Contents

About This Manual ................................................................. ix
Audience ......................................................................................... ix
How This Manual is Organized ..................................................... x
Recommended Reading ............................................................... xii
  Common Manuals ........................................................................... xii
  Optional Manuals ........................................................................... xiii
Related Reading ............................................................................. xiii
Syntax Conventions ...................................................................... xiv

Chapter 1 Getting Started ...................................................... 1

Chapter 2 Logon Director ..................................................... 3
Logon Director Actions at User Sign On........................................... 4
User Sign On Retry limit ............................................................... 5
Automating User Sign Ons ............................................................ 6
  Signing On at the Logon Director Screen ...................................... 7
  Receiving sign on Data From Another Application ....................... 8
  Customized Sign On Applications ................................................. 9
  VTAM Session Managers and Menu Systems ......................... 9
Other CICS Regions .................................................................... 10
Pre-defined sign on Data to Specific Terminals ............................... 11
Pre-defined Sign On Data to Bypass the Logon Director Screen .... 12
Part 1: Specifying Terminals to Bypass the Logon Director Screen 13
Part 2: Setting Up CICS Sign on .................................................. 15
Receiving Sign On Data ................................................................. 16
Sign on Message Display Option ................................................. 17
Customizing the Logon Director Screen ....................................... 18
  Authorization to Customize a Logon Director Screen .................. 18
  Format of the Logon Director Screen ......................................... 19
  Editing a Log On Screen ............................................................. 21
Maintaining the Logon Director Screen ....................................... 25
  Panel Definition Screen Commands and PF Keys ......................... 25
  Creating a Log On Screen .......................................................... 26
  Deleting a Log On Screen .......................................................... 26
  Renaming a Log On Screen ......................................................... 27
  Assigning Log On Screens ......................................................... 27
  Editing Log On Screen Help Members ......................................... 27

Chapter 3 Profiles ............................................................... 29
Assigning User Profiles ............................................................... 30
Chapter 4  Terminal and User Directories ...............43

Chapter 5  Customization Options ....................65

Chapter 6  Security ...........................................73
Chapter 7  Transparent Mode .............................. 89
Logon Director Screen ........................................ 90
Sessions Menu ...................................................... 90
PIE Pop-Up Menus ...................................................90
PIE Commands .......................................................90
System End Keys .......................................................91
AutoLock .................................................................91
Disconnected Sessions ...........................................91
Logging Off ............................................................92
Multiple Sessions .....................................................92
PIE Global Keys .......................................................92
View ................................................................. 92
Examples of Implementing Transparent Mode ................. 93
Example 1: Implementing Two Blank Screens ............... 93
Example 2: AutoLock Only ...................................... 96
Example 3: View Only .............................................. 98

Chapter 8  Multiple CICS Regions ......................... 101
MRO/ISC Support ................................................. 102
Sharing the Options File Between Regions ................... 103
Using Separate Options Files for Each Region .............. 103
Initializing PIE in an AOR ........................................ 103
Multiple Stand-alone Systems .................................. 104
Sharing Repository and Options Files ......................... 105
Using Different Options or Repository Files ................. 106
Running Multiple Versions of CICS ......................... 107

Chapter 9  VTAM Applications .............................. 109
Application Transfer Methods and Options .................. 110
Differences Between PIE/CICS Transfer Session Commands 110
Customizing Transfer Commands ............................... 113
Automating Log on Procedures ................................ 113
Controlling Target Application Programs ..................... 116
Coding the Execution String .................................... 117
Using the VTAM Log on Message .............................. 117
Using Scripts to Log on to Target Applications .............. 118
Dynamic Menus ..................................................... 118
Dynamic Transaction Routing .................................. 119
Preparing Dynamic Transaction Routing .................... 119
Coding the Execution String: Determining the Target Region 120
Including the Access Method in the Command String .......... 121
Dynamic Menus ..................................................... 121
About This Manual

This manual describes procedures to prepare PIE/CICS for use after it has been installed. Each chapter consists of a series of procedures to adapt a functional part of PIE/CICS to your site's system.

This manual describes how to prepare the common, shared PIE/CICS components. Separate manuals describe how to prepare the optional components that are part of the PIE family of products. This manual is a reference. The optional product manuals make extensive references to the procedures described in this manual to complete their own customization procedures. You should use this manual along with these optional manuals as you customize the optional components.

Audience

This book is intended for system administrators responsible for installing and customizing MVS software at their site. Readers are expected to understand MVS and CICS concepts. Also, many of the procedures described in this manual require site-specific JCL changes to run batch customization jobs. Readers should be proficient editing JCL and be familiar with their site's system standards.
How This Manual is Organized

This manual consists of 11 chapters and 8 appendixes. Listed below are the titles and a brief description of each chapter and appendix.

- **Chapter 1  Getting Started**  
  Gives an overview of PIE/CICS customization procedures.

- **Chapter 2  Logon Director**  
  Describes procedures to adapt the Logon Director and VTAM session managers to permit users to log on to PIE/CICS.

- **Chapter 3  Profiles**  
  Explains how to prepare profiles that set the operating conditions of a PIE/CICS working environment for individual users.

- **Chapter 4  Terminal and User Directories**  
  Describes procedures to prepare both directories that grant access to a PIE/CICS system.

- **Chapter 5  Customization Options**  
  Describes how to use the Customization Options utility that sets system-wide operating conditions for PIE/CICS.

- **Chapter 6  Security**  
  Explains how to authorize PIE/CICS resources to an external security system and specify object security.

- **Chapter 7  Transparent Mode**  
  Describes procedures to select a subset of PIE/CICS utilities and functions and implement them into a CICS environment without being recognized by the user community.

- **Chapter 8  Multiple CICS Regions**  
  Explains how to prepare PIE/CICS to support multiple CICS regions running under MRO/ISC.

- **Chapter 9  VTAM Applications**  
  Explains how transfer between VTAM applications with PIE/CICS commands and use Dynamic Transaction Routing to distribute transactions across remote regions.

- **Chapter 10  PIE/CICS Exits**  
  Describes how to customize PIE/CICS stand-alone and PUX1 exits.

- **Chapter 11  Performance and Tuning**  
  Describes tuning recommendations to improve PIE/CICS performance.

- **Appendix A  Customer Service**  
  Describes procedures to report problems with PIE/CICS to UNICOM Systems Software Customer Service.

- **Appendix B  Exit and System Table Worksheets**  
  Provides several worksheets to record changes or other modifications to PIE/CICS exits and tables.
• Appendix C  Compatibility with Other Software
  Describes compatibility issues with other MVS software that runs with PIE/CICS in a
  CICS system.
• Appendix D  PIE/CICS API Overview
  Summarizes the function of Application Programming Interfaces (APIs) used by
  PIE/CICS.
• Appendix E  PIE/CICS Messages
  Describes procedures to edit the text of PIE/CICS messages.
• Appendix F  Cutting and Pasting Across Regions
  Explains how to cut and paste data between separate session screens with the
  PIE/CICS cut and paste functions.
• Appendix G  PIE/CICS System Tables
  Describes procedures to modify PIE/CICS system tables.
• Appendix H  JCL Parameters
  Describes common JCL statements found in the batch jobs to complete
  customization procedures.
Recommended Reading

The title and a brief description of all PIE/CICS manuals are shown in the following lists. Some manuals provide common information that applies to both the common and optional components of PIE/CICS. Other manuals pertain only to optional PIE/CICS components. Optional manuals need to be read only if these components are part of the PIE/CICS system installed at your site.

Common Manuals

These manuals provide common information that applies to both the shared and optional components of the PIE/CICS family.

- PIE/CICS Installation Guide
  Includes a series of procedures to install PIE/CICS.
- PIE/CICS Release Notes
  Describes new features or enhancements introduced with PIE/CICS Release 3.6.0
- PIE/CICS Command Reference
  Describes PIE/CICS Application and Environment commands.
- PIE/CICS Customization Reference
  Describes common procedures to adapt PIE/CICS to your site’s requirements.
- PIE/CICS Operation and Administration Guide
  Describes common features or facilities that are available to all PIE/CICS products. Performance tuning techniques and implementing security also are described.
- REXX for PIE/CICS User Guide
  Describes how to write, compile, and execute SAA-compliant REXX programs that operate within a PIE/CICS external environment.
- PIE/CICS Custom Menus Administration Guide
  Describes how to create custom MultiCICS and Dynamic Menu screens that provide alternate language support.
Optional Manuals

These manuals describe optional PIE/CICS components.

- **PIE/CICS MultiCICS Administration Guide**
  Provides customization procedures and usage information to support multiple PIE/CICS sessions with MultiCICS.

- **PIE/CICS Dynamic Menus Administration Guide**
  Describes how to create custom PIE/CICS menus that provide extended security and enhanced transaction processing.

- **PIE/CICS NetGate Administration Guide**
  Explains how to access multiple VTAM applications through a PIE/CICS session with NetGate.

- **PIE/CICS NetMizer Administration Guide**
  Describes how to use NetMizer to optimize 3270-based data streams.

- **PIE/CICS Availability Plus Administration Guide**
  Explains how to use Availability Plus to distribute and balance work across multiple CICS regions.

- **PIE/CICS NonStop CICS Administration Guide**
  Describes how to use NonStop CICS to route work across CICS regions to balance the workload and minimize down time in the event of a region failure.

Related Reading

Refer to the following IBM manuals for VTAM information:

- **VTAM for MVS/ESA Messages and Codes, SC31-6546**
- **VTAM for MVS/ESA Network Implementation Guide, SC31-6548**
- **VTAM for MVS/ESA Resource Definition Reference, SC31-6548**
Syntax Conventions

A syntax diagram is included with each PIE/CICS command described in this manual. A syntax diagram shows the possible parameters, values, and variables associated with a command.

Syntax diagrams adhere to common conventions. The physical appearance of a diagram’s elements indicates whether a command parameter, variable, or other values are required, optional, or included by default.

- An **underlined** parameter is the default assigned to the command.
- Command names are presented in MIXed case. The uppercase portion of a command name is the requisite abbreviated form. Lowercase letters represent the optional remainder of the command name that need not be specified to execute the command.
- An italicized lowercase parameter represents a value assigned by the user.
- A vertical bar (|) separates two or more mutually exclusive parameter values. Only one value can be specified for each parameter.
- Parameters enclosed within brackets [ ] are optional. Only one value can be specified to a parameter.
- Parameters values enclosed within braces { } are required. If unspecified, the parameter default is assigned to the command.
- **Monospaced** type represents text displayed on a PIE/CICS screen or examples of JCL code. Also, commands entered on a PIE/CICS screen are shown as monospace examples.
Chapter 1  Getting Started

PIE/CICS is an integrated family of common and optional components. All components share certain PIE/CICS resources. The Logon Director, security, and profiles are examples of shared utilities and resources that are used by every component within the family. This manual describes how to prepare these shared utilities and resources after PIE/CICS has been installed.

Separate manuals are included for each optional PIE/CICS component. These manuals describe any additional customization procedures to prepare these optional components to work with PIE/CICS. The manuals are written specifically for a single product. However, they do not describe the resources and features that are common to all components. As you customize each product, follow the product manual and refer to this manual for additional information.

PIE/CICS provides a broad range of features to CICS systems. It is helpful to gain a general understanding of how it works before you begin planning and implementation. We suggest that you browse this manual and the PIE/CICS Administration and Operation Guide early in the implementation process. These manuals provide an overview of PIE/CICS. Refer to specific component manuals for a more detailed description of PIE/CICS facilities.

When you are ready to begin implementing PIE/CICS, install using the instructions in the PIE/CICS Installation Guide. Then, follow the customization instructions in the individual manuals for your PIE/CICS products.
Chapter 2  Logon Director

Users must log on through the Logon Director before they can use PIE/CICS services. Logging on can occur at the Logon Director screen, or by connecting the Logon Director to another sign on application.

When you first set up PIE/CICS, you should use the Logon Director the way it is shipped. After you have become more familiar with PIE/CICS, you will be in a better position to evaluate and implement other Logon Director options.

The Logon Director program is PCTSPGMM. Its default transaction code is PSGM.
Logon Director Actions at User Sign On

The Logon Director performs the following actions to log on a user to PIE/CICS.

- If the terminal is currently signed on to PIE/CICS, the CLOSE * command is issued to close and sign off the terminal from all current sessions.
- The terminal ID is verified against the Terminal Directory. If the terminal is not authorized to PIE/CICS, the Logon Director invokes the DFHGMM CICS good morning program. Users can log on to CICS with DFHGMMCICS and work outside of the PIE/CICS environment.
- If the terminal is authorized to PIE/CICS, it checks for sign on data from a number of sources. (See “Receiving Sign On Data” on page 16.)
- If appropriate, it displays the screen specified in the Terminal Directory and waits for user input. Users can press PF3 to exit from the Logon Director screen and cancel the sign on request at any time.
- The Logon Director verifies the user ID, terminal node name, and password with the PIE/CICS User Directory. (Most customers use a generic password and node name in the User Directory and leave password screening entirely to the security system.)
- If screening ends with a match, Logon Director passes sign on data to the external security system for authorization.
- If sign on data does not match entries in the User Directory or external security, the Logon Director issues an error message that appears on the Logon Director screen. The user can attempt to sign on again with new data.
- The Logon Director forwards messages issued by an external security system. You determine if messages are sent to the CICS blank screen or the Display sign on Msg field of the User Directory. You determine how long messages are displayed with the sign on Message Display Time field of the Customization Options utility.
- The Logon Director sets the conditions of the initial PIE/CICS environment with values specified in the user’s profile after signed on is complete.

Many of the Logon Director’s default actions can be changed with the PCSTPPVI and PUXINIT exits. See “PCSTPPVI—Sign on Verification Exit” on page 139 and “PUXINIT—Post Sign on Initialization” on page 149 for details.
User Sign On Retry limit

If a sign on is rejected by PIE/CICS or the security system, the Logon Director screen is redisplayed with a message requesting the necessary data be re-entered. The number of permissible retries is limited to a maximum specified from the PIE Customization Options utility. PIE/CICS increments the count for both User Directory and external security with each unsuccessful log on attempt. The count is reset to 0 after a successful sign on.

If the log on retry limit is reached, PIE/CICS places the terminal in protected mode to prevent further input. When a terminal is protected, the Logon Director prevents any further sign on attempts. All other functions available from the Logon Director screen including news zoom, help, and terminal disconnect requests are also disabled.

The Userid field on the Logon Director screen is left unprotected. If the user accessed CICS through a VTAM session manager, a switch/escape string can be entered from this field to return to the VTAM session manager and attempt to sign on to a different region.

Authorized helpdesk personnel must reset the terminal using the PIE/CICS Protected Terminal List to return the terminal to a normal, unprotected state. See the “Protected Terminal List” on page 60 of the Operation and Administration Guide. The Protected Terminal list shows the time, date, VTAM node ID, and the user ID of the failed sign on attempts.

Use the PIE Customization Options utility and update the Maximum logon attempt count field to change the number of log on attempts. The default limit is set to 5. Set the Maximum Long attempt count field to 0 to disable terminals from being placed into protected mode.
Automating User Sign Ons

The Logon Director can accept sign on data from other applications. Also, it can send sign on data to other applications accessed through PIE/CICS products; MultiCICS, Dynamic Menus, NetGate, or NonStop. Multiple systems appear as a single, unified system to users with automated sign ons.

The Logon Director has various sign on options. The option you choose depends on how you want users to enter sign on data and what you want the Logon Director to do after receiving sign on data. The options are:

- Full interactive, standard user sign on at the Logon Director screen
  This is the standard method recommended for most sites. It is put into effect with the standard installation and customization recommended for MultiCICS, NetGate, and Dynamic Menus.

- Pass sign on data from another sign on application and bypass the Logon Director screen.
  This method allows you to continue using your site's current sign on application; a VTAM session manager or a customized site-specific sign on screen.

- Pre-defined sign on data at the Logon Director screen
  Pre-defined user IDs and passwords are useful for terminals that are physically secured, or terminals placed in public areas that can be accessed by customers.

- Default user IDs, bypassing the Logon Director screen
  This option is useful when you want PIE/CICS to be transparent to users. Every terminal eligible for PIE/CICS processing is automatically given a default user ID, and the Logon Director screen is completely bypassed. With default user IDs for all terminals, there can be no sign on errors, and no strange Logon Director screens to appear with unfamiliar error messages.

The first three options supply a real user ID and password to the Logon Director. The Logon Director can pass sign on data to VTAM applications accessed under PIE/CICS. If user IDs and passwords are authorized on these VTAM applications, the additional sign ons will be processed automatically and transparently, helping to create a single system image. If the Logon Director knows only a default user ID— as is the case with the last option— the sign on data it passes is rejected, and the user must manually log on to the new application.

The following sections explain each Logon Director sign on option in more detail.
Automating User Sign Ons

Signing On at the Logon Director Screen

The Logon Director screen provides a simple, error-tolerant interface to sign on users to PIE/CICS. The sign on process simultaneously logs users on to PIE/CICS and the external security system.

The Logon Director consists of a full screen with fields that are completed by the user as part of a sign on request. The Logon Director screen also includes a window to display announcements or other news.

You can customize the appearance of the Logon Director screen. You can create multiple screens and assign them to different terminals. The figure below shows the default Logon Director screen shipped with PIE/CICS:

![Logon Director Screen Example]

Logon Director News Service

In the example of the Logon Director screen shown above, a news service window appears near the bottom of the screen. Topical news and announcements can be placed in the window for users to read when they sign on to PIE/CICS. The Logon Director news service is a convenient way to inform users about recent changes to your site’s system.

News bulletins are created and updated with the PIE Text utility. You can include PIE variables in your news members to set the time and date. Variables are updated with current information when the news member is displayed. Refer to “Text Utility” on page 67 of the PIE/CICS Operation and Administration Guide for more information.

The first four lines of a news member appear in the window when a Logon Director screen is displayed. You may want to use these lines to display headlines or titles that summarize the contents of the news. Users must press PF2 to display the rest of the news member.

News members are assigned to log on screens in the Logon Panel utility. Multiple log on screens can use the same news member.
A news display is optional and is set up or changed by using the Logon Panel utility. If the news is not displayed in the news window, the user can still press PF2 to browse the news member.

**Using the Logon Director as the Good Morning Message**

If you use the CICS VTAM good morning message program, you can display the Logon Director screen as the good morning message on all PIE terminals. The Logon Director screen appears when a terminal is turned on or connects to CICS.

Set the Logon Director program or transaction as the good morning message. If the terminal is not in the Terminal Directory, or its entry specifies PIE=NO, USE, or REQ, the Logon Director transfers control to the standard DFHGMM CICS good morning message program. (You can change that action with the PCSTPPVI exit. See "PCSTPPVI—Sign on Verification Exit" on page 139, for details.

Complete one of the following procedures to set the Logon Director as your good morning transaction:

- **Automatic support**
  Go to the Customization options screens and specify YES for Automatic PIE GMM Tran. See page 65, “Customization Options”. PIE/CICS automatically changes the CICS good morning transaction to PSGM.
  When PIE/CICS is operational, the Logon Director becomes the good morning transaction. Otherwise, CICS uses its current good morning transaction.

- **Manually replace the good morning transaction with PSGM**:
  Replace the GMTRAN=CSGM parameter in your CICS DFHSIT table with GMTRAN=PSGM.
  Define the CSGM program to be the Logon Director program:
  Issue the following CEDA command:
  
  ```
  CEDA DEFINE TRANS(CSGM) PROG(PCTSPGMM) GR(groupname)
  ```

  Define this transaction in a group (groupname) that is activated after group DFHVTAM at CICS start-up.
  You should receive a warning message “W—TRANSACTIONS NAMES BEGINNING WITH ‘C’ ARE RESERVED AND MAY BE REDEFINED BY CICS.” This message can be ignored.

**Receiving sign on Data From Another Application**

The Logon Director can receive sign on data automatically from VTAM session managers, menu systems, or customized site sign on applications. Sign on data includes user IDs, passwords, profiles, and the initial command that executes after signing on to PIE/CICS.
Customized Sign On Applications

When users sign on with a locally developed sign on application, you can pass sign on data to the Logon Director using the Logon Director API.

Complete the following procedure to code the API. You can use sample code found in the sign on member of the PIE/CICS SAMPLIB dataset.

1. **Edit your program.**
2. **Following your program’s sign on procedures, add code to write the user ID, password, and optionally the user profile or an initial PIE command to a temporary storage queue.**

   The name of temporary storage queue should be @YZLterm, where @YZ is the PIE/CICS temporary storage prefix and term is the CICS terminal identifier. If you changed the PIE/CICS temporary storage prefix on the customization options screen, use the new prefix.

   The record should be item 1 of the queue and should be formatted using one of the accepted formats for VTAM log on messages. See “PCTSPDCX—Logon Message Exit” on page 140, for accepted formats.

3. **Transfer control to the Logon Director.**

   You can do this with a CICS XCTL to program PCTSPGMM.

   The temporary storage queue will be read the next time program PCTSPGMM is executed at the terminal. This means the queue could be created when the user is currently logged on to PIE/CICS. When the EXIT SIGNOFF command is issued, the current session will be signed off and control given to PCTSPGMM, which will then read the queue and attempt sign on.

   The TS queue is deleted after it is read.

VTAM Session Managers and Menu Systems

When the user signs on through a VTAM session manager or menu system (even from the VTAM logo screen), sign on data is passed to the requested VTAM application using the VTAM log on message.

If the Logon Director does not receive sign on information from a source with higher priority, the VTAM log on message is read and interpreted. If the sign on data is valid, the Logon Director bypasses the Logon Director screen and places the user at their initial screen defined by the profile that was passed in the log on message. If the sign on data is rejected, the Logon Director screen is displayed and the user is asked to enter sign on data again.

You may have to do some customization to receive sign on data through the VTAM log on message in VTAM applications that are not running PIE/CICS. For more information and customization procedures, see “Receiving a VTAM log on Message” on page 10.
Other CICS Regions

Users can switch to another VTAM application using the PASS, TSWITCH, or ACCESS commands. When a switch occurs, PIE/CICS passes sign on data from the originating region to the target system using the VTAM log on message.

