynamic network driver 3
network driver 3

O
Open Client
installing client copies of CA certificates 64
setting up SSL 64
open command 66
ordering a permanent authorization key 15
Index

P
PAGENT_CONFIG_FILE 70
PAGENT_LOG_FILE 70
PAGENT_LOG_FILE_CONTROL 70
parameters
SYGWMCST macro 30
Policy Agent
creating configuration files 73
environment variables 70
environment variables search order for configuration files 70
starting 69, 77
private-key cryptography 62
public-key cryptography 62

R
RDO entries for CICS sockets interface 60
reference information
for SSL 78
result tracing 27
ROWLIMIT SYGWMCST parameter 32

S
sample network diagram 5
SBCS translation tables 45
 ASCII_8 47
cp437 51
cp850 55
ISO_1 50
SBCS translation, customizing
overriding defaults with SYGWMCXL 36
Secure Sockets Layer. See SSL
security 7
for Client Option 7
Secure Sockets Layer 7
using SSL 61
Server-Host Mapping table
about 6
setting up
CICS sockets interface 59, 61
SSL in z/OS 65
sockets load library
concatenating 60
SQL command entry 26
SQL commands
entering 26
SSL
authentication and encryption 62
authentication and encryption levels 65
creating an environment layer 66
description 61
features 61
private-key cryptography 62
public-key cryptography 62
reference information 78
running secure connections with Policy Agent 77
setting up in ASE 64
setting up in IBM z/OS 65
setting up in Open Client 64
System SSL 65
X.509 certificates 62
starting
Policy Agent 69
SYGMCST 29
SYGWDRIV macro 3, 39
SYGWHOST
formats 41
macro 41
SYGWMCST parameters 30
 ACCESSCODE 30
 ACCESSCODESW 30
 CHARSERTSV 30
 DEBUGSW 30
 DECPPOINT 31
 DQUOTETRAN 31
 IMSLOGTYPE 31
 LONGVARTIRUNC 31
 MVSDDDNAME 32
 NATLANGUAGESRV 32
 ROWLIMIT 32
 USEIBMUNICODE 32
 SYGWMCXL 33
 SYGWXCPH 3
definitions 7
symmetric cryptography 62
System SSL
using on z/OS 66

Mainframe Connect Client Option for CICS
Index

T
  tasks
    configuration for AT-TLS with z/OS  71
  TCP/IP
    using the SYGWHOST macro  7
    TCPCONFIG command  68
    temporary licensing key  15
    Trace/Error Display results  27
  tracing
    enabling  24
    setting with isql  24
  translation tables, SBCS  45
  trusted.text file  64
  two-tier
    description  2
    verifying connectivity  13
  TZ  70

U
  UNIX System Services (USS)  65
  USEIBMUNICODE SYGWMCST parameter  32

V
  viewing trace results  27

W
  windows
    CEDA  40
    Establish Open Client Server Connection  25

X
  X.509 certificates
    in SSL  62

Z
  z/OS
    setting up SSL  65
    using System SSL  66