If the target system is another PIE/CICS region, its Logon Director reads and interprets the incoming VTAM log on message and logs the user on automatically. Signing on by using data taken from a VTAM log on message works just like receiving log on information from a VTAM session manager or menu system.

Receiving a VTAM log on Message

Receiving sign on data from a VTAM log on message may require some customization. The following conditions must be met before the Logon Director can receive and interpret data from a VTAM log on message:

- The Logon Director must not have received sign on data from a source with a higher priority. See “Receiving Sign On Data” on page 16.
- The SIT must have LOGNMSG=YES specified in the CICS receiving region.
- The receiving region’s PIE Options file must specify YES for the Extract VTAM Logonmsg by Logon Dir field. See page 65, “Customization Options”.

If these conditions have been specified, the Logon Director looks for a VTAM log on message automatically. If a message is present, the Logon Director parses the contents of the message into user ID, password, profile, and PIE command.

As shipped, the Logon Director recognizes several different message formats. If it does not recognize the format used by your VTAM system, you can specify the format with the Logon Message Analysis exit. See “PCTSPDCX—Logon Message Exit” on page 140, for more information.

The Logon Message exit is invoked after a message has been parsed. You can use this exit to change fields and the action taken by the Logon Director. For example, the data following the password is usually regarded as a profile name. But the Logon Message exit can treat it as a PIE command and save it in a special temporary storage queue to be executed later by PIE/CICS or your own application.

For more information on the Logon Message exit, see “PCTSPDCX—Logon Message Exit” on page 140. To pass a PIE command with a log on message, see “VTAM Applications” on page 109.

If you switch to another PIE/CICS region, you can pass encrypted passwords in the VTAM log on message. Encryption works only when CPUs are synchronized to Greenwich time.
Pre-defined sign on Data to Specific Terminals

You can pre-define user IDs and passwords to specific terminals. The user sees a Logon Director screen with the user ID, and possibly the password already entered on the screen. The user must only press ENTER to complete the sign on process.

Pre-defining sign on data is useful for terminals that are physically secured. An assumption is made that anybody with physical access to the terminal also has the authority to use PIE/CICS. A dedicated PIE/CICS terminal within a data center control room is a typical example where pre-defining the Logon Director screen with a user ID and password can save time.

Pre-defining sign on data is also useful for terminals in public areas. For example, terminals may be placed in a public area to allow customers to browse through an online product catalog or to check current prices.

When a terminal connects to PIE/CICS, the log on screen appears with pre-defined data appearing on the appropriate screen fields. If the password field has been pre-defined, it will be present, but dark. Users still have the option of entering a different user ID/password pair from the Logon Director screen.

Perform the following procedure to prepare a terminal with pre-defined sign on screen:

1. **Check the User Directory to find an entry with the appropriate profile and PIE/CICS authority for the terminals to have pre-defined sign ons.**
   
   If there are no appropriate entries, create one. (See “Creating a New User Directory Entry” on page 60.)

2. **Go to the Terminal Directory.**

3. **Add entries for the terminals with pre-defined sign on data.**
   
   See “Creating a New Terminal Entry” on page 51. For each terminal entry, specify the user ID and password combinations that are acceptable to your site's security system in the Userid and Password fields.
   
   Specify a Logon Director screen name in the Logon field. The default screen name is PIELMAIN. You may want to create a custom screen with messages that indicate to the user that some fields contain pre-defined data.

4. **Make the Logon Director the good morning transaction.**
   
   See “Using the Logon Director as the Good Morning Message” on page 8, for more information.
   
   This Logon Director screen appears with completed fields when the terminal accesses CICS.

You can completely bypass the Logon Director screen if you want PIE/CICS to be transparent to users. To do so, see “Pre-defined Sign On Data to Bypass the Logon Director Screen” on page 12.
Pre-defined Sign On Data to Bypass the Logon Director Screen

Pre-defined user IDs and passwords can be specified to completely bypass the Logon Director screen when you want PIE/CICS to be transparent to users. This method is similar to the previous section, “Pre-defined sign on Data to Specific Terminals” on page 11. In the previous section, the terminals still display the Logon Director screen with pre-defined fields. With this method, the PIE sign on process is completely automatic and transparent to users. The Logon Director screen is never displayed. A terminal is signed on to PIE/CICS as soon as it accesses CICS.

If you access another VTAM application from PIE/CICS, the user's ID and password can be passed to the VTAM application. However, if you use a default user ID and password, PIE/CICS passes default user IDs and passwords; never the user's actual ID and password. The default sign on data will be rejected by the VTAM application. In that case, users must manually log on to the new application.
Part 1: Specifying Terminals to Bypass the Logon Director Screen

Perform the following procedure to identify the terminals that bypass the Logon Director screen:

1. Decide how to assign system administrator authority in PIE/CICS.
   
   You must have a system administrator authority to update PIE/CICS in the future. To determine this, decide whether all terminals will sign on to PIE/CICS using default user IDs.

   If not, then you can grant system administrator authority to a user who signs on to PIE/CICS from a terminal that does not use a default ID. That user will sign on using his actual ID, and you can give that ID system administrator authority. The remaining terminals can sign on with user IDs that have no authority.

   If all terminals will sign on using a default ID, choose one of the following options:
   
   - Use the same user ID for all terminals and give that user ID system administrator authority. All PIE users are assigned as system administrators. But, you can protect system administrator functions by protecting group 3 PIE/CICS transactions listed in “Securing PIE/CICS Transactions” on page 80.
   
   - Use different user IDs for different terminals. Then, limit the system administrator user ID to certain terminals.

   The special ID PIEUSER can help you do this. When you specify PIEUSER in the Userid field of the Terminal Directory, PIE/CICS will sign the terminal on with the user ID PIEUterm, where term is the CICS terminal ID. In the User Directory, an entry for the user ID PIEU* will handle all terminals.

   You can use the PIEUSER/PIEUterm feature to create different environments for different terminals. For example, if a system administrator uses the CICS terminal L001, you can create a system administrator entry with the user ID PIEUL001. If all your public terminal IDs start with the letter P, you could create a special User Directory entry for those terminals with the user ID PIEUP*.

2. Edit the User Directory according to the decisions you made above.

   In all cases, you must create an entry for a system administrator (specify Y in the System Administrator field). If your PIE/CICS system is new, you will have two entries with * as the user ID. The first is a system administrator entry. Specify an appropriate user ID in this entry.

   If you will have user IDs without system administrator authority, create entries for them (specify N in the System Administrator field of the entry). If your PIE/CICS system is new, the second * entry is for a standard user.

   For every user ID that will be used to sign terminals on to PIE/CICS automatically:
   
   - Type a specific, non-generic user ID in the Userid field or type the value PIEU*.
   - Enter an asterisk (*) in the Password field.
   - Specify a profile in the Profile field—you will be editing the profile later in this procedure.

   See “Editing a User Directory Entry” on page 54, for more information.

4. **Add entries for the terminals to have pre-defined sign ons. You may use wildcards.**

   For each terminal entry, specify one of the user IDs selected in step 1. If you used the PIEU* generic, specify PIEUSER as the user ID.

   You may specify a value for the Password field.

   For each of these terminal entries, specify (NONE) in the Panel field. PIE/CICS will bypass the Logon Director screen.

   If all terminals will be signed on to PIE/CICS automatically, ensure that at least one terminal signs on with a user ID that has system administrator authority.

   See “Editing a Terminal Directory Entry” on page 47, for more information.

5. **Make the Logon Director the good morning transaction.**

   See “Using the Logon Director as the Good Morning Message” on page 8, for more information.
Part 2: Setting Up CICS Sign on

Normally, the Logon Director signs on users to external security and CICS together as part of the overall sign on process. However, the default PIE/CICS user IDs and passwords are probably not defined to external security. In this case, the Logon Director’s automatic sign on cannot work.

You need to set up PIE/CICS to bypass the automatic sign on and transfer control to the existing sign on screen. Users can sign on from this screen as they have done in the past. You want PIE/CICS to pass control to either the CICS blank screen or the sign on screen, depending on the current configuration of your system.

1. **Access the Profile utility and edit the profiles specified in the User Directory entries you edited in Part 1.**
   
   See “Editing a Profile” on page 34, for more information on editing a profile.

2. **Specify OPEN 1 in the Initial PIE Command field of the main Profile screen.**
   
   Session 1 becomes the first open session after PIE/CICS signs on the terminal.

3. **Transfer to the Profile Sessions screen and enter one of the following on the Execute field of session 1:**
   - **Enter SYSTEM if the CSSN transaction is used to access the sign on screen.**
     
     Session 1 opens the CICS blank screen (system mode). When users turn on their terminal (or access CICS), the CICS blank screen appears on the screen. They can then enter the CSSN transaction to complete their CICS sign on.
   - **Enter CSSN if your sign on screen appears automatically.**
     
     Session 1 executes the CSSN transaction when users turn on the terminal (or access CICS).

4. **Modify the ATTACHX table to bypass the CSSN, CSSF, CESN, and CESF transactions.**
   
   You can edit the standard ATTACHX table or simply reassemble the table with the sample transparent ATTACHN table. To use the sample, edit member PG@RPCSC of the SAMPLIB dataset. Change the statement—
   
   COPY ATTACHX TO COPY ATTACHN

5. **Re-assemble the ATTACHX table using member PG@RPCSC of the PIE CNTL dataset.**
   
   If you used the PIEUSER feature (PIEUSER in the Terminal Directory and PIEU* in the User Directory) to make sign on transparent, PIE/CICS automatically bypasses security checking for those terminals. It bypasses the security check for user IDs that begin with PIEU and that are established from the Terminal Directory. All other terminals are processed with external security.
   
   However, if you used any other user IDs to make sign on transparent, you must manually turn off the security interface. This affects all terminals, whether they are signed on transparently or not.

6. **Edit members PCSTPPVI, PCTTPUX1, and PC@XPSEC of the CNTL dataset to turn off external security checking.**
   
   In each member, specify NONE for the SECURTY parameter.

7. **Run the jobs.**
Receiving Sign On Data

The Logon Director accepts sign on data from the following sources:

- Logon Director screen
- Terminal Directory entry for the terminal in use
- VTAM log on message
- Logon Director API

The Logon Director searches for sign on data from these sources in the sequence shown in the following list:

- If PIE=USE is specified in the Terminal Directory entry for the terminal submitting a log on request, the Logon Director accepts the current user ID as both user ID and password. If the terminal is not currently signed on, the Logon Director screen appears instead because no current user ID is assigned to the terminal.
- If PIE=USE is not specified, the Logon Director attempts to read the temporary storage record created with the Logon Director API.
- If the API temporary storage record does not exist, the Logon Director attempts to extract sign on data from the VTAM log on message.
- If there is no VTAM log on message, the Logon Director examines the Terminal Directory for a preset (default) user ID and password.
- If log on data is not present from any source, the Logon Director screen appears and users must enter the requisite data.
Sign on Message Display Option

After signing on, the screen is cleared and messages are displayed before executing the initial application. External security messages are displayed first followed by the Logon Director message. The Logon Director message is in the format:

userid SIGNED ON AT TERMINAL nodename AT hh:mm:ss, mmm dd.

You can turn off the sign on message display for specific users. Update the Display sign on Msg field in User Directory entries.
Chapter 2  Logon Director

Customizing the Logon Director Screen

Logon Director screens are created, customized, and maintained using the following screens of the Logon Panel utility:

- Panel List: This screen lists log on screens that can be edited.
- Panel Definition: This screen can be used to change a screen name, title, administrative rules, and comments. Also, it can be used to copy or delete an existing screen.
- Screen Editor: This screen provides editing functions to change the appearance of a log on screen.

You can add new log on screens by copying and editing the $$PIEPAN sample screen shipped with PIE/CICS. You cannot create an entirely new screen or copy data from an external source to a Logon Director screen.

Authorization to Customize a Logon Director Screen

Logon Director screens are stored in the Repository as PIE objects with the SYSTEM group name. Therefore, only system administrators have the authority to create new screens. However, after a screen is created, it can be edited by any user whose ID matches the userid entered on the screen’s Administrator field. Any user has the authority to view the screen whose ID matches the name placed in the Auditor field.

If you are a system administrator, you may want to delegate the task of customizing log on screens to a group administrator. First, create the screens to be administered by group administrators. Complete the Administrator and Auditor fields with the group administrator’s group name. By default, the $$PIEPAN sample screen has an * wildcard in these fields that authorizes all users to edit the screens. See “Editing a Log On Screen” on page 21, and “Creating a Log On Screen” on page 26, for more information.
Format of the Logon Director Screen

There are a number of distinct fields within a Logon Screen. These fields are located within predefined areas based upon the row and column position of a standard 3270 screen (24 rows by 80 columns).

Each field can accept data from the following categories:

- Free area for the screen title, logo, instructions, and other text
- Sign on prompts
- News area and border
- Reserved area for sign on input fields and messages

You may modify the text in all fields except the input and message fields.

You may want to create a help line for PF keys in the free area. See “Logon Director Commands and PF Keys” on page 13 of the Operation and Administration Guide, for these keys.

You can use PIE variables in your log on screen. The &NEWS and &INFO variables are particularly helpful. Placing &NEWS in the news area displays the first four lines of your news bulletin. &INFO writes the:

- Day of the week, abbreviated (Tue, Wed, etc.)
- Month and day, abbreviated (Jan 12)
- Month, day, and year as mm/dd/yy
- Time of day as hh:mm:ss
- Terminal ID as term=cics-id/vtam-nodename
- CICS applid as sys=applid
- CPU SMF identifier as cpu=xxxxxx

The following table shows the fields of the Logon Director screen based upon the row and column position of a 3270 screen.

<table>
<thead>
<tr>
<th>Row Number</th>
<th>Column Number</th>
<th>Field Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 to 14</td>
<td>All</td>
<td>Free area</td>
</tr>
<tr>
<td>15 to 16</td>
<td>1 to 10</td>
<td>Free area</td>
</tr>
<tr>
<td></td>
<td>71 to 80</td>
<td></td>
</tr>
<tr>
<td>15 to 16</td>
<td>12 to 19</td>
<td>sign on prompts</td>
</tr>
<tr>
<td></td>
<td>21 to 24</td>
<td></td>
</tr>
<tr>
<td></td>
<td>47 to 54</td>
<td></td>
</tr>
<tr>
<td></td>
<td>56 to 59</td>
<td></td>
</tr>
<tr>
<td>15 to 16</td>
<td>All other columns</td>
<td>Reserved</td>
</tr>
<tr>
<td>17</td>
<td>All</td>
<td>Message display, reserved</td>
</tr>
<tr>
<td>18</td>
<td>All</td>
<td>News box border</td>
</tr>
<tr>
<td>19 to 22</td>
<td>1 and 80</td>
<td>News area</td>
</tr>
<tr>
<td>23</td>
<td>All</td>
<td></td>
</tr>
<tr>
<td>19 to 22</td>
<td>2 to 79</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>All</td>
<td>Free area</td>
</tr>
</tbody>
</table>

Line 15 contains two input fields: the first is for the user ID and the second is for the profile name. Line 16 also contains two input fields: the first for password, the second for a new password.
Each input field has a pair of prompt fields immediately to the left. The first field in the pair is eight characters long and usually contains the field name, such as User ID or Password. Text within the field is displayed in normal intensity. The second prompt field is four characters long and usually points to the input field, with ===> or something similar. Text appearing within the second field is displayed in high intensity.

You can make an input field autoskip. Simply blank out the second prompt for the field. You may want to use this feature for login panels that are automatically filled in by the Logon Director. (See "Pre-defined sign on Data to Specific Terminals" on page 11.)

Messages appear on line 17 and are centered on the screen. If the message is more than one line long, the remaining portion of the message continues on the news field of the screen.

Displaying news is optional. To display news, type the variable &NEWS as the first character string in the box. Otherwise, you may type whatever you like in this area. If the news is not displayed in the news window, the user can still press PF2 to browse the news member.
Customizing the Logon Director Screen

Editing a Log On Screen
You should begin customizing log on screens by editing a copy of the $$PIEPAN sample screen distributed with PIE/CICS. To edit this sample or another screen, complete the following procedure.

1. **Display the Logon Panel utility by one of the following methods:**
   - If you are using the ACCOUNT profile, select session 1 from the Sessions menu to access the PIE master menu. Select option 1, the System Administrator menu. Then select option 4, LOGON.
   - Access the PIE/CICS System Administrator menu using the transaction PEXE MENU PIEMADM. Select option 4, LOGON.
   - Enter: ==LIST PANEL [groupid] from any session.

   Use groupid to specify the screens appearing on the Panel List screen by Group ID. If you omit the groupid parameter, only screens with your Group ID are listed. Specify * to list all Group IDs.

   An example of the Panel List screen is shown in the following figure.

   2. **Select the screen you want to edit from the Panel List by entering the S line command on the Sel column to the left of the screen name.**
Chapter 2  Logon Director

The sample screen distributed with PIE/CICS is named SYSTEM.$$PIEPAN. The Panel Definition screen shown in the following figure appears after selecting a screen from the Panel List screen.

PIE Utility (Logon Panel) ----------------------------------- (C) 1995 TSC, Inc
Command ===> Update History:
Panel Name ===> $$PIEPAN                        Name: SYS1
News Bulletin ===> E                            Term: TERM8E37
Panel Title ===> Model Logon Panel               Date: 07/20/96
Administrator ===> *                        Time: 9:08:29
Auditor ===> *                                      Terminal ===> *
Terminal ===> *
Generic forms, ?, % and * characters are accepted above.
Comments ===> Comments ===> Enter Edit command to edit panel.

3. Change any of the following fields of the Panel Definition screen.

Panel name  Member name of the screen. If you are editing an existing screen, leave this field unchanged. However, specify another name to create a new screen.

News Bulletin  Name of the text member to use as the news file for this screen. The group name of the text member must be SYSTEM.

Panel Title  Short comment to describe this screen. Comments appear in the Title field next to the screen name on the Panel List.

Administrator  User ID mask that identifies users who can edit this screen.

Auditor  User ID mask that identifies users who can browse this screen with the Logon Panel utility.

Terminal  Node name mask that identifies terminals from which the screen can be edited or browsed.

Comments  Comments about this screen.

See “Panel Definition Screen Commands and PF Keys” on page 25, for information on Panel Definition PF keys and commands.
4. **Press PF2 to edit the selected screen.**

The following figure shows an example of the $PIEPAN screen that appears after being selected for editing. The editing screen allows you to see the screen as it appears to users when they log on to PIE/CICS. You change the appearance of a log on screen by simply entering or deleting text from the editing screen.

![Example of the $PIEPAN Screen](image)

5. **Change text and spacing on most lines.**

You can change the prompt fields for user input, but you cannot change the field positions. You can change the border for the news box, but you cannot change its position.

You may use PIE variables on a log on screen. When you place the special variable \&INFO at the first column of a line, PIE/CICS displays a full line of system information, including the date, time, and terminal ID. If you want current news displayed in the news box, place the special variable \&NEWS at the beginning of the news box. You can omit the news box by removing the border and the \&NEWS variable.

For more information on log on screen layout, see “Format of the Logon Director Screen” on page 19.

If your escape string is == (the default) and you edit a field that contains == (such as an arrow prompt field), PIE will execute that field as the escape string when you press ENTER, PF3, or PF4. The Switch pop-up window appears and you cannot save your changes and exit from the screen. If this occurs, go to your run-time profile and temporarily change or remove your escape string. See “Changing User Profiles” on page 18 of the Operation and Administration Guide, for more information.

To undo changes you have made, press CLEAR (Reshow).
6. **Press PF3 (End) or PF4 (Return) to save your editing changes and exit from the editing screen.**
   
   You should return to the Panel Definition screen.

7. **Press PF3 to exit this screen and save the changes.**
   
   (PF4 also saves your changes.) To exit the screen without saving the changes on it, enter CANCEL.

8. **Press PF3 to exit from the Panel List screen.**

   Changes to the Logon Director screen become effective immediately after they are saved. If a user presses ENTER at a changed screen, the new screen is displayed.
Maintaining the Logon Director Screen

The remainder of this chapter describes procedures to maintain Logon Director screens. In most cases, maintenance procedures are completed from the Panel List screen.

Panel Definition Screen Commands and PF Keys

There are commands and PF keys associated with each of the three screens of the Logon Panel utility. Most maintenance procedures require you to enter a line command from the Panel List screen to select the screen and complete the indicated maintenance action.

Panel List Commands

For the Panel List screen commands and PF keys, see “Logon Director Commands and PF Keys” on page 13 in the Operation and Administration Guide.

The Panel List screen supports the following line commands.

- **C** Copy
- **D** Delete
- **E** Edit
- **R** Rename
- **S** Select

**Other Profile Screens**

The Panel Definition screen accepts the following commands. Enter the command at the command line or press the associated PF key.

- **HELP (PF1)** Online help.
- **EDIT (PF2)** Go to the Screen Editor to change the screen's appearance.
- **RESHOW (CLEAR)** Discard unsaved changes and refresh the screen.
- **CANCEL** Discard unsaved changes and return to the Panel List screen.
- **UNDO** Discard unsaved changes and remain in this screen.
- **END (PF3)** Save changes on this screen and return to the Panel List screen.
- **RETURN (PF4)** Save changes on this screen, terminate the Logon Panel utility, and return to the highest level—the Sessions menu, the last “main” menu, or end of session.
- **SAVE (PF5)** Save changes without leaving the screen.
  - For all save operations: If you are a system administrator and you have changed the name in the Panel Name field to a new name, a new screen will be created.
  - If you have changed the name to one that already exists, a message window will ask you to confirm whether you want to replace the old screen with the new.

- **RETRIEVE (PF6)** Retrieve a screen by name. Retrieving a screen is just like selecting it from the Panel List.
  - The RETRIEVE command replaces what is currently on the screen and does not save any changes. Issue SAVE before RETRIEVE to save your changes.
PREVIOUS (PF10) Retrieves the screen just before the current screen on the Panel List.
NEXT (PF11) Retrieves the screen just after the current screen on the Panel List.
Use PREVIOUS and NEXT command to scan for a screen when you don’t know its name. You may specify a generic screen name. PREVIOUS and NEXT will use that name as a starting point, and retrieve the first screen that matches or the next screen from that point.
DELETE (PF9) Delete the current screen from the Repository.

Creating a Log On Screen
Only system administrators have the authority to copy and edit existing screens to create new log on screens. The following procedure summarizes the major steps to create a new log on screen. “Customizing the Logon Director Screen,” beginning on page 18 gives a more thorough explanation of the steps.

1. Invoke the Logon Panel utility, as described in “Editing a Log On Screen” on page 21.
The Panel List screen appears with a list of current log on screens.
2. Press PF2 (add) to create a copy of the first screen appearing on the Panel List screen. Or, enter C on the command field next to an existing screen that you want to use as a model for your new screen.
3. Enter a new screen name in the message box that appears.
   If there is a current screen with that Group ID and name, a pop-up window appears with a message informing you the screen you are copying will replace the existing screen. Press ENTER to replace the screen or PF3 to cancel the copy request.
4. Press PF6 to refresh the screen.
5. Now you can select the screen from the list and continue with the procedures in “Editing a Log On Screen.”

Deleting a Log On Screen
Complete the following procedure to delete a log on screen.

1. Invoke the Logon Panel utility, as described in “Editing a Log On Screen” on page 21.
The Panel List appears with a list of all existing log on screens.
2. Enter D in the command field next to the name of the screen to be deleted.
3. Press ENTER to confirm the delete.
   You can press PF3 to keep the screen and cancel the delete request.
4. Press PF6 to refresh the screen and verify the screen has been deleted.
Renaming a Log On Screen
Complete the following steps to rename a screen:

1. **Invoke the Logon Panel utility, as described in “Editing a Log On Screen” on page 21.**
2. Enter R in the command field next to the screen to be renamed.
3. Enter the new screen name in the message box that appears.
   - If there is already a screen with that Group ID and name, a message window appears indicating the screen you are copying will replace the existing screen. Press ENTER to replace the screen or PF3 to abort.
4. **Press PF6 to refresh the screen and confirm the screen has been renamed.**

Assigning Log On Screens
Log on screens are assigned to terminals through the Terminal Directory. Specify the name of the log on screen on the Logon field of the Terminal Directory entry.

Editing Log On Screen Help Members
PIETLDIR is the standard help member that provides online help for all PIE/CICS log on screens. PIETLDIR is shipped with PIE/CICS. You can edit PIETLDIR or replace it with a custom member created at your site.

Use the PIE Text utility to edit PIETLDIR or create a new member. See page 67, “Text Utility”. of the Operation and Administration Guide for instructions to edit text members. The help member must be a PIE text object. If you created a help text member using another utility, you can load the object using the Repository Load utility. See page 77, “Repository Load Utility”. of the Operation and Administration Guide.

You can rename a help text member to store multiple members in the Repository. Access the Customization Options screen and change the Name of Logon Director Help field to the name of the member that you want to use for your log on screens. See “Using the Customization Options Utility” on page 66.
Chapter 3 Profiles

Profiles set the conditions of a user's PIE/CICS session. Each user’s PIE/CICS run-time environment is controlled by options set in their personal profiles. A user profile defines the following:

- Initial session configuration
- Default application for new sessions
- Initial command that executes after sign on is complete
- Final command before exiting from PIE/CICS
- PF key assignments
- Default characteristics of PIE/CICS screens

This chapter describes how to create and maintain PIE/CICS user profiles.
Assigning User Profiles

Default profiles are assigned to users when they sign on to PIE/CICS. You can also request a specific profile by entering the name of the profile from the Profile field of the Logon Director screen.

Profiles are listed in the User Directory. Profiles are PIE objects that can have unique Group IDs; a single profile name can have several matching profiles. Consequently, users can select from a variety of profiles. PIE/CICS assigns user profiles based on the following criteria:

- If a user specifies a profile from the Logon Director screen, that profile is used for the duration the user is signed on to PIE/CICS.
- If no profile is specified at sign on, PIE/CICS uses the profile specified as the default in the User Directory.
- PIE/CICS determines the Group ID based upon the results of a hierarchical search. First, it looks for a profile member with the user’s ID as its Group ID, then the user’s group as its Group ID, and finally SYSTEM as the Group ID.
- If a matching profile is not found, PIE/CICS assigns a default user profile with the following screen and key definitions:

  ```
  == PIE escape string
  ; Command delimiter
  YES Titles display
  YES Message ID
  PF3 and PF15 System end keys
  CURSOR Scroll Default
  ```

  The session configuration, global keys, initial command, default application, and final command fields are not specified in the default user profile.

  After sign on is complete, users are placed at the Sessions menu with the message:

  `Requested Profile group.member not found.`

Private and Run-Time Profiles

When PIE/CICS selects a profile for a user, it loads the profile into temporary storage. The profile becomes the user’s run-time profile. Users can change their run-time profile with the PROFILE command. These changes remain in effect until the user signs off PIE/CICS.

You can limit access to the PROFILE command by restricting access to class 3 PIE/CICS commands, which is the command class that PROFILE belongs. Other commands belong to Class 3. Before restricting access to the command class, refer to the PIE/CICS Command Reference for a list of commands that belong to class 3.

Private Profiles

When a profile’s Group ID is the user’s ID, the profile is a private profile. For example, user J SMITH is using profile ACCOUNT; his private profile is J SMITH.ACCOUNT. Only users with that user ID can use that profile—thus it is private.
**Private Object Authority**

Users with private object authority can create private objects. A user with private object authority can make changes to a public profile and save the changes permanently into a private profile. For example, suppose J SMITH is working with the profile named PAYROLL.ACCOUNT. If he changes this profile, using the PROFILE command, and he saves his changes, PIE/CICS will save his changes into a new private profile, called J SMITH.ACCOUNT.

All users can make and save changes to existing private objects—even if they do not have private object authority. Users who do not have private object authority simply cannot create private objects—the system administrator must create the private object for them.

For more information on private object authority, see the Private Objects field in “Editing a User Directory Entry,” beginning on page 54.

**Granting Private Object Authority**

Private object authority is granted through User Directory entries. Set the Private Objects field to either Y (Yes) or O (Only). See “Editing a User Directory Entry,” beginning on page 54 for more information.

**PROFILE Command**

Use of the PROFILE command can be controlled using PIE security classes. For more information on security classes, see the Security Classes field in “Editing a User Directory Entry,” beginning on page 54. For information on using the PROFILE command, see “Maintaining Profiles,” beginning on page 41.
Creating Profiles

System administrators have complete authority to create and maintain all profiles. Group administrators can create and maintain profiles whose Group IDs match the administrator’s group ID. Users with private object authority can create their own private profiles by modifying other profiles they are authorized to use. They can modify only their own private profiles.

Profiles are created and maintained with the Profile utility. The Profile utility consists of five screens:

- **Profile List**
  Lists existing profiles
- **Main screen**
  Contains control information, such as profile name and access masks. This screen also specifies the initial and final commands and default application. Other Profile utility screens are listed as options from this screen.
- **Terminal**
  Specifies some command information (escape string, command string delimiter, system end key) and terminal information.
- **Keys**
  Assigns commands or transactions to PF and PA keys
- **Sessions**
  Specifies the initial session configuration consisting of allocated sessions when you sign on to PIE/CICS and the applications assigned to run in each session.

When a user accesses the Profile utility with the PROFILE command, the Profile List is skipped. Otherwise, the screens are almost identical. When a user issues the PROFILE command, the profile they are currently using is selected. They can retrieve other profiles using the RETRIEVE command. Changes will take place in their run-time profile immediately.

Changes to a profile made by system or group administrators do not become effective for users who have already signed on with that profile. To pick up these changes, these users must sign on to PIE/CICS again.

Conditional Commands Defined in User Profiles

There are several commands defined in user profiles that are invoked automatically under certain conditions:

- **Initial PIE command**
  Executes immediately after the user signs on to PIE/CICS. Usually, this command opens a pre-defined session or sends the user to the Sessions menu.
- **Default Application**
  Executes when the user creates a session without specifying an application to run in the session. Also, the default application runs after certain PIE command failures.
- **Final PIE Command**
  Executes when the user ends a session and there are no more active sessions. The final PIE command determines the user’s destination after closing the last session.
Creating Profiles

Command Execution
There are several profile elements that affect how commands are processed by PIE. They are:

- **Global PIE Keys**
  Perform an assigned PIE function in all PIE/CICS sessions. These keys override normal application functions for the corresponding key. Typically, global PIE keys are used to navigate between sessions or return to the Sessions menu.
  
  To conserve temporary storage space, the global PIE key TSQ is not created unless there is at least one specified global key. The global PIE TSQ is variable length, containing only the keys and their assigned functions.

- **Escape String**
  Allows you to execute a PIE command from a session screen. If you enter the escape string alone, you will get a pop-up menu. If you enter it with a PIE command, PIE/CICS will execute that command.

- **Command Delimiters**
  Separate one command from another when multiple commands are entered as a string. You can specify up to seven different delimiter characters in a single profile.

- **System End Keys**
  The system end key returns to the next higher processing level, whether that is returning to a previous menu, ending the session, or exiting PIE/CICS. You can specify two system end keys in a profile. You might use a system end key when Auto End on is inactive and a blank screen appears after ending an application. You could also use it after going to the blank screen from a menu.

Creating a New Profile
To create a profile, perform the following.

1. **Invoke the Profile utility, as described in “Editing a Profile”**.
2. **Press PF2 (add) to create a new profile with default values; no fields are filled in.**
   Or, to use another profile as a model, enter C in the command field next to the profile. Enter the new profile name in the message box that appears on the screen.
   
   If there is already a profile with that Group ID and name, a message window appears informing you that the profile you are copying will replace the existing profile. Press ENTER to replace the profile or PF3 to cancel the copy request.

3. **Press PF6 to refresh the screen.**

4. **Now you can select the profile from the list and continue with the procedures in “Editing a Profile,” on page 34.**
Editing a Profile

There are two ways to use the Profile utility based upon your status within a PIE/CICS system.

- **Administrator mode**
  An administrator can create, edit, delete, and otherwise maintain profiles. In administrator mode, data can be modified from all input fields of profile screens.

- **End User mode**
  Users can edit their current profile and the changes become effective immediately. The changes remain in effect for the duration the user is logged on to PIE/CICS. Also, if users have PRIVATE=YES specified in the Users directory, they can create and retrieve personal profiles.

  In End User mode some screen fields cannot be modified. Only system or group administrators are authorized to modify these fields.

This section explains how to edit a profile from the perspective of a system or group administrator. You should begin customizing profiles by customizing a single profile that will serve as a template for all subsequent profiles. PIE/CICS is shipped with sample DEMO, ACCOUNT, and USER profiles.

The following procedure explains the major steps to edit a profile:

1. **Display the Profile utility by one of the following methods:**
   - If you are using the ACCOUNT profile, select session 1 from the Sessions menu. This will access the PIE master menu. Select option 1, the System Administrator menu. Then select option 8, PROFILE.
   - Access the PIE/CICS System Administrator menu using the transaction PEXE MENU PIEMADM. Select option 8, PROFILE.
   - From any session, enter `==LIST PROFILE` or just `==LIST`. Use groupid to select the Group IDs listed. You may use wildcards. If you omit groupid, all Group IDs will be listed.

   The following figure shows an example of the Profile List that should appear after being selected.
2. **Select the profile you want to edit by entering the `S` line command on the `Sel` column next to the profile name.**

The following figure shows an example of the Profile utility main screen that appears after a profile has been selected for editing. In this example, the ACCOUNT profile has been selected.

```
Profile (SYSTEM.ACCOUNT) -------------------------------- (C) 1995 TSC, Inc
Option ===> Update History:
  1 Terminal - Specify Terminal Characteristics Name: USR1
  2 Defaults - Specify defaults for PIE (reserved) Term: PIE10013
  3 Keys - Specify Global PIE keys Date: 6/19/1999
  4 Sessions - Specify Sessions Configuration Time: 11:59:11

PIE profile group ===> SYSTEM
PIE profile name ===> ACCOUNT
PIE profile title ===> System Administrator profile

Default Application ===> system prompt
Initial PIE command ===> open 1
Final PIE command ===> exit logoff

Administrator ===> * Terminal ===> *
Auditor ===> * Terminal ===> *
Generic forms, ?, % and * characters are accepted above.

PF: 1 Help 3 END 4 RETURN 5 SAVE 6 RETRIEVE 9 DELETE
```

3. **Change any of the following fields of the Profile utility main screen**

   **PIE profile group** Group name of the profile.
   **PIE profile name** Member name of the profile.
   **PIE profile title** Short comment to describe this profile. Comments appear on the `Title` field of the Profile List screen.

   **Default Application**
   Default application that becomes active in the event a session is opened without an assigned application. The default application also executes as the default PIE Application command.
   Usually, a command that executes a menu or a PIE/REXX program is entered on this field. The default is the SYSTEM command.
   The field expects a PIE Application command. To code an Environment command, begin the string with `EC`.

   **Initial PIE command**
   First PIE command that executes after sign on is complete. You cannot stack commands: only one command can be entered on this field.
   The most common use of the initial command is to open a session. For example, the command `OPEN 1` places the user in session 1 immediately after signing on to PIE/CICS.
   This field expects a PIE Environment command. If you specify a PIE Application command instead, (like `MENU XYZ` to execute menu `XYZ`), the command will execute if there is a free session. It is usually
preferable to assign an Application command to a session and set OPEN or SWITCH as the initial PIE commands to open that session. You must not allocate all available sessions on the Sessions screen. ("Available sessions" is defined by the Maximum Sessions field in the User Directory.) If all sessions are allocated, a message window appears with the message “No available sessions” after sign on is complete. The user can switch to an allocated session from the window. See page 35, “Case studies”. in the MultiCICS Administration Guide and “Case Studies,” on page 83 of the Dynamic Menus Administration Guide for examples of the initial command.

**Final PIE command**

Command that executes when the last active session is closed. This command determines whether a PIE pop-up menu, the Logon Director screen, the VTAM prompt, or Session screen appears when no more sessions are available.

The field expects a PIE Environment command, but you can specify any PIE command. Typically, the command EXIT LOGOFF is used to automatically log users off when they close their last session. Another variation of the final PIE command is simply EXIT. This causes an exit pop-up menu to appear, allowing users to select a number of exit options from the menu.

The Customization Options field Action on session close determines whether “eligible sessions” includes the Sessions menu. If the Sessions menu is included, and the user has accessed the Sessions menu, the final PIE command will not execute unless the user ends the Sessions menu. Users can exit from PIE without ending the Sessions menu by entering the EXIT command from the Sessions menu.

The following fields are displayed only is the user has system or group administrator authority.

- **Administrator** User ID mask that identifies users who can edit profiles.
- **Auditor** Node name mask that identifies users who can browse this profile.
- **Terminal** Node name mask that identifies terminals from which the profile can be browsed and edited.

After completing the fields of the Profile utility main screen, other profile screens can be edited.

4. **Save your changes to the Profile utility main screen.**

Enter CANCEL instead to exit the Profile utility main screen without saving editing changes.

5. **Enter 1, 3, or 4 from the command line of the Profile utility main screen.**

These options select the remaining three profile screens. The screens can be edited in any order.
The following figure shows an example of the Terminal Characteristics screen that appears after selecting option 1 from the Profile utility main screen.

<table>
<thead>
<tr>
<th>PIE Profile Terminal Characteristics ------------------------------------------------- (C) 1995 TSC, Inc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command ====&gt;</td>
</tr>
<tr>
<td>Pie Escape String ====&gt; == (1-5 char. start at 2nd position)</td>
</tr>
<tr>
<td>Command Delimiters ====&gt; ; (Characters for command stacking)</td>
</tr>
<tr>
<td>Titles display ====&gt; YES (Yes - display titles)</td>
</tr>
<tr>
<td>Message ID ====&gt; NO (Yes - display message ID)</td>
</tr>
<tr>
<td>SYSTEM END Key ====&gt; PF3 (PF01-PF24, PA01-PA03, CLEAR)</td>
</tr>
<tr>
<td>SYSTEM END Key ====&gt; PF15 (PF01-PF24, PA01-PA03, CLEAR)</td>
</tr>
<tr>
<td>Scroll Default ====&gt; CURSOR (Page, Data, Half, Cursor are valid)</td>
</tr>
</tbody>
</table>

INSTRUCTIONS:

Press ENTER key to update terminal options
Enter END or RETURN command to update terminal options and exit

PF: 1 Help 3 END 4 RETURN 5 SAVE

You can change any of the following fields of the Terminal Characteristics screen:

**PIE escape string** (For MultiCICS users only) 1 to 5-character string that transfers control to PIE/CICS when entered from a session. An escape string can be used to switch to another session, produce a pop-up menu, or execute a PIE/CICS command. The default is ==. Leave a space before the string in the field.

**Command Delimiters**
Characters placed between commands in a string to separate them. Up to seven characters can be designated as command delimiters. The default is a semi colon (;). At least one delimiter is required.

**Titles Display**
Choice (Yes or No) to display field titles on line 3 of a screen. Specify N to suppress the display, Y to allow it. Users can change this option with the SET TITLE command.

**Message ID**
Choice (Yes or No) to display the message ID with the text of PIE/CICS error messages. The ID appears over the (C) TSC, Inc 19xx trademark at the upper right corner of the screen. Specify N to suppress the message ID, Y to allow it. Users can change this option with the SET SMSG command.

**SYSTEM END Key**
PF key, PA key, or CLEAR designated as the system end key. You may specify two end keys, one in each field. Read “Command Execution,” on page 33 for more information about the uses of a system end key.
Scroll Default  Length to scroll the screen forward or backward when using certain PIE/CICS utilities such as the Text Editor or the Terminal Directory Utility.

You can specify one of the following scroll lengths in this field.

- PAGE  Scroll one page.
- HALF  Scroll half a page.
- DATA  Scroll one full page minus one line.
- CURSOA  Scroll the length from the top of the screen to the current cursor position. The default.
- nn  Scroll nn screen lines.

6. **Enter 3 from the command line of the Profile utility main screen.**

The following figure shows an example of the Global PIE Keys screen that appears after selecting option 3 from the Profile utility main screen.

This screen assigns functions or commands to PA keys and all 24 PF keys. Users can change these assignments with the SET command.

- PIE Application commands run in a separate, free session. PIE Environment commands do not require a free session. You do not need to prefix Environment commands with EC or ENV.
- You can stack the commands to perform multiple functions with each key. Separate commands with a command delimiter. When you switch to another session, the command stack is ended. Create a PIE/REXX program and assign it to a key to continue command processing when a switch is made to another session.
- Cut and paste commands should be assigned to global PIE keys. You can set up standard cut and paste or enhanced cut and paste with PIE/REXX programs. See “Cut and Paste Commands,” on page 28 of the Operation and Administration Guide and the PIE/REXX Manual for details on the cut and paste commands and sample PIE/REXX programs.
Global PIE keys override application keys. Because global keys operate in each PIE/CICS session, do not assign PIE/CICS functions to keys that are normally by applications.

There are several ways to keep global and application keys separate from each other.

- Many installations assign PF keys 1-12 to applications and use only PF keys 13-24 as global PIE keys.

- Users can deactivate the global PIE keys with the SET KEYS OFF command and reactivate them with SET KEYS ON. You can make this easier by setting a key to toggle the keys on and off with the TOGGLE command. The TOGGLE key remains active when you turn global PIE keys off. You determine whether global PIE keys are set on or off when the user signs on.

- The SKIPNEXT command suppresses the next PIE global key pressed and allows the application function defined for that key to be performed instead. For example, the command SKIPNEXT 2 suppresses the PIE function and performs the application function for the next two PIE global keys pressed.

To set up the global PIE keys, edit the following fields the Global PIE Keys screen.

**Keys Choice** (Yes or No) to have active global PIE keys when the user signs on to PIE/CICS. This field is on the right side of the command line.

**PF 1-12, PA 1-2, and PF 13-24, PA 3**

Commands assigned to global PIE keys. Refer to the notes above for information about PIE commands that can be assigned to these keys. Leave a key field blank if you want the existing application command to remain active for this key in the PIE session.

Do not assign PA keys to switch sessions. Data entered on a session screen is lost if you switch with a PA key before pressing the ENTER or a PF key. This is a 3270 hardware limitation. When a PA1, PA2, or PA3 key is pressed, new screen data is not sent to CICS; so MultiCICS is unaware of this data and cannot save it.

Keys can be defined on two Global PIE Keys screens. Press PF8 and PF7 to toggle between screens.
Chapter 3  Profiles

7. Enter 4 from the command line of the Profile utility main screen.

Use the Profile Sessions Configuration screen to pre-define user sessions. You can pre-define applications to run in specific sessions and assign their names and titles as they appear on the Sessions menu.

Each user can have up to 99 sessions with MultiCICS. Without MultiCICS, each user is restricted to a single session. The maximum number of sessions a user can have allocated is determined in the user’s User Directory entry. If you define more than this number in the profile, PIE/CICS allocates sessions up to the maximum. (Later sessions will not display on the user’s menu.) The most common or critical sessions should be defined first in the list.

Name Name of the session as it appears on the Name field of a screen.
Title Description of the session that appears on the Sessions menu.
Hide Choice (Yes or No) to hide a session. Hidden sessions are skipped when you issue the SWITCH NEXT or SWITCH PREV commands. Specify YES to hide the session. Leave it blank or type NO to leave the session unhidden.
Execute Command or transaction that executes when a session is initially opened. It is often useful to execute PIE scripts to execute a particular function. See page 93, “Scripts”, of the Operation and Administration Guide and “Running REXX Programs as Scripts,” on page 9 of the PIE/REXX User Guide for more information about scripts. If you want this session to close when the user closes the application running in it and prevent the blank screen, use the AutoEnd command. This field expects a PIE Application command. To code an Environment command, prefix it with EC.

Use PF8 and PF7 to scroll forward or backward within the Sessions Configuration screen. You can press PF7 from the first screen to access session 99.
Maintaining Profiles

The remainder of this chapter describes procedures to maintain profiles. In most cases, maintenance procedures are completed from the Profile List screen.

Panel Definition Screen Commands and PF Keys

There are commands and PF keys associated with each of the three screens of the Profile utility. Most maintenance procedures require you to enter a line command from the Profile List screen to select the screen and complete the indicated maintenance action.

For the Profile List screen commands and PF keys, see “Profile List,” on page 58 of the Operation and Administration Guide.

The Profile List supports the following line commands.

- **C** Copy
- **D** Delete
- **R** Rename
- **S** Select

Other Profile Screen Commands

Beginning with the main screen, each of the Profile utility screens accepts many or all of the commands shown in the following list. Enter the command from the command line or press the associated key.

Display online help to see which commands are supported by a particular profile screen.

- **HELP (PF1)** Elicit online help.
- **END (PF3)** Exit the current screen without saving. When pressed from the terminal, keys, or sessions screens, it returns you to the Profile utility main screen. When pressed from the main screen, it returns you to the Profile List. Be sure to press PF5 to save your changes before pressing PF3 from the main screen.
- **RETURN (PF4)** Terminate the Profile utility without saving changes and return to the highest level—the Sessions menu, the last “main” menu, or end of session. Press PF5 first to save your changes before exiting.
- **SAVE (PF5)** Save changes. If you have changed the PIE profile group or PIE profile name fields, a new profile will be created. If the new name already exists, a message window will ask you to confirm whether you want to replace the old screen with the new.
- **RETRIEVE (PF6)** Retrieve a profile by name. Retrieving a profile is just like selecting it from the Profile List.
  - The RETRIEVE command replaces what is currently on the screen and does not save any changes. Issue SAVE before RETRIEVE to save your changes.
- **UP (PF7)** Display the previous page of data. It is accepted only at the Keys and Sessions screens.
Chapter 3 Profiles

DOWN (PF8) Display the next page of data. It is accepted only at the Keys and Sessions screens.
DELETE (PF9) Delete the current profile from the Repository.
RESHOW (CLEAR) Discard unsaved changes and refresh the screen.
CANCEL Discard unsaved changes and return to the Profile List screen.
UNDO Discard all changes accumulated since entry to this screen, including the last SAVE or RETRIEVE (whichever is the latest action), and remain in this screen.
PROFILE Return to the main screen from the terminal, keys, or sessions screen.
TERMINAL or 1 Switch to the terminal screen.
KEYS or 3 Switch to the keys screen.
SESSIONS or 4 Switch to the sessions screen.

Deleting a Profile
1. Invoke the Profile utility, as described in “Editing a Profile,” on page 34.
2. Enter D in the command field next to the profile to be deleted.
3. Press ENTER to confirm the delete.
   You can press PF3 to cancel the delete request and retain the profile.
4. Press PF6 to refresh the screen and verify the profile has been deleted.

Renaming a Profile

Complete the following steps to rename a profile:
1. Invoke the Profile utility, as described in “Editing a Profile,” on page 34.
2. Enter R in the command field next to the profile to be renamed.
3. Enter the new profile name in the message box that appears.
4. Press PF6 to refresh the screen and confirm the profile has been renamed.
Chapter 4 Terminal and User Directories

The Terminal and User directories establish specific operational characteristics of PIE/CICS terminals and users. The Terminal Directory identifies all terminals authorized to access PIE/CICS. The User Directory identifies all users who can sign on to PIE/CICS.

Both directories are discussed together in this chapter because they are similarly designed with many of the same functions. For example, the User Directory entry defines which profile the user runs, and the Terminal Directory entry defines which Logon Director screen is displayed when the terminal is turned on. See “Terminal Directory,” beginning on page 46, and “User Directory,” beginning on page 52, for the characteristics you can define with either directory. Both directories allow you to specify wildcards to generically authorize groups of users and terminals.
Required Directory Customization

Only system administrators and the last person to update the directory can access or update the Terminal and User Directories. Access by the last person to update the directory is a unique form of authorization to allow corrections to be made in the event all system administrator authorization was accidentally deleted.

The User Directory should be edited soon after PIE/CICS is installed. The default User Directory authorizes all users as system administrators and grants them complete access to install and customize PIE/CICS. After PIE/CICS is installed and customized, system administrator access should be restricted to only site personnel who administer PIE/CICS.

The default values of the Terminal Directory shipped with PIE/CICS may be acceptable and not require further customization. The default values authorize all terminals to sign on to PIE/CICS through the standard Logon Director screen. See “Terminal Directory,” beginning on page 46, and “User Directory,” beginning on page 52, for information about customizing both directories.
Directory Entries

The Terminal Directory is searched when the Logon Director runs (when the terminal is turned on or the user enters the PIE transaction, etc.). The User Directory is searched when the user logs on. For both directories, PIE/CICS selects the first matching entry of the terminal or user ID.

MODEL is the first entry in both directories. This entry is never used to match terminal or user IDs. Instead, MODEL contains the defaults for all other entries in a directory. All subsequent entries define terminal or user IDs. If a field is left blank in these entries, the corresponding value from the MODEL entry is substituted as the default value.

Wildcard characters identify more than one terminal or user ID in a field. The wildcard * identifies any character (or no characters) in that position and forward. The wildcard ? identifies any character in a field position. The wildcard % identifies any number in a field position.

It is important to list entries in the correct order within a directory. Place more specific entries above generic entries. For example, imagine preparing entries for a payroll department. The user IDs within the department are PAYnnn, where nnn is a unique three-digit number that identifies each employee. Most users have the same PIE/CICS requirements. A single entry, PAY*, serves all users within the department. The supervisor, however, needs more PIE/CICS authority than other users. The supervisor’s user ID is PAY001. A separate entry is prepared for PAY001 and it is entered above the PAY* entry for all other users in the department.

PAY001   * . .
PAY*   * . .

If you have an entry that is unrelated to all other entries, you can place it anywhere within the directory. More frequently-used entries should be placed towards the top of the directory to reduce the search time needed to find the entry.


Usually, only a few entries are needed: for both directories:

- MODEL entry
- Unique system and group administrator entries in the User Directory
- Wildcard * entry to define all terminals or users

You should end both directories with an * entry. This entry authorizes all terminals or users to access PIE/CICS.

There is no size limit for either directory except temporary storage space requirements for a CICS/VS system and the VSAM data space for the PIE/CICS Repository.
Terminal Directory

Terminal Directory entries are created and maintained using the Terminal Directory utility, which consists of a series of screens containing data entry fields. This section discusses each entry field and how to maintain directory entries.

Overriding Directory Defaults

Many Terminal Directory attributes, such as network optimization, can be overridden by the PUXINIT exit. A preset user ID and password can be specified in the PCTSPDCX exit. See “PUXINIT—Post Sign on Initialization,” on page 149, and “PCTSPDCX—Logon Message Exit,” on page 140 for more information about both exits.
Editing a Terminal Directory Entry

You should begin customizing the Terminal Directory by editing the MODEL entry. Complete the following procedure.

1. Invoke the Terminal Directory utility by one of the following methods:
   - If you are using the ACCOUNT profile, select session 1 from the Sessions menu. The PIE master menu should appear on the screen. Select option 1, the System Administrator menu. Then select option 2, TERMINAL.
   - Access the PIE/CICS System Administrator menu using the transaction PEXE MENU PIEMADM. Select option 2, TERMINAL.
   - From any session, enter: ==TERMDIR

   The following figure shows an example of the Generic Terminal Directory screen.

   2. Update the following fields of the Terminal Directory screen. Remember, PIE/CICS substitutes the corresponding value from the MODEL entry if a field is left blank.

   Terminal
   - VTAM node name or CICS termid. You can use wildcards (*, ?, and %). Always begin CICS termids with a semi-colon (:). For example, specify the CICS termid 1234 as :1234.
   - Code terminal entries with more specific entries above generic entries. The final entry should be an * that generically authorizes all terminals to use PIE/CICS. The Terminal field is the only required field for non-model entries.
   - PIE/CICS supports only 3270-compatible terminals.

   If a terminal ID is not included in the directory, the Logon Director issues the XCTL command to transfer to the CICS good morning program. PIE/CICS processing is not permitted at the terminal. See “Receiving Sign On Data,” on page 16 for more information.

   The Userid and Password fields automate user sign ons at specific terminals. See “Automating User Sign Ons,” on page 6, for more information about using this feature. You must specify the full user ID and password. Wildcards are not allowed.

   Userid
   - User ID for automated sign ons at this terminal. The value you
enter must correspond to an authorized ID in your security system, unless you specified no PIE security interface. (See “Changing a Site Security System,” on page 82, for details.)

PIEUSER is a special user ID that bypasses the Logon Director screen. When a user logs on, the ID is converted to PIEUtttt, where tttt is the user’s CICS termid. See “Pre-defined sign on Data to Specific Terminals,” on page 11 for more information.

**Password**
Password for automated sign ons at this terminal. The password must correspond to the authorized password of the user specified in the Userid field. See “Changing a Site Security System,” on page 82, for details.

**Logon**
Name of the Logon Director screen that appears when the terminal accesses CICS. The sample Logon Director screen shipped with PIE/CICS is called PIELMAIN.

If the screen you specify is not defined or accessible, PIE/CICS uses the default screen. The default log on screen is internally defined. It does not appear on the Panel List.

Specify NONE if you do not want a log on screen displayed (transparent mode), or if you are using another log on system.

**Logoff**
Terminal log off authorization from CICS.

- **YES** Allow CICS log offs from this terminal.
- **NO** Prohibit CICS log offs. If users attempt to log off, they are signed off PIE/CICS and the log on screen appears on the terminal.
- **USE** Use the value in the CICS TCT entry for this terminal. The value used is set in RDO TYPETERM, the Discreq field.

**PIE**
Terminal action after the Logon Director (transaction PSGM) is invoked. Terminal action depends on whether the Logon Director is invoked as the good morning message, or explicitly by the user.

- **YES** Display the Logon Director screen when the Logon Director is invoked.
- **NO** Deny the terminal access to the Logon Director and PIE/CICS. When the Logon Director is invoked, either by the user or as the good morning message, PIE/CICS sends the user to the CICS DFHGMM good morning program instead.
- **REQ** Allow the terminal access to PIE/CICS only if the user issues the PIE or PSGM transaction. When the Logon Director is invoked as the good morning message, PIE/CICS sends the user to the CICS DFHGMM good morning program. Only after users request PIE/CICS by entering the PIE or PSGM transaction is the Logon Director screen displayed.
- **USE** Allow the terminal to access PIE/CICS only if the user requests it with the PIE or PSGM transaction. USE differs from REQ in that the Logon Director signs on the terminal to PIE/CICS automatically if the terminal is currently.
signed on to CICS. The Logon Director screen is not displayed and users do not have to enter sign on information.

PIE/CICS uses the current user ID as both user ID and password. If the password recognized by PIE/CICS is incorrect, PIE/CICS cannot process automatic sign ons for the user. Also, if the user’s terminal is locked (manually or with AutoLock), the user must enter their user ID and password to unlock it. Except in special circumstances, REQ is the preferred option.

See “Logon Director Actions at User Sign On,” on page 4, and “Receiving Sign On Data,” on page 16, for details on Logon Director processing.

Dial
Terminal connection as either a dial-up port or contention port. Specify YES if the terminal uses a dial-up port. Otherwise, specify NO.

PIE/CICS must recognize a terminal that uses a dial-up port to prevent possible security violations. When a terminal connects to another VTAM application using the TSWITCH command, PIE/CICS disconnects the terminal from the previous application. When the user returns to the originating region, PIE/CICS reconnects the user automatically. However, if the terminal uses a dial-up port, a second terminal may have acquired the original terminal ID during the period the terminal was disconnected. If PIE/CICS reconnects the terminal automatically, the original user would have access to the second user’s sessions. To prevent this from occurring, PIE/CICS does not automatically sign on terminals to dial-up ports when users return to an application. Instead, the Logon Director screen is displayed and users must manually sign on.

Opt
Type of 3270 NetMizer datastream optimization used by this terminal. The optimization method is used as the default for all sessions that run on this terminal. You can change an optimization method for a session using the same values as PIE State commands. Valid types of data stream optimization include:

- **OP1** Level-1 optimization, which includes outbound optimization and normal imaging.
- **OP2** Level-2 optimization, which includes outbound optimization and semantic imaging.
- **OP3** Level-3 optimization, which includes outbound and inbound optimizations plus normal imaging.
- **YES** Same as Level-3 optimization.
- **FUL** Full optimization, which includes outbound and inbound optimizations plus semantic imaging.
- **NO** No optimization.
N32  No optimization is applied to this terminal. N32 differs from NO by not maintaining a screen image. N32 disables the following features:
    Session switching
    Lock and Autolock
    Receipt of super-messages
    Disconnect/reconnect (session outage recovery)
    Bulletin command
    Use of Global PIE keys
    Screen prints with the PIE PRINT command
    Script
    View capture by another terminal
    Cut and Paste
    If these features are not required at this terminal, specify N32 to improve performance.

SLM  Short Logon Message. Specify YES to display the message Connected to cics-name, press Enter to continue. rather than the Logon Director screen when a terminal logs on to CICS. When users press ENTER, they receive the normal log on screen.

A short Log on message is useful for CICS networks with a large number of auto-sign on terminals. The short log on message requires less resources and places less stress on a CICS region at start-up. See “Short Log on Messages,” on page 159, for details. Specify NO to display the log on screen instead.

TRM  TSWITCH Return Message. TRM determines the screen that appears after a user returns to the original region from a TSWITCH operation when the terminal is on a dial-up port or the user has exceeded the TSWITCH time-out limit.
      F  Full Logon Director screen.
      S  Short log on message. See “SLM” above.
      N  None. Users receive the Logon Director screen when they press ENTER.
      The short option leaves any messages displayed by the application partially intact you are returning from. The N option leaves such messages completely intact.

RST  Terminal Reset
      YES  Reset the terminal port if your site uses the IBM 7171 ASCII Device Attachment Control Unit. PIE/CICS sends the 7171 reset sequence to the terminal when the user transfers to another region with the TSWITCH or PASS commands. The reset sequence restores the port and makes it available to another terminal.
      NO  Disable the terminal reset function if your site does not use a 7171 control unit.

3.  Press PF3 to save your changes and exit the screen when you have finished editing records.
Creating a New Terminal Entry

1. Invoke the Terminal Directory, as described in “Editing a Terminal Directory Entry,” on page 47.
2. Use the Insert, Copy, or Repeat line commands to create a new entry in the Terminal Directory.
   
   If you want to create an entirely new entry, type I (or inn) on the command field of the line where you want to insert the new entry in the Terminal Directory listing. To insert an entry on the first line, enter INSERT from the command line of the Terminal Directory screen.
   
   If you want to create an entry using an existing entry as a template, type R in the command field of the entry to be duplicated. If you want to copy an entire block of entries, type RR on the command fields of the first and last entries of the block.
   
   Use the C line command to copy an entry to another location within the Terminal Directory. First, type C (or C nn) on the command field of the entry you want to copy. Next, identify where you want to place the copy by entering an A (after) or B (before) on the command field of the line that will be immediately above or below the copied entry.
   
   If you want to copy a block of entries, type CC on the command fields of the first and last entries of the block. Type A or B where you want the block of entries copied.
3. Edit the new entry and make any changes described in “Editing a Terminal Directory Entry”.

Deleting a Terminal Entry

1. Invoke the Terminal Directory utility, as described in “Editing a Terminal Directory Entry,” on page 47.
2. Type D in the command field of the entry you want to delete and press ENTER. If you want to delete a block of entries, type DD in the command field of the first and last entries in the block; then press ENTER.

Renaming a Terminal Entry

To rename an entry, simply modify the entry, as described in “Editing a Terminal Directory Entry,” on page 47.

Reorganizing Terminal Entries

You should periodically reorganize your entries in the Terminal Directory. It is important to keep terminal entries in the right order to ensure efficient search operations. Generally, more specific entries should be placed above more generic entries in the directory listing.

1. Invoke the Terminal Directory utility, as described in “Editing a Terminal Directory Entry,” on page 47.
2. Type M on the command field of the entry you want to move.
3. Identify the destination by entering A (After) or B (Before) on another line within the directory. The moved line is placed immediately below or above this line. Then press ENTER.
   
   If you want to move a block of entries, type MM in the command field of the first and last entries in the block. Type A or B where you want the block moved. Then press ENTER.
4. Press PF3 to save your changes and exit after you have finished reorganizing the entries within the Terminal Directory.
Chapter 4  Terminal and User Directories

User Directory

The User Directory defines a great deal of customization information. In it you determine:

- PIE/CICS user status
- Membership in a PIE/CICS user group
- Authority to sign on to PIE from multiple terminals
- Default profile
- Maximum number of active sessions
- Default menu, bulletin, idle time before an automatic lock, and other environment defaults

There are two general methods to specifying entries in the User Directory. You can create a few general entries to handle your whole company, assigning the same environmental conditions to most users. Or, you can create a large number of entries, creating highly specialized environments. The first method provides a low maintenance system without specialized customization. The second method is more complex, requiring more ongoing maintenance.

The User Directory utility has two screens. The first screen is similar to the Terminal Directory utility. It lists the entries and allows you to update fields. However, the User Directory entries have more fields than will fit on a line entry. These fields are changed in a second detail screen.

Be careful making changes to the User Directory. If you delete all valid entries, no one will be able to access PIE/CICS, including system administrators. If you inadvertently delete all administrator entries, no one will be able to change the User Directory. In either case, you must reload the User Directory from a backup Repository.

User Directory Maintenance

User Directory maintenance varies by the number of entries within the directory. If the directory contains only a few generalized entries, little maintenance is required. However, if you create a large number of entries for specialized environments, the User Directory may require regular, systematic maintenance.

You can help automate maintenance with the Post Signon Initialization exit. For example, you can get User Directory information from your external security system. See “PUXINIT—Post Sign on Initialization,” on page 149, for more information.

User List Utility

You can see a list of users currently signed on to PIE/CICS with the User List utility. See “User List,” on page 63 of the Operation and Administration Guide for details.
**Default User Directories**

The User Directory shipped with PIE/CICS has three entries. MODEL is the first entry with default values that are substituted for blank fields of other entries. The second entry is an asterisk entry with system administrator authority. The third entry is also an asterisk entry without administrator authority. The third entry is intended for all other users without system administrator authority.

The first asterisk entry qualifies everyone as a system administrator. You should change this entry as soon as possible to limit the number of users with system administrator authority.

PIE/CICS creates a replacement directory if the User Directory for a TOR is unavailable or all entries are deleted in the User Directory. The only difference is in the second entry, the system administrator entry: in the starter directory, the system administrator entry specifies the user ID of the first user to sign on the TOR, not an asterisk.

**Overriding Attributes**

Most attributes, such as groupid, security classes, and user ID type, etc., can be overridden by the Post Signon Initialization exit.
Editing a User Directory Entry

You should begin customizing the User Directory by editing the MODEL entry. Complete the following procedure.

1. **Invoke the User Directory utility by one of the following methods:**
   - If you are using the ACCOUNT profile, select session 1 from the Sessions menu. This will access the PIE master menu. Select option 1, the System Administrator menu. Then select option 3, USER.
   - Access the PIE/CICS System Administrator menu using the transaction PEXE MENU PIEMADM. Select option 3, USER.
   - From any session, enter: USERDIR

   The following screen shows an example of the PIE Generic Users Directory

   ![Example of PIE Generic Users Directory](image)

   If your User Directory list is long and you want to locate a particular user entry, type the user ID on the command line. You will receive the detail screen for that entry.

   You can also specify the user ID as a parameter of the USERDIR command when you access the User Directory. You go directly to the user entry. When you exit that entry, you also exit from the User Directory.

2. **Update the following fields.**

   If a field is left blank, PIE/CICS substitutes the corresponding value from the MODEL entry.

   **Userid**
   
   User ID for automated sign ons at this terminal. The value you specify must correspond to an authorized ID in your security system, unless you specified no PIE security interface. (See “Changing a Site Security System,” on page 82, for details.) PIEUSER is a special user ID, which bypasses the Logon Director. Specify a user ID. You may use the wildcards *, ?, and %. Be sure
to code the entries in the correct order. (See “Directory Entries,” on page 45.) This field is required.

In certain circumstances, users will be signed on as “PIEUSERs” instead of their regular user ID, such users will be signed on with the special user ID of PIEUtermid. Ensure that this ID is authorized. You may code a special entry of PIEU* for these IDs.

**Terminal**

VTAM node names or CICS termids that identify terminals that can access PIE/CICS. Begin CICS termids with a :. You may use the wildcards *, ?, and %. This field is required.

Usually, an asterisk is specified in this field that authorizes all terminals to PIE/CICS.

**Password**

Password mask if you want PIE/CICS to perform password screening. This field is required.

Usually an asterisk is specified in this field to bypass password screening.

**Groupid**

Name of the Group to which this user belongs. If both the line entry and model entry are left blank, the group name defaults to SYSTEM.

The user is authorized to use PIE objects with this Group ID. Group administrators are authorized to change and create objects with this Group ID. Only system administrators have authority over objects with the SYSTEM group ID.

**Menu Name**

Name of the user’s default PIE/CICS screen. This screen appears whenever the user invokes the MENU command without specifying the menu name as a parameter.

**Profile**

Name of the user’s default profile. See “Assigning User Profiles,” on page 30, for information on profile selection. Access the Profile utility for a list of existing profiles. (See “Editing a Profile,” on page 34.)

There is a second screen of fields for User Directory entries.

3. If you want to edit more fields of an entry listed on the Generic Users Directory screen, do one of the following:

   - Place the cursor on the entry and press PF2.
   - Enter the S line command next to entry and press ENTER.
   - Type all or part of the User ID in the command line and press ENTER or PF2.
The following figure shows an example of the detail screen of the Generic Users Directory that appears after selecting an entry.

SHAPE GENERIC USER DIRECTORY PORTION -------------------------- (C) 1995 TSC, Inc

Command ===> Update History

Userid ===> (MODEL) System Admin ===> N (Y/N) Name:
Terminal ===> (MODEL) Userid Type ===> U (U/R/S) Term:
Password ===> (MODEL) (Unique, Reusable, Shared) Date:

Generic forms, ?, % and * characters are accepted above.

RESOURCES:
Groupid ===> Group Administrator ===> N (Y/N)
User Data 1 ===> Lock Timeout ===> 0 (Min)
Menu ===> PIEMMAIN Display Sign-on Msg ===> (Y/N)
Profile ===> USER Log Sign-on Msg ===> (Y/N)
Bulletin ===> PIEMMAIN Allow Disc Status ===> (Y/N/A)
Helpdesk ID ===> (NONE) Allow Session Cancel ===> Y (Y/N)

Maximum Sessions ===> 12 (1-99)
Printer Id ===> Private Objects ===> N (Y/N/O)
Alt Printer ===> Security Classes ===> (0-7)
Language ===> TSM Timeout ===> (Y/N)
Receive SuperMessage ===> (Y/N)

PF: 3 End 4 Return

Update any of the following fields shown on the detail screen.

Userid Refer to the previous description of this field given on page 54.
Terminal Refer to the previous description of this field given on page 55.
Password Refer to the previous description of this field given on page 55.
System Admin Choice (Yes or No) to grant system administrator authority to these users. System administrators have the authority to update all PIE/CICS resources.
Userid Type Type of userids.
U Unique for one user at one terminal. If a user attempts to sign on with a unique user ID when that ID is already assigned to another terminal, an error occurs. When a unique user ID is disconnected, the user may sign on again at any terminal to reconnect to PIE/CICS.
R Reusable IDs can be signed on to many different terminals at once. Reusable IDs are helpful for users who must sign on at several different terminals. If someone signs on with a reusable ID that is already signed on, PIE/CICS will give the user a new set of sessions, as if the ID were being signed on for the first time.
If a reusable ID is disconnected, users may sign on again from any terminal and their ID will be reconnected to their previous session. If more than one terminal is disconnected at once, sign on will be reconnected in the order of disconnection. For this reason, you may not want to allow more than one person to use a particular reusable ID.

Shared IDs can be signed on to many different terminals at once. They differ from reusable IDs in the way they are reconnected.

If the shared user ID is disconnected, users must sign on from the same terminal or specify their previous terminal ID at sign on to reconnect to their previous sessions. If users sign on from another terminal without specifying the previous terminal ID, they are given a new sign on.

Because of the extra security on reconnection, you may give multiple users the same shared user ID.

### Groupid
Refer to the previous description of this field given on page 55.

### User Data
Default user data assigned to the &ZUDATA variable. This data is supplied whenever the &ZUDATA variable is used. Refer to the PIE/CICS Command Reference for more information about using data variables.

### Menu
Refer to the previous description of this field given on page 55.

### Profile
Refer to the previous description of this field given on page 55.

### Bulletin Name
Name of the default bulletin board menu. This menu is invoked when the user issues the BULLETIN command.

### Helpdesk
IDs of users authorized to display other user screens with the VIEW utility. These users are typically helpdesk personnel who need to diagnose user problems by remotely displaying their screens. You may use wildcards.

System administrators have authority to view all user screens. Group administrators have authority to view the screens of users in their groups. The Helpdesk field does not affect this authority. To remove View authority for system and group administrators, use the PUXVSCN exit.

If the users defined in this User Directory entry have access to sensitive information, such as payroll data, the users authorized to view these screens should be limited. Leave the Helpdesk field blank to deny view authority to all users except system and group administrators.

Users can also set this value with the HELPDESK command.

### Printer Id
CICS termid of the default printer. This printer is used when the
user executes the PRINT command or presses the CICS print key. Users can change their printer with the PIE PRINTID command.

Alt Printer

CICS termid of the alternate printer. This printer is used when the default printer is busy or unavailable. Users can change this printer with the PIE PRINTID command.

Group Administrator

Choice (Yes or No) to grant group administrator authority to these users. Group administrators can update resources belonging to their group (specified in the Groupid field). Specify N to deny group administrator authority.

Lock Timeout

Number of minutes users can leave their terminals unattended before the terminals are automatically locked. Users can unlock their terminals by entering their password at the prompt, or by pressing PF3/PF15 to exit PIE/CICS. To disable AutoLock, specify 0.

You can specify longer lock values for different applications using the Time-out Exit. See "PCSSPTOE—Time-out Exit," on page 137, for details.

Display Sign-on Msg

Choice (Yes or No) to display sign on messages returned from an external security system. Specify N to suppress them. You determine how long messages are displayed with the Signon Message Display Time field of the Customization Options utility.

Log Sign-on Msg

Choice (Yes or No) to record user sign on messages to the Signon message log, no matter what is entered in the Display signon msg field. You can access the sign on message log with the BROWSE PIE.SIGNMSG command.

Allow Disc Status

User authorization to disconnect from and reconnect to PIE/CICS. Y authorizes disconnect in all circumstances: from TSWITCH and PASS and terminal outage and the EXIT DISC command.

N authorizes disconnect only with TSWITCH and PASS. With terminal outage and the EXIT DISC command, users are logged off instead.

A (always) logs the user off for all disconnect requests except TSWITCH requests.

Again, for N and A, the EXIT DISC command and terminal outages are treated as log offs. The user’s open sessions are cancelled.

Some sessions can be defined as non-cancelable. (See “CANTABLE: Prevent Transactions from Being Cancelled,” on page 184, for more information.) If the user has a non-cancellable session open and the EXIT DISC command is entered, PIE/CICS switches the user to the session and issues a message that the session must end normally. If a
non-cancellable session is open and the user has a terminal outage, no message is possible. In this case, the user is disconnected.

Allow Session Cancel
Choice (Yes or No) to allow users to cancel sessions either manually or automatically when they log off from PIE/CICS. Specify N to prohibit all sessions from being cancelled.

For you can protect specific applications from being cancelled with the Cancel Table. See “CANTABLE: Prevent Transactions from Being Cancelled,” on page 184.

Maximum Sessions
Maximum number of sessions that users can allocate or open at once. Up to a maximum of 99 open sessions are available if MultiCICS is installed. Otherwise, users are restricted to a single session.

Private Objects
Choice (Yes or No) to grant users the authority to create private objects. A private object is one that has the user’s ID as its group ID.

If you specify Y, users are able to create private objects. Any changes they make to their profile are saved automatically when they log off.

If you specify O (for only), users have authority to create private objects, but changes to their profiles are saved only if they explicitly request SAVE from the Profile utility.

Specify N to deny private object authority. Users can change existing private objects and save their changes, but they cannot create new private objects.

Security Classes
Commands authorized to users based upon the security class of the commands. Security classes govern use of PIE commands. The PIE/CICS Command Reference lists commands and their security classes.

List command security class numbers consecutively, with no intervening blanks. For example, to grant authority for classes 0, 1, and 3, type 013. To remove authority for all classes 1 through 7, type 0.

For all users must have class 0 authority. However, specifying 0 only removes authority for classes 1-7. To grant authority for all classes, leave the field blank.

Language
Specify the default language if you are using PIE/CICS Custom Menus.

TSM Timeout
Choice (Yes or No) to manage user’s terminals with the Terminal Security Monitor. Terminals that have been idle for an extended period of time are disconnected from PIE/CICS and CICS after they have exceeded a time-out period. The terminal is returned to VTAM. (This field corresponds to the PUXITMSE field passed to PUXINIT) The default is Y.
Receive SuperMessage
Choice (Yes or No) to permit the user to receive SuperMessages. (This field corresponds to the PUXINMSG field passed to PUXINIT.) The default is Y.

4. **Press PF3 to exit the screen after you have finished editing the directory.**
   PF3 does not save the changes you made on this screen. You must enter SAVE, END, or RETURN from the primary screen to save your editing changes to the User Directory.

5. **Press PF3 and exit from the User Directory utility.**

**Creating a New User Directory Entry**

1. **Invoke the User Directory utility, as described in “Editing a User Directory Entry,” on page 54.**

2. **Use the insert, copy, or repeat line commands to create a new entry.**
   If you want to create an entirely new entry, type i (or inn) on the command field of the line where you want to place the entry. To insert an entry on the first line of the User Directory, enter INSERT on the command line.
   If you want to create an entry based on one that already exists, type R in the command field of that entry. If you want to copy an entire block of entries, type RR on the command field of the first and last entries in the block.
   You may also use the C line command to copy the model to another location within the User Directory. First, type C (or Cnn) in the command field of the entry you want to copy. Next identify where you want the line copied to: type A (for after) or B (for before) in the command field on that line. Then press ENTER.
   If you want to copy a block of entries, type CC in the command field of the first and last entries in the block. Type A or B where you want the block copied. Then press ENTER.

3. **Modify your new entry as described in “Editing a User Directory Entry.”**

**Deleting a User Entry**

1. **Invoke the User Directory utility, as described in “Editing a User Directory Entry,” on page 54.**

2. **Type D (or Dn) in the command field of the entries you want to delete and press ENTER. Or, type DD in the command field of the first and last entries in a block; then press ENTER.**

**Renaming a User Entry**

To rename a screen, simply modify its entry, as described in “Editing a User Directory Entry,” on page 54, and change the User field.
Reorganizing User Directory Entries

Because it is important to keep User Directory entries in the right order (see “Directory Entries,” on page 45, you may want to reorganize your entries after they have been created. To do so, perform the following.

1. **Invoke the User Directory utility, as described in “Editing a User Directory Entry,” on page 54.**

2. **Type M in the command field of a line entry you want to move.**

3. **Identify the destination by entering an A (after) or B (before) on the line to which you want the entry moved.**

   If you want to move a block of entries, type MM in the command field of the first and last entries in the block (or Mn in the first entry of the block). Type A or B on the line where you want the block moved.

4. **Press PF3 to save your changes and exit the utility when you have finished moving entries.**
Terminal and User Directory Utility Commands

The Terminal and User Directory utilities accept the following commands.

HELP (PF1) Browse the help file.
SAVE Save your changes and continue editing.
RESHOW (CLEAR key) Discard changes that were typed onto the screen but not entered with ENTER or a PF key. Then refresh the screen.
CANCEL Discard all unsaved changes and end the utility.
UNDO Discard changes made since the last SAVE, refresh the screen, and continue editing. If you press PF3 to exit the detail screen of the User Directory, changes on that screen will be saved.
RECALL Recall the last command entered.
OBJID Display the group and member name of the Terminal or User Directory. (The group name for both directories is SYSTEM.)
SELECT (PF2) Select the entry indicated by the cursor or on the command line and present the User Directory detail screen. (Available for the User Directory only.)
END (PF3) Save your changes and exit the utility.
RETURN (PF4) Save your changes, exit the utility, and return to the highest level—the Sessions menu, the last "main" menu, or end of session.
UP (PF7) Scroll back in the list.
DOWN (PF8) Scroll forward in the list.

Your profile has a default scroll amount. You can change the scroll amount by typing any of the following in the Scroll field.
M or maximum.
P or page.
H or half.
C or cursor.
D or data—that is, page forward or back, minus one line; the last line of data becomes the first line or the first line becomes the last line.

nnn—that is, the number of lines to scroll, for example, UP 37.
User Directory Line Commands

A    Move or copy marked entries after this entry.
B    Move or copy marked entries before this entry.
C or Cnn or CC Copy one or more lines or a block of lines before or after an entry marked with A or B.
D or Dnn or DD Delete one or more lines or a block of lines.
I or Inn Insert one or more blank lines after this line.
M or Mnn or MM Move one or more lines or a block of lines before or after an entry marked A or B.
R or Rnn or RRnn Repeat one or more lines or a block of lines nn times, after this line.
S    Select entry to edit in the User Directory detail screen.

User Directory Detail Screen Commands

The User Directory detail screen accepts RESHOW, CANCEL, UNDO, END, and RETURN commands.

RETURN saves your changes, ends the utility, and returns to the highest level—the Sessions menu, the last “main” menu, or end of session.

END exits to the primary User Directory screen. It does not save your changes to disk. To do so, issue the SAVE or END command from the primary screen.

The other commands work as they do in the primary screen.
Chapter 5  Customization Options

System options set system-wide operational default values for PIE/CICS. These options are specified from the PIE/CICS Customization Options utility and stored in the Options file. This chapter describes system options and the Customization Options utility.

Each TOR must have access to an Options file. You can have a separate Options files for each region, or share them between regions. To share an Options file, define ddname PC@OPTN in the File Control Table (FCT) as read-write in one CICS region and read-only in all other regions. For read-only regions, you may copy the FCT entry in member MROFCT of the CNTL dataset. (During installation, you will have defined PC@OPTN as read-write in the region into which you installed. Now, you need simply to define read-only access to other regions.)

To have separate Options files in each region, define separate datasets as read-write in all regions. Option file sharing is explained more fully in "Multiple CICS Regions," beginning on page 101.

The Options file also contains password information, which is set by completing the fields of the License panel. The License panel is described in "Enter PIE/CICS License Information," on page 14 of the PIE/CICS Installation Guide.
Chapter 5  Customization Options

Using the Customization Options Utility

1. **Invoke the Customization Options utility by one of the following methods.**
   - If you are using the ACCOUNT profile, select session 1 from the Sessions menu. The PIE master menu should appear. Select option 1, the System Administrator menu. Then select option 10, OPTIONS.
   - Access the PIE/CICS System Administrator menu using the transaction PEXE MENU PIEMADM. Select option 10, OPTIONS.
   - From any session, enter: ==PSYS
     PIE/CICS does not have to be operational to run PSYS.

   The following figure shows an example of the first Customization Options screen.

   ![Customization Options Screen](image)

   PIE/CICS displays the default values of each screen field if no options are stored currently in a region’s Options file.

2. **Update any of the following fields of the first Customization Options screen.**
   - Maximum Logon Attempt Count
     Maximum number of attempts users can enter their password to sign on to PIE/CICS or unlock their terminal and restore their sessions.
     If the log on attempt count is exceeded at sign on, the terminal is placed in protected mode. System administrators must use the Protected Terminal List to restore the terminal to unprotected mode. See “Protected Terminal List,” on page 60 of the Operation and Administration Guide.
     If the limit is exceeded during an attempt to unlock a terminal, the user ID is disconnected from PIE/CICS and the terminal placed at the Logon Director screen.
     Specify 0 to disable the maximum logon attempt count. The default is 5 attempts.
Using the Customization Options Utility

• Avg No. of Concurrent Users
  Anticipated average number of concurrent PIE/CICS users. The value in this field is used with the Avg No. of Sessions per User field to calculate the amount of storage required for various PIE resident tables. An accurate average should allocate sufficient system resources without wastage. The default is 100 concurrent users.

• TSWITCH Auto-Signon timeout
  Maximum number of minutes users can transfer out of CICS to another VTAM application with a TSWITCH command and still remain signed on to PIE/CICS. If the time-out limit is exceeded, users are returned to the Logon Director screen instead of the original session. After successfully signing on again, the user’s sessions are restored. Specify 0 to disable the TSWITCH Auto-Signon time-out limit. The default is 60 minutes.

• Avg No. of Sessions per User
  Anticipated average number of concurrent sessions users are expected to have open. The value entered in this field is used with the Avg No. of Concurrent Users field to calculate the amount of storage required for various PIE resident tables. The default is 5 concurrent open sessions.
  Like the Avg No. of Concurrent Users field, an accurate value in this field provides sufficient resources to PIE without wastage.

• Temp Storage Name Substitution
  Choice (Yes or No) to allow the substitution of unique temporary storage names for each session. If YES is specified, temporary storage names are scanned for values that match fields specified in the TSQFLDS member of the PIE customization macros. See the installation and customization guide for more detailed explanation. By default, the terminal id and operator id are substituted. The default is YES.
  You may need to exclude some applications from temporary storage name substitution. For more information, see “PCSSSEXT—TSQ Substitution Exit,” on page 138.

• MRO/ISC Support
  Type of MRO/ISC support provided by PIE/CICS. There are four possible values: YES, FS, FUL, and NO. There is some resource overhead associated with each type of MRO/ISC support. Specify only the support you need.
  If you are not running MRO/ISC, code NO. If you are running MRO/ISC, consider each of the following factors. A combination of factors may suggest more than one value for the field; for any combination of YES, FS, or FUL, code FUL. If none of the conditions apply, code NO.
  Specify FUL to use AutoEnd with remote transactions that do ATI (EXEC CICS) STARTs back to the terminal. Specify YES or FUL to use the PIE Command API or the User Data API in AORs.
  Specify YES for pseudo termid support. See “TSQFLDS and TERMMASK: Substitute Temporary Storage Queues,” on page 186, for more information on pseudo termids.
PIE/CICS uses either transaction routing or function shipping to support TSQ substitution in an AOR. The method is determined by the application requesting the TSQ.

<table>
<thead>
<tr>
<th>Transaction Routing</th>
<th>Function Shipping</th>
<th>MRO/ISC Support Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>NO</td>
<td>YES</td>
<td>FS</td>
</tr>
<tr>
<td>YES</td>
<td>YES</td>
<td>FUL</td>
</tr>
</tbody>
</table>

For FUL, FS, or YES, PIE/CICS must be initialized in applicable AORs and TORs.

- **Temp Storage on Auxiliary**
  Choice (Yes or No) to maintain PIE/CICS temporary storage within auxiliary storage. YES maintains temporary storage within auxiliary storage. Specify NO to place temporary storage in main memory. The default is YES.

- **Name of Logon Director Help**
  Name of the Logon Director help file. The default name is PIETLDIR.

- **Transient TCTTEs**
  Choice (Yes or No) to delete auto-installed transient TCTTEs when the user logs off or disconnects from PIE/CICS. If you specify NO, TCTTEs are regarded as non-transient and are not deleted. Use NO to provide quicker logon to CICS. The default is YES.

- **AOR AE/ATI support-1 byte field**
  Decimal offset relative to zero for PIE/CICS use in the TCTTE user area. This one byte field is required to support AutoEnd and ATI starts for transactions that execute in an AOR.

  Be sure all PIE terminals have a TCTTE user area large enough to contain the one byte field. See the Create TCTTE Extension field.

- **Create TCTTE Extension**
  Choice (YES or NO) to dynamically create an extended work area for each TCTTE. Specify NO if you want PIE/CICS to use a four-byte field set aside for PIE/CICS in the TCTUAL (user area of each TCTTE) instead. Then enter the decimal offset, relative to zero, of the beginning of the four-byte field in the TCTUAL. This number can be from 0 to 251. The default is YES.

- **Temp Storage Queue Prefixes**
  Three-character prefix that identifies all PIE/CICS temporary storage queue names. The default is @YZ.

- **Load Balancer 16 byte field**
  Decimal offset into the TCTTE user area to the starting address of a 16-byte field used by Availability PLUS to balance loads by CPU usage. Otherwise, leave the field blank. The default is blank.

- **Automatic PIE GMM Tran**
  Choice (YES or NO) to change the CICS good morning transaction code to PSGM when PIE/CICS is running. If you specify YES, PIE/CICS users are shown the Logon Director screen to sign on to CICS. When PIE/CICS is unavailable, CICS uses the current good morning transaction. If you specify NO, PIE/CICS does not change the existing CICS good morning transaction. This option is valid only in CICS version 1.7 or higher. The default is NO.
• **Message Destination ID**
  CICS Destination ID from the DFHDCT table. This field allows you to change the default PIE message log from PIEL to some other CICS Destination ID that you may already have defined in your CICS system (DFHDCT table). The default is PIEL.

• **Auto-start Network Monitor**
  Choice (YES or NO) to automatically start the Network Monitor during PIE/CICS initialization if NetGate or Availability Plus are installed. If you specify NO, you must start the Network Monitor by including program PCNMPINI in the PLTPI. You can copy and uncomment the entry from member PIEPLTPI of the CNTL dataset. Insert the entry after your PCSMPINI entry. The default is NO.

3. **Press PF8 to scroll forward to the second screen of the Customization Options utility.**

The following figure shows an example of the second Customization Options screen.

![Customization Options Screen](image)

<table>
<thead>
<tr>
<th>PIE Customization Options (page 2 of 2)</th>
<th>(C) 1995 TSC, Inc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command ====&gt;</td>
<td>Version03.03.00</td>
</tr>
<tr>
<td>Signon Message Display Time . . . . 2 (Seconds)</td>
<td>Date 03/21/1999</td>
</tr>
<tr>
<td>Auto-start Terminal Security Monitor NO</td>
<td>Julian 1999.80</td>
</tr>
<tr>
<td>Removeable Token Delimiters . . . . /_</td>
<td>APPLID TSCDB</td>
</tr>
<tr>
<td>Terminal Alarm on Cut/Paste . . . . YES</td>
<td>MVS Loaded Programs DDNAME . . . . DFHRPL</td>
</tr>
<tr>
<td>Dynamic Transaction Routing . . . . YES</td>
<td>Menu Security Re-check interval .. 60 (Minutes)</td>
</tr>
<tr>
<td>Menu Line Security Option . . . . OMIT (OMIT</td>
<td>PROT)</td>
</tr>
<tr>
<td>Limit Transactions to One Session . .</td>
<td>Terminal Alarm on Cut/Paste . . . YES</td>
</tr>
<tr>
<td>Handle Unavailable Printer in NEP . YES</td>
<td>MVS Loaded Programs DDNAME . . . . DFHRPL</td>
</tr>
<tr>
<td>Transid Validation by Session Mgr . YES</td>
<td>Dynamic Transaction Routing . . . YES</td>
</tr>
<tr>
<td>Extract VTAM Logonmsg by Logon Dir . YES</td>
<td>Multi-Language Characters . . . .</td>
</tr>
<tr>
<td>-MRO API, use 1 byte field at . . . (decimal offset in the TCT user area)</td>
<td>Generic matching on Menus/Sessions . . . . ALL (NO/MEN/SES/ALL)</td>
</tr>
<tr>
<td>Action on max sessions . . . . . . POP (SES/POP)</td>
<td>Action on session close . . . . . . LAS (BSM</td>
</tr>
</tbody>
</table>

PF: 1 Help 3 End 5 Save 6 RETRive 7 Up 8 DOwn 9 Delete

4. **Update the following fields of the second Customization Options screen.**

• **Signon Message Display Time**
  Time in seconds from 0 to 10 that messages are displayed when users signs on to PIE/CICS. The option, Display Sign-on Msg = Y, must be set in the User Directory to display sign on messages. The default is 2 seconds.

Specify a time just long enough to allow the user to read the entire message. Do not specify a longer period because the PIE/CICS signon task is suspended and using CICS storage during this period.

• **Auto-start Terminal Security Monitor**
  Choice (YES or NO) to automatically start the Terminal Security Monitor (TSM) at PIE initialization. See “Terminal Security Monitor,” on page 85 for more information about setting parameters with the Terminal Security Monitor. The default is NO.
• **Changeable Token Delimiters**
  Characters (/-_.) or blanks selected as delimiters that separate tokens in a command string. You can remove any delimiter in this field by coding a blank in place of the delimiter. PIE/CICS also uses the period (.), comma (,), and blank token delimiters; these delimiters cannot be removed. The default token delimiters are forward slash (/), hyphen (-), underscore (_), and period (.).

  If you have any transaction codes or APPLIDs that use a listed token delimiter, you must remove that delimiter by blanking it out in this field.

• **Terminal Alarm on Cut/Paste**
  Choice (YES or NO) to produce a 3270 terminal beep when a cut or paste operation is completed. The default is YES.

• **MVS Loaded Programs DDNAME**
  1 to 8-character MVS DDNAME that specifies the ddname of MVS program loads for the Network Monitor. MVS loads from the DFHRPL cause problems for some software products like DADS. If you have DADS or an equivalent product, you must define the PIE/CICS LOADLIB twice in the CICS start up JCL. Use another ddname for the second definition. Then, specify the second DDNAME in this field. The default is DFHRPL.

• **Dynamic Transaction Routing**
  Choice (YES or NO) to use Dynamic Transaction Routing. See “Dynamic Transaction Routing,” beginning on page 119 for more information. The default is NO.

• **Menu Security Re-check interval**
  Interval in minutes (0 to 999) that Dynamic Menus verifies menu lines with external security. The verification interval begins when the user initially accesses a menu. Dynamic Menus verifies menu lines after the interval completes and the user presses an AID key at the menu, or returns to the menu from another application. If security rules have changed during the interval, the menu will reflect the changes. The default is 60.

  Dynamic Menus must call external security for each item listed on a protected menu. You should specify a fairly long interval to minimize system resource usage resulting from repetitive security verifications. System overhead is particularly high for ACF2 because each call requires a CICS LINK.

• **Menu Line Security Option**
  Choice (OMIT or PROT) to display protected Dynamic Menu lines to unauthorized users. The default is OMIT.

  **OMIT**
  Protected lines are removed from the menu and not shown to unauthorized users.

  **PROT**
  Protected menu lines are displayed to unauthorized users, but the line cannot be selected. The selection field is displayed with a different 3270 attribute to indicate the field cannot be selected.

• **Menu Auto-select Single line**
  Choice (YES or NO) to automatically select the single remaining option from a Dynamic Menus screen because all other options have been omitted by menu line security. The default is NO.
• Limit Transactions to One Session
   Choice (YES or NO) to restrict transactions to a single session. Refer to "Limiting
   Transactions to One Session," on page 67 of the MultiCICS Administration Guide. The
   default is NO.

• Handle Unavailable Printer in NEP
   Choice (YES or NO) to queue CICS screen print requests if the printer is not available
   If you specify NO, the screen print is discarded, which is the CICS default. If you
   specify YES, screen print requests are queued to the printer. The default is NO.

• Tranid Validation by Session Mgr
   Choice (YES or NO) to validate PIE/CICS transaction IDs before executing a command.
   The default is YES.
   YES Transactions are validated before they are executed. Invalid CICS
   transactions result in a PIE/CICS message stating the tranid is invalid.
   NO PIE/CICS bypasses validation and passes the command to PIEEXEC.
   This permits the PUXCMMD exit to assume control and process invalid
   tranids. The sample P01CMMD, changes ==xxxx to a request for menu
   line xxxx from the user's default menu.

• Extract VTAM Logon msg by Logon Dir
   Choice (YES or NO) to permit the Logon Director to extract data from the VTAM log on
   message. The default is YES.
   YES A PIE/CICS region can extract the user ID and password from the
   VTAM log on message. Users who transferred between regions with
   either the PASS, TSWITCH, or ACCESS commands are automatically
   signed on to the destination region if the appropriate log on information
   was sent as a command parameter.
   NO Data is not extracted from the VTAM logon message. Users must sign
   on manually. NO is mandatory if you are using the VTAM Interpret table,
   which also passes data using the logon message for your terminals.

• Multi-Language Characters and MRO API, use 1 byte field
   Refer to the Custom Menus Administrator manual for an explanation of these fields.

• Generic matching on Menus/Sessions
   Generic name matching for Dynamic Menu lines or session names listed on the
   Sessions menu. The default is ALL.
   MEN Generic name matching is possible for Dynamic Menu lines.
   SES Generic name matching is possible for session names listed on the
   Sessions menu.
   ALL Generic name matching is done for the lines of Dynamic Menus and
   sessions listed on the Sessions menu.
   NO Generic name matching is not allowed.

• Action on max sessions
   Screen that appears if users attempt to create a new session when additional sessions
   are not available. The default is POP.
   POP Display the Switch pop-up menu
   SES Display the Sessions menu.
   Users receive the message “No sessions available” with either screen.
Chapter 5  Customization Options

• Action on session close
  Destination session that appears after ending an application with AutoEnd active. The default is LAS.
  LAS  Return to last used application session or the Sessions menu, whichever is the most recently used.
  BSM  Return to the most recent application session and skip the Sessions menu.
  
  The BSM option can be confusing to users if they previously accessed the Sessions menu. When they press PF3 to end a session, they return to a previous session rather than the expected Sessions menu.
  
  With either option, the final PIE command is executed instead if there is no eligible session as a destination.

5. **Press PF5 to save your changes.**

   If any field of the Customization Options screens contains an invalid value, PIE/CICS displays an error message and positions the cursor over the screen field with an error.
   
   PF3 does not save changes made to the Customization Options screens. Press PF5 to save your changes before ending the session with PF3.

6. **Press PF3 to exit from the Customization Options utility.**

7. **Cycle CICS to activate your customization options.**

**Customization Options Utility Commands**

The following commands can be submitted from the Customization Options screens.

- **HELP (PF1)**  Browse the help file associated with the Customization Options utility.
- **END (PF3)**   Exit from the Customization Options screen. END does not save changes made to the Customization Options screen.
- **RETURN (PF4)** Exit the Customization Options screen and return to the highest menu in the session stack. The highest menu may be the Sessions menu, the most recent “main” menu, or end of session. RETURN does not save editing changes made to either Customization Options screen.
- **SAVE (PF5)**   Save any changes made to the Customization Options screens and continue editing.
- **RETRIEVE (PF6)** Discard unsaved changes and refresh the screen using the current values stored in the Options dataset.
- **UP (PF7)**     Scroll backwards to page 1.
- **DOWN (PF8)**   Scroll forward to page 2.
- **DELETE (PF9)** Deletes all options set with the Customization Options screens. When you issue DELETE, a message window appears requesting a confirmation of the deletion. The next time PIE/CICS is initialized, the default options are restored from the Options file.
- **CANCEL**  Discard unsaved changes and exit from the Customization Options screen.
## Chapter 6 Security

PIE/CICS provides a extensive range of security options to protect CICS and other system resources. Some of these options are described more fully in other PIE/CICS manuals. The following table summarizes the security options available with PIE/CICS and includes a reference for more information.

<table>
<thead>
<tr>
<th>Category</th>
<th>Security Considerations</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access rules</td>
<td>Authorization to execute, browse, add, update, and delete PIE/CICS objects</td>
<td>“Object Security,” on page 75</td>
</tr>
<tr>
<td>Log on</td>
<td>Verification of Logon Director sign on with external security</td>
<td>“Logon Director Actions at User Sign On,” on page 4</td>
</tr>
<tr>
<td></td>
<td>Number of permissible sign on attempts before the terminal is locked</td>
<td>“PCSTPPVI—Sign on Verification Exit,” on page 139</td>
</tr>
<tr>
<td></td>
<td>Prevent screen stealing from dial-up ports</td>
<td>“Using the Customization Options Utility,” beginning on page 66, the Maximum logon attempt count field</td>
</tr>
<tr>
<td></td>
<td>Forced sign ons to PIE/CICS</td>
<td>“Using the Logon Director as the Good Morning Message,” on page 8, “Short Log on Messages,” on page 159</td>
</tr>
<tr>
<td></td>
<td>Control multiple sign ons per user ID</td>
<td>“Editing a User Directory Entry,” beginning on page 54, the Userid Type field</td>
</tr>
<tr>
<td></td>
<td>Automatic sign on from physically secured or public terminals</td>
<td>“Automating User Sign Ons,” beginning on page 6</td>
</tr>
<tr>
<td></td>
<td>Determine destination when users cancel the Logon Director screen; return to VTAM or the CICS sign on screen</td>
<td>“PCSTPPVI—Sign on Verification Exit,” on page 139</td>
</tr>
<tr>
<td></td>
<td>Receiving sign on messages from RACF</td>
<td>“PIE/CICS and External Security,” on page 79</td>
</tr>
<tr>
<td>Idle terminals</td>
<td>AutoLock time-out</td>
<td>“Protecting Unguarded Terminals,” on page 84</td>
</tr>
<tr>
<td></td>
<td>Terminal Security Monitor Manual LOCK TSWITCH time-out</td>
<td></td>
</tr>
<tr>
<td>Secure PIE/CICS commands, transactions, and variables</td>
<td>PIE Transactions</td>
<td>&quot;Securing PIE/CICS Transactions,&quot; on page 80</td>
</tr>
<tr>
<td>------------------------------------------------------</td>
<td>----------------</td>
<td>-----------------------------------------</td>
</tr>
<tr>
<td>PIE Application commands</td>
<td>&quot;PUXCMMD—PIE Application Command Exit,&quot; on page 148</td>
<td></td>
</tr>
<tr>
<td>PIE Environment commands</td>
<td>&quot;Editing a User Directory Entry,&quot; beginning on page 54, the Security Classes field &quot;PCSMPFIL—PIE Environment Command Exit,&quot; on page 135</td>
<td></td>
</tr>
<tr>
<td>View (special controls)</td>
<td>&quot;Restricting Access to VIEW,&quot; on page 117 of the Operation and Administration Guide</td>
<td></td>
</tr>
<tr>
<td>Controlling which variable values can be displayed—for example, disallowing display of the &amp;ZPSWD variable</td>
<td>&quot;PUXVARS—Variable Substitution Exit,&quot; on page 151</td>
<td></td>
</tr>
<tr>
<td>Dynamic Menus</td>
<td>Menu line security</td>
<td>See page 67, &quot;Defining Menu Security&quot;.</td>
</tr>
<tr>
<td>Prevent access to the CICS blank screen</td>
<td>Make PIE/CICS the good morning transaction</td>
<td>&quot;Using the Logon Director as the Good Morning Message,&quot; on page 8</td>
</tr>
<tr>
<td>Set up default application for new sessions</td>
<td>&quot;Editing a Profile,&quot; beginning on page 34, the Initial PIE Command field</td>
<td></td>
</tr>
<tr>
<td>AutoEnd all sessions</td>
<td>&quot;AutoEnd,&quot; on page 25 of the MultiCICS Administration Guide</td>
<td></td>
</tr>
<tr>
<td>Prevent users from entering transaction codes</td>
<td>Pre-defined all user sessions</td>
<td>&quot;Editing a User Directory Entry,&quot; on page 54, the Maximum Sessions field &quot;Assigning User Profiles,&quot; on page 30</td>
</tr>
<tr>
<td>Disallow the escape string</td>
<td>&quot;Editing a Profile,&quot; on page 34</td>
<td></td>
</tr>
<tr>
<td>Exits</td>
<td>Override security decisions for a number of features</td>
<td>See page 129, &quot;PIE/CICS Exits&quot;.</td>
</tr>
</tbody>
</table>
Object Security

PIE/CICS objects are secured at two levels.

- Authorization is verified against a set of access rules.
- User exits, which can override authorization set by access rules.

Access Rules

Access rules specify who can execute, browse, add, update, and delete PIE objects. This section explains how PIE object security is granted to users according to their status within PIE/CICS.

Access to a PIE object is based upon authorization granted to defined PIE/CICS user classes. Membership within a class confers certain authorization rights to manage PIE objects based upon the anticipated needs of users who belong to that class. Usually, users are assigned to a class based upon their role within a PIE/CICS system; whether it be system administrators, group administrators, or simply end-users. The following list shows the different PIE/CICS user classes:

- System administrator
- Group administrator of the group to which the object belongs
- Object’s administrator
- Object’s owner, with private object authority
- User who last updated the object

A user will have access authority to an object with any one of the four types of authorization. To control an object, you must control all four paths to it.

The following discusses each of the access authority types.

System Administrator

System administrators have complete, unrestricted access to all PIE objects. This authority supersedes any other authorization to access an object. System administrator authority is granted through the User Directory with values set on the System Admin field.

Only system administrators have the authority to manage objects with the SYSTEM Group ID. System objects include the User Directory, the Terminal Directory, and all Logon Director screens.

Group Administrator

A group administrator has unrestricted access to all PIE objects belonging to the administrator’s group; the object belongs to the same Group ID as the administrator.

Group administrator authority is granted through the User Directory with the Group Administrator field. For example, suppose a user has the following entries in the User Directory:

Groupid—SALES
Group Administrator—Y

The user is a group administrator for the SALES group and has full access to any object belonging to that group, for example, the menu SALES.MENU21, the profile SALES.USERPROF, etc.

Object Administrator

An object administrator has control over a specific object, rather than a general class or group of objects. Object authorization is granted to a user by either the system or group administrator on an object-by-object basis.

An object administrator cannot create a new object. An object administrator can only change existing objects. The object must be created by a system or group administrator before an object administrator can make any modifications to it.

An object is assigned an administrator from the object’s Administrator field. For example, suppose that a menu is defined with the following specification.

Administrator  SALES14

User SALES14 is the menu’s administrator.

You can use wildcards on the Administrator field to assign multiple administrators to an object.

Object Owner

Users are owners of their private objects. A private object is identified by the user’s ID as its Group ID. For example, if your user ID is TECH24 and there is a menu named TECH24.MAINMENU, the menu is a private object and you are the owner.

Object owners can make changes to their private objects. However, you can create private objects only if you have private object authority. Private object authority is granted through the User Directory with the Private Objects field.

Last User to Update an Object

The user who last updated an object has full authority to immediately update the object again. This access method is a preventative measure intended for rare occasions when an object has been rendered unusable by mistake. For example, if a system administrator updated the User Directory and accidentally deleted all system administrator entries, this user still has administrator authority to update the User Directory again and correct the mistake.
Authorization Summary

The following table summarizes the authorization rights granted to each class of PIE/CICS users. The table shows the actions each user class can perform with a PIE object.

<table>
<thead>
<tr>
<th>PIE/CICS User Class</th>
<th>Authorization Rights</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Execute</td>
</tr>
<tr>
<td>System administrator</td>
<td>x</td>
</tr>
<tr>
<td>Last user to update this object</td>
<td>x</td>
</tr>
<tr>
<td>Group administrator</td>
<td>x</td>
</tr>
<tr>
<td>Owner of object, Private Objects = Y or O</td>
<td>x</td>
</tr>
<tr>
<td>Owner of object, Private Objects = N</td>
<td>x</td>
</tr>
<tr>
<td>Administrator of object</td>
<td>x</td>
</tr>
<tr>
<td>Auditor of object</td>
<td></td>
</tr>
<tr>
<td>Group User (not a private object)</td>
<td></td>
</tr>
</tbody>
</table>

"There is no inquire function for menus"  

Examples of Access Authorization

Listed below are some excerpts from a User Directory.

First Entry:

```
Userid ===> TECH5  
System Admin ===> Y  
Groupid ===>  
Group Administrator ===> N
```

Second Entry:

```
Userid ===> AGENT44  
System Admin ===> N  
Groupid ===> MARKET  
Group Administrator ===> Y
```

Third Entry:

```
Userid ===> SALES25  
System Admin ===> N  
Groupid ===> MARKET  
Group Administrator ===> N
```

Here are excerpts from several menu definitions:

First menu:

```
Groupid ===> SALES  
Menu Name ===> MENU1  
Administrator ===> *
```

Second menu:

```
Groupid ===> MARKET  
Menu Name ===> MAINMENU  
Administrator ===> NONE
```

Third menu:

```
Groupid ===> SALES  
Menu Name ===> MENU2  
Administrator ===> SA*
```
For each of the these menus, consider which of the three user IDs described above can update the menu?

SALES.MENU1: All three of the users can update this menu. In fact, anyone can update it, because its Administrator field contains a single asterisk which, by definition, matches any possible user ID.

MARKET.MAINMENU: TECH5 can update this menu because TECH5 is a system administrator. AGENT44 can also update the menu because AGENT44 is the group administrator of the MARKET group.

SALES.MENU2: TECH5, a system administrator, can update this menu. SALES25 can also update it because the menu’s Administrator field contains SA*, which matches SALES25.

Access Exit

You can override the access rules with the PIE Object Authorization exit. See “PUXOBJ S—PIE Object Authorization Exit,” on page 150, for more information.
PIE/CICS and External Security

PIE/CICS supports RACF, CA-ACF2, CA-TOP SECRET, and OmniGuard security products. If your site uses another external security system, or you have written an internal security system for particular applications, you can customize PIE/CICS exits to support that security.

The following PIE exits interface with an external security system:

- **PCSTPPVI** Sign on Verification exit
- **PCXPSEC** Menu Line External Security exit
- **PCMNUL** Menu Line Access exit

The PUX1 source exits can also interface with security. For example, the standard PUX1 member, PCTTACF2, has special code to support ACF2.

Refer to the chapter titled "PIE/CICS Exits," on page 129 and the appendix titled "PIEEXEC API" of the Dynamic Menus Administration Guide for more information about authorizing security with these exits.

**ACF2**

If your site uses the ACF2 sign on appendage program, you must notify PIE/CICS of the program name. To do so, complete the following procedure:

1. **Edit member PCSTACF2 of the PIE SAMPLIB dataset.**
2. **Specify the name of the sign on appendage program at label USERLAP. The default program name is APPENDGE.**
3. **Reassemble the PCSTPPVI exit using member PCSTPPVI of the PIE CNTL dataset.**

**OMNIGUARD**

If you are using OMNIGUARD, you must add a 2048 byte transaction work area (TWA) to all PIE transactions.

**RACF**

PIE/CICS sends a sign on message to users that shows the number of remaining days until their current password expires. PIE/CICS begins sending the message when the password expiration date reaches 14 days.

PIE/CICS uses a RACF post-processing exit to generate the message. To implement the exit, do the following.

1. **If you are not currently running RACF exit ICHRIX02, assemble the RACF post-processing exit ICHRIX02 using member ICHRIX02 of the PIE CNTL dataset.** If you are already using a ICHRIX02 exit, do not run the PIE ICHRIX02 assembly job because this will replace your existing exit in SYS1.LPALIB. Instead, take the code from the ICHRIX02 sample exit in the PIE SAMPLIB and merge it into your existing exit. Then, reassemble ICHRIX02 using the JCL supplied by RACF.
2. **Specify Y in the Display Signon Msg field of the User Directory for every user who should receive the password expiration message. PIE/CICS begins sending the message after the next MVS IPL.**
Securing PIE/CICS Transactions

The following table is an alphabetic listing of all PIE transactions. Each transaction belongs to one of three possible security groups.

- **Group 1:**
  Transactions that do not require security authorization. These transactions can be executed even if the user is not signed on. This group includes only sign on transactions that interface with your current security system. They are like the CICS transaction CESN, which is never protected.

- **Group 2:**
  Transactions that are accessible to users after they have signed on to PIE.

  - **RACF**
    Define transactions to RACF with UACC(READ). In addition, for CICS version 3.1 and below, you must also specify EXTSEC=NO on the transactions’ PCT entries, or with EXTSEC(NO) on the TRANSACTION resource if RDO is used.

  - **ACF2:**
    Define transactions to ACF2’s Safelist.

  - **TOPSECRET**
    Add the transactions to the Bypass List.

- **Group 3:**
  Transactions that should be restricted to PIE system or group administrators.

<table>
<thead>
<tr>
<th>Tranid</th>
<th>Security Class</th>
<th>Description</th>
<th>PIEEXEC Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>P#EI</td>
<td>3</td>
<td>Text edit</td>
<td>EDIT</td>
</tr>
<tr>
<td>P#ER</td>
<td>3</td>
<td>Alert window</td>
<td></td>
</tr>
<tr>
<td>P#V</td>
<td>2</td>
<td>Installation Verification</td>
<td></td>
</tr>
<tr>
<td>P#LU</td>
<td>2</td>
<td>Repository Load utility</td>
<td>LOAD</td>
</tr>
<tr>
<td>P#MN</td>
<td>3</td>
<td>Invoke a menu</td>
<td>MENU</td>
</tr>
<tr>
<td>P#ND</td>
<td>3</td>
<td>User Directory detail editor</td>
<td></td>
</tr>
<tr>
<td>P#NT</td>
<td>3</td>
<td>NetGate</td>
<td>ACCESS</td>
</tr>
<tr>
<td>P#OL</td>
<td>3</td>
<td>Limit transactions message</td>
<td></td>
</tr>
<tr>
<td>P#PC</td>
<td>3</td>
<td>PROC command</td>
<td>PROC</td>
</tr>
<tr>
<td>P#PI</td>
<td>3</td>
<td>Printer change command</td>
<td>PRINTID</td>
</tr>
<tr>
<td>P#PO</td>
<td>3</td>
<td>Pop-up pseudo command</td>
<td></td>
</tr>
<tr>
<td>P#PN</td>
<td>2</td>
<td>Pseudo-termid shutdown</td>
<td></td>
</tr>
<tr>
<td>P#RN</td>
<td>3</td>
<td>Repository refresh</td>
<td></td>
</tr>
<tr>
<td>P#PR</td>
<td>3</td>
<td>Screen print</td>
<td>PRINT</td>
</tr>
<tr>
<td>P#RM</td>
<td>3</td>
<td>Run any remote tranid</td>
<td></td>
</tr>
<tr>
<td>P#RS</td>
<td>3</td>
<td>Resetuser command</td>
<td>RESETU</td>
</tr>
<tr>
<td>P#RT</td>
<td>3</td>
<td>Restart command</td>
<td>RESTART</td>
</tr>
<tr>
<td>P#ST</td>
<td>3</td>
<td>Session Menu</td>
<td>SM SESSIONS</td>
</tr>
<tr>
<td>P#SX</td>
<td>2</td>
<td>PIE control block display</td>
<td>STORAGE</td>
</tr>
<tr>
<td>Tranid</td>
<td>Security Class</td>
<td>Description</td>
<td>PIEEXEC Command</td>
</tr>
<tr>
<td>--------</td>
<td>----------------</td>
<td>--------------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>P #S1</td>
<td>3</td>
<td>Sample Applications</td>
<td></td>
</tr>
<tr>
<td>P #TD</td>
<td>2</td>
<td>Terminal directory edit</td>
<td>TERMDIR</td>
</tr>
<tr>
<td>P #TM</td>
<td>3</td>
<td>Terminal Security Monitor control</td>
<td></td>
</tr>
<tr>
<td>P #TR</td>
<td>3</td>
<td>Trace command</td>
<td>TRACE</td>
</tr>
<tr>
<td>P #UD</td>
<td>2</td>
<td>User Directory edit</td>
<td>USERDIR</td>
</tr>
<tr>
<td>P #XE</td>
<td>3</td>
<td>REXX main controller</td>
<td></td>
</tr>
<tr>
<td>P #XP</td>
<td>3</td>
<td>REXX script processor</td>
<td></td>
</tr>
<tr>
<td>P #XS</td>
<td>3</td>
<td>REXX script application suspend</td>
<td></td>
</tr>
<tr>
<td>P #XX</td>
<td>3</td>
<td>REXX top level controller</td>
<td>REXX</td>
</tr>
<tr>
<td>PAFU</td>
<td>2</td>
<td>Profile edit</td>
<td>PROFILE</td>
</tr>
<tr>
<td>PAMU</td>
<td>2</td>
<td>Menu edit</td>
<td></td>
</tr>
<tr>
<td>PAPP</td>
<td>3</td>
<td>Logon panel editor</td>
<td></td>
</tr>
<tr>
<td>PAPU</td>
<td>2</td>
<td>Logon panel edit</td>
<td></td>
</tr>
<tr>
<td>PATU</td>
<td>2</td>
<td>Text edit</td>
<td></td>
</tr>
<tr>
<td>PBAL</td>
<td>3</td>
<td>Load balance sample transaction</td>
<td></td>
</tr>
<tr>
<td>PBRO</td>
<td>1</td>
<td>Browse text</td>
<td>BROWSE</td>
</tr>
<tr>
<td>PBUL</td>
<td>3</td>
<td>Invoke bulletin menu</td>
<td>BULLETIN</td>
</tr>
<tr>
<td>PCDP</td>
<td>1</td>
<td>Pass command</td>
<td>PASS</td>
</tr>
<tr>
<td>PEED</td>
<td>1</td>
<td>Generalized line editor</td>
<td></td>
</tr>
<tr>
<td>PEXE</td>
<td>3</td>
<td>PIEEXEC</td>
<td></td>
</tr>
<tr>
<td>PEXX</td>
<td>3</td>
<td>PIEEXEC pseudo-converse tranid</td>
<td></td>
</tr>
<tr>
<td>PGED</td>
<td>3</td>
<td>Profile detail editor</td>
<td></td>
</tr>
<tr>
<td>PIE</td>
<td>3</td>
<td>Sessions commands</td>
<td>SM</td>
</tr>
<tr>
<td>PINO</td>
<td>2</td>
<td>Installation options (password)</td>
<td></td>
</tr>
<tr>
<td>PLOG</td>
<td>3</td>
<td>Logmode change</td>
<td>LOGMODE</td>
</tr>
<tr>
<td>PLOK</td>
<td>3</td>
<td>Lock command</td>
<td>SM LOCK</td>
</tr>
<tr>
<td>PLSO</td>
<td>2</td>
<td>Object list display</td>
<td>LIST</td>
</tr>
<tr>
<td>PMND</td>
<td>3</td>
<td>Menu detail edit</td>
<td></td>
</tr>
<tr>
<td>PMZT</td>
<td>3</td>
<td>Funny display</td>
<td></td>
</tr>
<tr>
<td>PNAM</td>
<td>2</td>
<td>Network Monitor Commands</td>
<td></td>
</tr>
<tr>
<td>PNZT</td>
<td>2</td>
<td>Netmizer stats</td>
<td></td>
</tr>
<tr>
<td>POPU</td>
<td>3</td>
<td>Pop-up menus</td>
<td></td>
</tr>
<tr>
<td>PPER</td>
<td>3</td>
<td>Perform command</td>
<td>PERFORM</td>
</tr>
<tr>
<td>PSCR</td>
<td>3</td>
<td>Script command</td>
<td>SCRIPT</td>
</tr>
<tr>
<td>PSEN</td>
<td>2</td>
<td>SuperMessage send panel</td>
<td>SEND</td>
</tr>
<tr>
<td>PSGM</td>
<td>1</td>
<td>Logon Director</td>
<td></td>
</tr>
</tbody>
</table>
Changing a Site Security System

During installation, PIE/CICS exits are assembled based upon the security system running at your site. If your site changes to another security system, these exits must be reassembled.

Complete the following procedure if your site changes its external security system.

1. **Edit members PCSTPPVI, PCTTPUX1, and PC@XPSEC of the CNTL dataset.**
2. **Update the SECURITY parameter in each member.**
   - Specify RACF, TOPS, ACF2, OMNI, or NONE. If you run ACF2 or OMNIGUARD, also edit the SECMAC parameter and specify the name of your security system macro library.
3. **Run the job to reassemble the exits.**

### 3.6.0 Tranid | Security Class | Description | PIEEXEC Command
---|---|---|---
PSND | 1 | SuperMessage background sender | |
PSNY | 3 | SuperMessage deliverer | |
PSTG | 2 | PIE storage stats | |
PSWC | 3 | Switch transaction | |
PSWH | 3 | Tswitch command | TSWITCH
PSXM | 1 | Force Logon Director next tranid | |
PSYS | 3 | Customization Options screen | |
PTSQ | 2 | TSQ utility | |
PUSR | 3 | User data update | USERDATA
PVCU | 3 | CUT & PASTE | SM CUT
PVVC | 3 | View capture | SM VIEW
PVVS | 3 | View session menu display | |
Password Reverification

PIE/CICS re-verifies user IDs and passwords when users enter their passwords at the Lock screen or port (force reconnect) their sign ons from another terminal. PIE/CICS compares the user ID and password with data retained from the user’s initial sign on.

The comparison is made between either PIE/CICS sign on data or the site security system depending upon the version of CICS and security system used at your site. PIE/CICS performs password re-verification checks as follows:

<table>
<thead>
<tr>
<th>Security System</th>
<th>CICS 2.1 or 3.x</th>
<th>CICS 4.1 and Above</th>
</tr>
</thead>
<tbody>
<tr>
<td>RACF</td>
<td>PIE/CICS sign on data</td>
<td>RACF</td>
</tr>
<tr>
<td>TOP SECRET</td>
<td>PIE/CICS sign on data</td>
<td>TOP SECRET</td>
</tr>
<tr>
<td>ACF2</td>
<td>ACF2</td>
<td>ACF2</td>
</tr>
</tbody>
</table>
Protecting Unguarded Terminals

Users frequently leave their terminals without signing off PIE/CICS. An unauthorized user could potentially gain access to a system through this terminal. PIE/CICS provides a set of related features that minimize the potential threat to site security posed by unguarded terminals:

- AutoLock
- Manual Lock
- TSWITCH time-out
- Terminal Security Monitor

The remainder of this chapter explains how to prepare each of these features to protect your system from intrusions through an unguarded terminal.

AutoLock

AutoLock disables further input from a terminal after it has been idle (that is, the user has not pressed an AID key) during a specified period. When AutoLock occurs, there is no visible change to the terminal. However, the next time an AID key is pressed, the request is not processed. Instead, the Lock screen appears on the terminal, as shown in the following figure. You specify the permitted idle period for each user with the Lock Timeout field of the User Directory.

When the Lock screen appears, users simply enter their password and their sessions are restored. No work is lost. (Retry attempts on the screen are controlled by your external security retry limit.)

You can specify longer time-out periods for particular applications using the Time-out Exit. See “PCSSPTOE—Time-out Exit,” on page 137, for details.

For information on the Lock Timeout field, see "Editing a User Directory Entry," beginning on page 54.
Manual Lock
If users know they will be away from their terminals for an extended period, they can lock their terminals by entering the LOCK command. Refer to the LOCK of the Command Reference or more information.

TSWITCH Time-Out
A TSWITCH time-out changes the action of a TSWITCH transfer to another region. Normally, TSWITCH disconnects users from their originating region and sends them to a target region. On return to the originating region, TSWITCH automatically reconnects users to their existing work.

If users have not returned to the originating region within a specified period of time, TSWITCH does not reconnect to the originating region. Instead, users are sent to the Logon Director screen to sign on to PIE/CICS again.

Use the TSWITCH Auto-Signon Timeout field of the Customization Options screen to set the time-out interval. See “Using the Customization Options Utility,” beginning on page 66 for more information.

Terminal Security Monitor
The Terminal Security Monitor works in conjunction with AutoLock. Some terminals may be idle for extended periods. In that case, the Terminal Security Monitor disconnects users from PIE/CICS and CICS and returns their terminal to VTAM. This clears the screen of potentially sensitive data. It also frees CICS resources, such as virtual storage (temporary storage queues, DSA storage, Aux Temp VSAM storage) and the terminal itself. When users sign on again, they resume their work in progress.

Because the Terminal Security Monitor takes stronger action than AutoLock, you may want to set a longer time-out period than your AutoLock period.

There are several conditions that affect the ability of the Terminal Security Monitor to disconnect users.

Disconnect/Log Off
The Terminal Security Monitor may log off some users rather than disconnecting them from PIE/CICS. If they are logged off, their unsaved session work may be lost.

Users are logged off by the Terminal Security Monitor if the value ALLOW DISC=N is specified in their User Directory. However, ALLOW DISC=N is overridden if the user has one or more open sessions that are defined as non-cancellable. See “Closing and Cancelling Sessions,” on page 55 of the MultiCICS Administration Guide for instructions to disconnect from non-cancellable sessions.

Excluding Users and Terminals from Being Disconnected
You can exclude specific users and terminals from being disconnected by the TSM. To exclude a user, specify N in the TSM Timeout field of the User Directory.

You can also use the PUXINIT exit to prevent a user from being disconnected by the TSM. The PUXINIT exit verifies several conditions before permitting the TSM to disconnect a user.
The PUXITSME flag of the PUXINIT exit controls the operation of the TSM. Based upon the outcome of tested conditions, the PUXITSME flag can be set to different values to control the operation of the TSM. The following example shows several assembler statements added to the PUXINIT exit. This example excludes system administrators from being disconnected by the TSM.

```assembly
    CLI PUXISCTY,C'Y'            IS USER SYSTEM ADMIN?
    BNE PUXRESUM                ...NO, EXIT
    MVI PUXITSME,C'Y'           ...YES, OMIT FROM TSM TIMEOUT
    B     PUXRESUM                EXIT
```

### Running the Terminal Security Monitor

The operation of the TSM is controlled by the P#TM transaction, or by values entered on selected fields of the Customization Options screen. You can issue the P#TM transaction from a PIE terminal, the MVS console, or a CRLP terminal. Also, you can issue the P#TM transaction before or after PIE/CICS is initialized.

To set the operating conditions of the Terminal Security Monitor, you must first execute the P#TM transaction with its parameters. After that, you start and stop the TSM by entering the P#TM transaction again with either the START or STOP parameters.

The following syntax diagram shows the parameters of the P#TM transaction. The options you choose for the parameter are stored in the PIE Options file. You can change the options after the monitor is started and they are saved to the Options file.

```
P#TM IDLE=mmmm,SCAN=mmmm, 
    LOG=ALL,SUM,NO, AUTOSTRT=YES,NO
```

- **IDLE**
  - Idle time in minutes. PIE/CICS users who have not pressed an AID key within this interval are disconnected from CICS at the next scan time. There is no default.

- **SCAN**
  - Scan interval in minutes. The TSM is timer driven. After the TSM is started, it scans all PIE terminals at this interval and disconnects users who have exceeded their idle time. There is no default.

- **LOG**
  - Message logging to the PIEL log. Specify one of the following:
    - **SUM**
      - Logs a summary message at each scan interval indicating how many users were disconnected.
    - **ALL**
      - Logs a summary message and a message for each disconnected user (USERID/TERMID).
    - **NO**
      - Does no logging. However, initialization and termination of the TSM are logged, as are any severe errors. This is the default.
AUTOSTRT Choice (Yes or No) to start the Terminal Security Monitor automatically at PIE initialization. This option can be also set with the Customization Options Utility.

YES Terminal Security Monitor starts automatically at PIE initialization.

NO Terminal Security Monitor does not start at PIE initialization. The default is NO.

Starting the TSM

After setting operational conditions with P#TM parameters, you must start the Terminal Security Monitor. The TSM can be started manually or automatically during PIE/CICS initialization.

Enter the P#TM transaction with the START parameter to manually start the Terminal Security Monitor.

P#TM START

TSM remains active until PIE/CICS is shut down or stopped by the P#TM transaction. Specify YES on the Auto-start Terminal Security Monitor field of the Customization Options screen to start the TSM automatically during PIE/CICS initialization. Setting this field to YES is equivalent to specifying the AUTOSTRT=YES parameter of the P#TM transaction. If you execute the P#TM transaction after setting this field and you use the AUTOSTRT parameter, whatever you specify for the parameter will be copied to this field.

Stopping the TSM

Enter the P#TM transaction with the STOP parameter to stop the Terminal Security Monitor:

P#TM STOP
Chapter 7  Transparent Mode

Occasionally, all PIE/CICS features are not necessary at some sites. Usually, these sites want a selected number of PIE features without the incumbent training issues that result from offering the complete range of product options to their user community. For example, the following list is a minimal combination of PIE/CICS features that enhance user’s CICS productivity without requiring significant training:

- NetMizer optimization
- NonStop CICS support
- Two sessions and a toggle key
- AutoLock

PIE/CICS can run in transparent mode using only this combination of features; users are not aware that PIE/CICS is running concurrently with CICS.

Depending on your site requirements, the following PIE/CICS features or functions can be made transparent or unavailable to users:

- Logon Director sign on screen
- Sessions menu
- PIE pop-up menus
- System end keys
- PIE AutoLock screen
- Multiple sessions
- PIE Global keys

This chapter describes procedures to implement transparent PIE/CICS features. The chapter concludes with several examples that show how selected features are typically implemented and made transparent or restricted to certain user groups.
Logon Director Screen

In transparent mode, the Logon Director sign on screen and any sign on error messages should not be seen by users when they sign on to their system. Your site’s existing sign on procedures are used instead.

There are two ways to log on to CICS transparently through PIE/CICS:

- **Everyone can sign on to PIE/CICS using a single default ID.** PIE/CICS can execute your current CICS sign on program, or pass the user to a CICS blank screen to sign on manually.
  
  To accomplish this, run PSGM as the good morning transaction (see “Using the Logon Director as the Good Morning Message,” on page 8). Then, complete the procedures described in “Pre-defined Sign On Data to Bypass the Logon Director Screen,” on page 12.

- **You can have everyone sign on at your current sign on screen and pass sign on data to PIE/CICS through the Logon Director API.** With this method, PIE/CICS knows each user’s actual ID and password. PIE/CICS can pass this sign on data automatically to other applications.
  
  Read “Receiving sign on Data From Another Application,” on page 8 for instructions to pass data from your sign on screen to the Logon Director.

Sessions Menu

You must assign a session that opens automatically after sign on is complete to restrict users from accessing the Sessions menu. Edit the user’s profile and code OPEN x on the Initial PIE command field of the main Profile Utility screen, where x is the session number of an available user session.

You must also prevent the escape string from being used. Otherwise, users can switch to the Sessions menu using a PIE pop-up menu or the ==SESsion command. Edit the user’s profile and blank out the Escape String field of the Profile Terminal screen.

See “Editing a Profile,” beginning on page 34 for more information.

PIE Pop-Up Menus

Again, to restrict user access to PIE pop-up menus, you must make the escape string unavailable. Edit the user’s profile and blank out the Escape String field of the Profile Terminal screen,

See “Editing a Profile,” beginning on page 34, for more information.

PIE Commands

You can restrict the use of PIE Application commands with the PUXCMMD exit. Also, you can restrict the use of PIE Environment commands by limiting only security classes in the Security Class field of the User Directory. For example, to prohibit use of the CREATE command, omit security class 3. That is, specify 0124567 on the Security Classes field of the User Directory.

The PIE/CICS Command Reference lists the security class for each command.
System End Keys

Users press a system end key from the CICS blank screen to end the session and return to a PIE/CICS menu. If you want PIE/CICS to be transparent, you must make the system end keys unavailable.

Edit the user's profile. Blank the System End Keys fields of the Profile Terminal screen. See “Editing a Profile,” beginning on page 34, for more information.

AutoLock

AutoLock is a convenient feature, but it displays a different screen that can be confusing to inexperienced users:

```
PIE - Terminal locked, confirm password for: PRD1
Lock entered at: 12:48:26
Enter password and press enter to resume work, or
Press PF3 or PF15 to disconnect terminal
```

If you want to eliminate the AutoLock screen, specify 0 on the Lock Timeout field of the User Directory for each user entry.

See “Protecting Unguarded Terminals,” on page 84, for more information on AutoLock and “Editing a User Directory Entry,” beginning on page 54, for more information on the Lock Timeout field.

Disconnected Sessions

The disconnect feature allows users to resume their work again within a session after the terminal had been disconnected. Again, the disconnect feature is convenient, but you may want to prevent it from being used.

Edit the User Directory entry for each user that you want to prevent from using the disconnect feature. Specify NO in the Allow Disc Status field.

You may want to specify YES for the Allow Session Cancel field. If the user's terminal is accidentally disconnected, PIE/CICS cancels all sessions for this user. When the user logs back on, all work is lost that had not been saved prior to the terminal disruption.
Chapter 7  Transparent Mode

Logging Off
If you want users to continue to log off their sessions with the CESF transaction, you have two options:

- You can have PIE/CICS end all sessions automatically when the user enters CESF from any session.
  Make the following changes in the User Directory:
  Allow Session Cancel—Y
  Then remove CESF from the ATTACHX table.
- You can force the user to enter CESF from each session. When the user exits from their last session, have them log off PIE/CICS and CICS.
  Make the following changes to the User Directory entry:
  Allow Session Cancel—N
  In the user's profile, specify:
  Final PIE command—EXIT LOGOFF or EXIT SIGNOFF
  The CESF transaction must remain in the ATTACHX table.


Multiple Sessions
Transparent mode can be used to limit the number of available PIE/CICS sessions for specific users. Edit each user's User Directory entry and specify a number in the Maximum Sessions field. The number represents the maximum number of concurrent sessions available to this user.

See “Editing a User Directory Entry,” beginning on page 54, for more information.

PIE Global Keys
PIE global keys can be assigned the limited features you want available to users in transparent mode. For example, you can assign the SWITCH NEXT command to a PIE global key if you want users to have two sessions with CICS blank screens. Users can toggle between both sessions with the key. If you want users to be able to lock their screens, you can assign the LOCK command to a PIE global key.

PF key definitions are specified from each user's Profile Keys screen. See “Editing a Profile,” beginning on page 34, for more information about assigning commands or functions to PF keys.

View
The PIE/CICS View facility allows authorized users to display another user's current screen on their own terminal. This is a particularly useful feature for system administrators who need the capability to remotely diagnose user problems. View authority is automatically granted to system and group administrators. If you want to change that authority, see “Restricting Access to VIEW,” on page 117 of the Operation and Administration Guide.
Examples of Implementing Transparent Mode

This section provides several examples that show how to implement selected MultiCICS features. Examples of implementing specific features of NonStop CICS and NetMizer transparently can be found in each product’s manual.

The examples implement PIE/CICS with:

- Two blank screens
  Users switch between screens with a PF key. This method signs users on to PIE/CICS with default user IDs.
  A variation implements PIE/CICS with multiple pre-defined sessions.
- AutoLock only
  This method signs users on to PIE/CICS by passing sign on data from your current sign on screen.
  A variation implements PIE/CICS with one session running a dynamic menu.
- View only

Example 1: Implementing Two Blank Screens

This example describes how to specify two CICS blank screens. Users can run separate applications in each session by switching between them with a PF key. All other PIE features are transparent.

Currently, users sign on to CICS with the CESN transaction. As far as PIE/CICS features go, you could have users sign on to CICS first and pass their sign on data to PIE/CICS, or you could set up the Logon Director to accept a default user ID and password automatically, then have them sign on to CICS using your current sign on screen. However, because you use CESN, not a homegrown sign on system, you cannot add the Logon Director API to the program. You must implement the default user ID log on.

In the following procedure, you will go to the User Directory and Profile Utility several different times. The procedure is broken into functional sections that explain how different results are achieved.

First, you must prepare the Logon Director to accept a default sign on. (See “Logon Director Screen,” on page 90.)

1. Edit your User Directory.
2. Specify system administrator authority to a single terminal: CICS terminal ID L001.
   If you have installed a new PIE/CICS system, your User Directory should have two entries with * as the user ID. The first is the system administrator entry.
4. Edit this entry and give it the user ID PIEUL001 (remember that L001 is the system administrator's terminal).
   Make sure the Password field contains an asterisk (*) and the System Administrator field is Y.
5. **Modify the remaining * entry. Specify the user ID PIEU*.**

   This entry accepts sign ons from all other terminals. Make sure the Password field contains an asterisk (*) and the System Administrator field is N.

   Make note of the profile used for both entries.

6. **Press PF3 to save your changes and exit the User Directory.**

7. **Go to the Terminal Directory Utility.**

   All terminals should have pre-defined sign ons. You need only one terminal entry, the wildcard * entry.

8. **Specify PIEUSER in the Userid field.**

9. **Specify any value for the Password field.**

10. **Specify (NONE) on the Logon field to bypass the Logon Director screen.**

    With these User and Terminal Directory entries, PIE sign on is automatic for all terminals. The user will be signed on with the ID PIEUterm, where term is the terminal’s CICS term ID. When the system administrator terminal, L001, is turned on, that terminal will be granted system administrator authority. Anyone on that terminal can update PIE objects.

    The section “Pre-defined Sign On Data to Bypass the Logon Director Screen,” on page 12, tells you to make the Logon Director the good morning transaction at this point. We will put off that step until our customization is complete. That way, users will not get PIE/CICS until you are ready.

    While PIE sign on is now completely automatic, sign on to CICS is not. You want users to sign on as before: from your current CICS sign on screen which appears when the terminal accesses CICS. To set this up, you do the following:

11. **Access the Profile Utility and create or edit the profiles specified in your User Directory entries. (By default they are ACCOUNT and USER.)** The profiles should have SYSTEM as their Group ID.

    In the main Profile screen, the Initial PIE Command field, specify OPEN 1. Then on the Profile Sessions screen, in the session 1 Execute field, type CESN. (Once the user is signed on to PIE/CICS, PIE/CICS will switch the user to session 1, which will execute CESN. When users sign on, they will go to the CICS blank screen.)

    For each profile, press PF5 to save your changes and PF3 to exit.

12. **Now modify the Attach table to bypass transactions CESN, CESF, CSSN, and CSSF. You will reassemble the Attach table using our transparent mode sample.**

    Edit member PG@RPCSC of the PIE SAMPLIB. Change the statement—

    ```
    COPY ATTACHX
    to—
    COPY ATTACHN
    ```

    Then re-assemble the table using member PG@RPCSC of the PIE CNTL library.

13. **Prevent the Logon Director from signing the user on to external security.** Edit members PCSTPPVI, PCTTPUX1, and PC@XPSEC of the CNTL dataset. In each exit, specify NONE for the SECURITY parameter. Then run the jobs.

    Now it is time to set up your two sessions and the PF key to toggle between them.
14. **Return to the Profile Utility and edit the profiles specified in your User Directory entries.**

15. **Go to the Profile Sessions screen.**

   You have already defined session 1. Now define session 2 to access the CICS blank screen. In the session 2 Execute field, type SYSTEM.

   All other Sessions screen fields can remain blank. (Even the Name and Title fields can be blank for sessions 1 and 2. Both Name and Title are displayed on the Sessions menu and the name is used with PIE/CICS commands. However, since neither the Sessions menu nor the PIE commands will be available to your users, there is no need to specify them.)

16. **Go to the Profile Keys screen.** Press PF8 to go to the second screen. In the PF24 field, specify **SWITCH NEXT**. When users press PF24, they toggle between sessions. (If you had more than two sessions, SWITCH NEXT would cycle through all sessions.)

   Make sure all other PF key fields are blank.

   Press PF5 to save your changes and PF3 several times to exit from the Profile Utility.

17. **Return to the User Directory.** For both entries, go to the detail screen and specify **2** in the Maximum Sessions field.

   The remainder of the procedure makes other PIE features transparent to the user.

18. **Enter the values shown with the following fields of the User Directory for both entries:**

   - **Lock Timeout**—0
     This disables AutoLock and the Lock Time-out screen.

   - **Allow Disc Status**—N
     This prevents users from disconnecting from their sessions.

   - **Allow Session Cancel**—Y
     This allows PIE/CICS to cancel an active session if the terminal is accidentally disconnected, or the user enters CESF to log off.

   When users enter CESF LOGOFF, PIE/CICS does not interfere. The user is logged off CICS and PIE/CICS cancels any open sessions. For this feature to work, the CESF transaction must be removed from the ATTACHX table. You took care of that earlier when you reassembled the table with COPY ATTACHN.

19. **Go to the Profile Utility and select the Terminal option screen.**

20. **Specify blanks in the following fields of the Terminal screen:**

   - **Pie Escape String** field
     This prevents users from displaying PIE pop-up menus and executing PIE commands

   - **Both SYSTEM END key** fields
     This prevents users from ending their sessions by pressing a system end key)
21. For each profile, press PF5 to save your changes and PF3 to exit.  
   Now your PIE/CICS system is set up. You can make it available to all users.

22. Set up the Logon Director as your good morning transaction. Go to the Customization Options screen. Change the Automatic PIE GMM Tran field (on the second screen) to YES.

23. Press PF5 to save your changes and PF 3 to exit.

A simple variation of this procedure pre-defines several sessions. You need to change the Sessions screen on both profiles and the Maximum Sessions field in the User Directory:

- Specify the number of active sessions you want available to users on the Maximum Sessions field of the User Directory.
- Specify a CICS transaction on the Execute field of the Profile Sessions screen for each session. See page 23, “Session Execution String”, in the MultiCICS Administration Guide for more information.

Users can switch between sessions with the PF24 key, which has been previously assigned the SWITCH NEXT command.

Example 2: AutoLock Only

Unattended terminals are a potential security risk. In this example, you need the additional security provided by the PIE/CICS AutoLock feature. Your site also needs the PIE/CICS disconnect/reconnect feature that retains user work if a terminal is accidentally disconnected from CICS. You want both PIE/CICS features, but no others.

PIE/CICS must know each users’ ID and password to enable AutoLock. It must also interact with your external security system. As a result, you cannot use the log on method described in the first example. With that method, PIE/CICS knows only a default user ID. In this example, users sign on with their current sign on program and the data is passed to PIE/CICS.

First modify your sign on program to pass sign on data to the Logon Director. (See “Receiving sign on Data From Another Application,” on page 8, for more information.)

1. Edit your sign on program.

2. Using the member SIGNON of the PIE SAMPLIB as sample code and following your program’s sign on procedures, add code to write the user ID and password to a temporary storage queue.

   Use @YZLterm as the name of the temporary storage queue, where term is the CICS terminal identifier.

   In item 1 of the queue, write the user ID and password in the format:
   
   userid/password
   (Code the slash as shown. Follow the slash immediately with the password. Do not code any intervening blanks.)

3. Transfer control from your program to the Logon Director: XCTL to program PCTSPGMM.

   Now set up AutoLock and disconnecting.

4. Edit the User Directory.
Examples of Implementing Transparent Mode

Your PIE/CICS system is new and has two entries with a user ID of *. The first is customized for a system administrator. The second is customized for a regular user.

5. **Edit the first entry.**

6. **Type your user ID in the Userid field. Edit all other fields in the entry as you like.** *(The System Administrator field must specify Y.)*

7. **Edit the second entry.**

   Go to the detail screen and specify the following:
   
   **Lock Timeout**—15. Users terminals will be locked when they do not press an AID key for 15 minutes. Their work will be saved, including all unsaved data, but no one will be able to use the terminal until the correct password is entered.

   **Allow Disc Status**—Y. Users will be disconnected if their terminals are disconnected from CICS. When they sign on again, they will be placed where they left off.

   Now make all other PIE/CICS features transparent.

8. **Still in the User Directory, specify the following:**

   **Maximum Sessions**—1 (to limit users to a single session)

9. **Go to the Profile Utility.**

10. **Create or edit the profile specified in the User Directory * entry. The profile should have SYSTEM as its Group ID.**

    First set up the users’ session and what will happen when users sign on.

    In the main Profile screen, the Initial PIE Command field, specify OPEN 1. This will send users directly to their session.

    Go to the Profile Sessions screen. In the session 1 Execute field, type SYSTEM. This will put user at the CICS blank screen.

11. **Go to the Terminal screen.**

    Specify blanks for:

    • the Pie Escape String field (to prevent users from seeing PIE pop-up menus and executing PIE commands)
    • both SYSTEM END key fields (to prevent users from ending one of their sessions by pressing a system end key)

    For each profile, press PF5 to save your changes and PF3 to exit.

    Now your PIE/CICS system is set up. You can make it available to all users. Set up the Logon Director as your good morning transaction.

12. **Go to the Customization Options screen.**

13. **Change the Automatic PIE GMM Tran field (on the second screen) to YES.**

14. **Press PF5 to save your changes and PF 3 to exit.**

   An easy variation on this implementation would set up a dynamic menu in session 1. Users could select applications from their menu. They would run in a single session and have no other PIE/CICS features available. To do this, change the session 1 Execute field to MENU MAIN menu-name. *(You may set the Lock Timeout field to 0 and the Allow Disc Status field to N, if you like.)*
Example 3: View Only

This example restricts access to the PIE/CICS View facility to supervisors and helpdesk personnel. All other users are prohibited from using View.

This example restricts access to View by preparing separate PIE/CICS groups for users and administrators. Each group is granted specific access privileges to PIE/CICS facilities. You can set up user groups by associating entries within the User Directory to your security system database. This example uses RACF.

Each users’ ID must be known to PIE/CICS to assign View authority correctly. In this example, users sign on through a locally designed program that passes sign on data to PIE/CICS.

First set up sign on using the Logon Director API. The steps are exactly the same as for “AutoLock,” on page 91.

Prepare separate groups for users and administrators.

1. **Edit the User Directory.**
   
   In this example, PIE/CICS has been recently installed and the User Directory has two wildcard entries (*). This example customizes the first entry for system administrators. The second entry is customized for a regular user.

   1. **Edit the first entry.**
   2. **Type your user ID in the Userid field.**
       
       Edit all other fields in the entry as you like. (The System Administrator field must specify Y.)
   3. **Create a new entry for your first group administrator.**
   4. **Enter R in the line command field of the new entry.**
       
       You will have two entries. Edit the first. Change the following fields:
       
       Userid—the administrator’s user ID
       
       Group Administrator—Y
       
       (You do not have to specify a Group ID. It will be specified automatically through RACF.)
       
       Repeat and change this entry to define all your group administrators.
   5. **Edit the remaining * entry.**
   6. **Go to the detail screen and specify an ID in the Helpdesk field.**
       
       You may use wildcards. If your help desk personnel do not have similar user IDs, you can grant them View authority using the View Screen Authorization exit, PUXVSCN. System administrators and group administrators for this group have View authority for this user no matter what you put in this field. To prohibit access for administrators, use PUXVSCN.
       
       To get Group names from RACF, you will reassemble the Post Signon Initialization exit, PUXINIT, using sample exit P04INIT of the macro library. P04INIT is already set up for this purpose.
   7. **Edit member PCTTRACF of the SAMPLIB dataset.**
   8. **Change COPY PUXINIT to COPY P04INIT and save your changes.**
9. Edit member PCTTPUX1 of the CNTL dataset to reassemble the exit.
   Verify the SECURITY parameter specifies RACF.

10. Check other parameters to ensure they are correct for your site and run the job.
    The remaining steps of this procedure make all remaining PIE/CICS features transparent to
    the user.

    For the * entry, especially, and for any other entry, specify the following:
    
    Lock Timeout—0 (to prevent the lock screen)
    Allow Disc Status—N (to force users off if their terminal disconnects from CICS)
    Allow Session Cancel—Y (to allow PIE/CICS to cancel an active session if the terminal
    is accidentally disconnected)
    Maximum Sessions—1 (to limit users to a single session)

12. Go to the Profile Utility.

13. Create or edit the profile specified in the User Directory * entry.
    The profile should have SYSTEM as its Group ID.
    Set up the users’ session and what will happen when they sign on.
    In the main Profile screen, specify OPEN 1 on the Initial PIE Command field. This
    command sends users directly to their single session.

14. Press PF5 to save your changes and PF3 to exit.

15. Go to the Profile Sessions screen.

16. Type SYSTEM in the session 1 Execute field.
    This places the user at the CICS blank screen.

17. Go to the Terminal screen.

18. Specify blanks for:
    • the Pie Escape String field (to prevent users from seeing PIE pop-up menus and
      executing PIE commands)
    • both SYSTEM END key fields (to prevent users from ending one of their sessions by
      pressing a system end key)

19. Press PF5 to save your changes and PF3 to exit.
    Now your PIE/CICS system is set up. You can make it available to all users. Set up the
    Logon Director as your good morning transaction.

20. Go to the Customization Options screen.

21. Change the Automatic PIE GMM Tran field (on the second screen) to YES.

22. Press PF5 to save your changes and PF 3 to exit.
Chapter 8 Multiple CICS Regions

This chapter explains how to customize PIE/CICS to run under the following CICS configurations:

- MRO/ISC
- Multiple stand-alone systems
- Multiple versions of CICS
MRO/ISC Support

PIE/CICS can be customized to provide MRO/ISC support. Using MRO/ISC, only a TOR needs access to the PIE Repository; AORs do not require access to the Repository. All regions need access to the Options file.

You can share an Options file between regions. In this configuration, the TOR needs read/write access to the file and the AORs need read-only access. The following figure shows how read/write access is configured between these regions and PIE/CICS files:

This configuration may not be desirable or even possible at some sites. For example, if the TOR and AORs are in different physical locations, you cannot share the Options file. In this case, each region must have its own Options file with read-write access. The figure below shows remote regions, each with their own personal Options file.

Procedures for both configurations follow.
Sharing the Options File Between Regions
1. Install PIE/CICS into a TOR.
2. Complete the procedures described in the PIE/CICS Installation Guide, “Step 6: Add MRO/ISC Support,” on page 12, for each AOR.

Using Separate Options Files for Each Region
1. Install PIE/CICS into a TOR.
2. Create a new VSAM Options file for each AOR.
   Make a copy of the DEFVSAM member of the CNTL dataset and edit the copy.
   You may want to REPRO information from your existing Options files to reduce the amount of PIE/CICS customization required in the new region. To do so, specify the high-level qualifiers of the existing dataset (the datasets to copy from) for the Q1 and Q2 substitution parameters. Then, check the OPTSRCE DD statement to see that it correctly refers to your existing Options file. Specify the new dataset name in the IDCAMS statements at the end of the job. Also, specify the volume and space allocation for the new dataset.
   If you want to create a new, default dataset, rather than copy from the existing Options file, specify the high-level qualifiers for the new dataset in the Q1 and Q2 substitution parameters. You can leave the default dataset name in the OPTSRCE DD statements. Specify the new dataset name in the IDCAMS statements at the end of the job. Also, specify the new volume and space allocation.
   Comment out the IDCAMS statements referring to the Repository file.
   Run the job after the changes have been made.
3. Complete steps 4 and 6 through 9s of the PIE/CICS Installation Guide in the AOR.
   For each CSD, you must perform all CICS table changes, including running the PIERDO job. If your site shares CSDs, you can omit the procedure described in “Step 4: Add Entries to CICS Tables,” on page 6 of the PIE/CICS Installation Guide.

Initializing PIE in an AOR
We recommend that you start PIE/CICS from the PLTPI in each AOR. During testing, this may not be desirable. You can start PIE manually in an AOR with the following procedure.
1. From the TOR, enter CRTE SYSID=aor.
2. Enter PIE
Multiple Stand-alone Systems

If you have multiple MRO systems, you can share the Repository and Options files between MRO systems with function shipping. The following figure shows an example of a system configuration that shares the Repository and Options file.

In this configuration, systems may not be always synchronized because the first time a public PIE object (that is, a group or system object) is read, PIE/CICS places the object in memory. After that, the object is accessed from memory. If the object is updated in one system, that system will be aware of the change and will replace the object in memory the next time it is used. However, other systems are not aware of any changes. They continue to use the original object in memory. A change to the Repository will be realized in other systems only when the changed object is edited (accessed for update but not changed) in those systems or when those systems are cycled.

You may need separate Options or Repository files for different systems. For example, you may want to allocate separate Options and Repository files in a test region before placing it into production. Or, you may be unable to share files between systems because they are in different locations or they are non-MRO. In that case, your system would look like the example shown in the following figure.: 
Sharing Repository and Options Files

Complete the following procedure to share both the Repository and Options files between systems:

1. **Install PIE/CICS into one system (we’ll call it CICS1).**
   
   This system “owns” the Repository and Options files.

2. **Make the table changes described in “Step 4: Add Entries to CICS Tables,” on page 6 of the PIE/CICS Installation Guide for each stand-alone system.**

3. **Edit the FCT and make the following changes:**
   
   - Replace the entry for the Repository file (PC@REPS) with the entry in member FSFCT of the CNTL dataset.
   - Specify the CICS1 system name for the SYSIDNT parameter. For example, specify SYSIDNT=CICS1.
   - Change the SERVREQ parameter to BROWSE for the PC@OPTN Options file entry.

4. **Perform steps 5, 7, and 9 of the PIE/CICS Installation Guide for each stand-alone system.**

   □ You must perform all CICS table changes for each release, including running the PIERDO job.

   Because you are sharing your Options and Repository files, customization is complete with “Step 9: Testing and Implementation,” on page 17—all product customization has been done.

**WARNING**

Do not use VSAM Option 3 to share the files—having the file read/write in one system and read-only in another. This can cause unpredictable results. Use the procedure described in “Using Different Options or Repository Files,” on page 106.
Using Different Options or Repository Files

Complete the following procedure to prepare different Options or Repository files:

1. **Install PIE/CICS in one region, CICS1, as usual.**

   The remaining steps apply to the second region, CICS2.

2. **Create a new VSAM file for your Repository or Options file, or both. To do so, make a copy of member DEFVSAM of the CNTL dataset and edit the copy.**

   You may want to REPRO information from your existing Options or Repository files to minimize the amount of work to customize PIE/CICS in your new region. To do so, specify the high-level qualifiers for the existing datasets (the datasets to copy from) in the Q1 and Q2 substitution parameters. Then, check the REP SRCE and OPTSRCE DD statements to see that those statements correctly refer to your existing datasets. Then, specify the new dataset names in the IDCAMS statements at the end of the job. Also, specify the volumes and space allocation for the new datasets.

   If you want to create new, default datasets, rather than copy from existing datasets, specify the high-level qualifiers for the new datasets in the Q1 and Q2 substitution parameters. You can leave the dataset names in the REP SRCE and OPTSRCE DD statements as is. Specify the new dataset names in the IDCAMS statements at the end of the job. Also, specify the new volumes and space allocations.

   If you do not want to create either an Options or Repository file, comment out the IDCAMS statements referring to the file you do not want.

   When you have finished your changes, run the job.

3. **Complete the procedure described in “Step 4: Add Entries to CICS Tables,” on page 6 of the PIE/CICS Installation Guide.**

   If you are sharing the Repository file, replace the FCT entry for the Repository file (PC@REPS) with the entry in member FSFCT of the CNTL dataset. For the SYSIDNT parameter, specify the CICS1 system name. For example, specify SYSIDNT=CICS1.

   If you are sharing the Options file, change the Options entry (PC@OPTN). Change the SERVREQ parameter to BROWSE.

   You must perform all CICS table changes, including running the PIERDO job, for each CICS System Definition (CSD). If you are sharing CSDs, you may skip this step.

4. **For CICS2, perform steps 5-9, except step 6, of the PIE/CICS Installation Guide.**

   If you are sharing your Options file, you can skip “Step 8: Enter PIE/CICS License Information,” on page 14.

   If you are sharing your Repository file, customization is complete with “Step 9: Testing and Implementation,” on page 17—all product customization has been done.

   If you are not sharing your Repository file, you must now customize PIE/CICS for each product. If you have copied your existing Repository file, you can add, change, and delete PIE objects to suit your new region. If you did not copy it, you can still upload objects from the old repository to the new using the Repository Load utility. See “Repository Load Utility,” beginning on page 77 of the Operation and Administration Guide for instructions.
Running Multiple Versions of CICS

You can share PIE/CICS datasets between regions running different versions of CICS. The following procedure assumes PIE/CICS has been supporting regions running a single version of CICS. Now, PIE/CICS needs to support a new region running a different version of CICS. The procedure refers to the PIE/CICS Installation Guide for instructions to complete some steps.

1. **Create a new PIE/CICS load library for the regions running the original version of CICS.**

   Use a different low level dataset qualifier. Instead of using LOADLIB, use LOADnnn, where nnn is the version number of CICS running in this region. The new load library should be empty.

2. **Copy the following modules into the new PIE/CICS load library.**

   - PCSTPPVI: Signon verification program
   - PCTTPUX1: Resource access verification exit
   - PC@XPSEC: External security interface
   - PG@RPCSC: PIE/CICS table assembly program

3. **Concatenate the new load library ahead of the original PIE load library in the region's start-up JCL.**

4. **Create a separate PIE/CICS load library for the region running a different version of CICS.**

   Again, the low level qualifier of the dataset name should be LOADnnn.

   The same modules listed in step 2 must be added to the new load library. However, instead of simply copying the members, you must reassemble them into the load library. The members must be reassembled to support the new version of CICS.

5. **Edit members PCSTPPVI, PCTTPUX1, PC@XPSEC, and PG@RPCSC, and PIERDO of the CNTL and JCLLIB datasets.**

   Change the following parameters listed in each member.

   - **CICSQ**: High-level qualifier of the new CICS load library (not the PIE/CICS load library).
   - **LOAD**: Name of the second new PIE/CICS load library

6. **Run the assembly jobs.**

   The load modules should be placed in the new PIE/CICS load library. (You will add this library to your start-up JCL later.)

7. **Continue with the procedure described in "Multiple Stand-alone Systems," on page 104.**

   When you update your start-up JCL, add the new loadlib created in step 4 to the job. Place the new loadlib before the regular PIE loadlib.
Chapter 9  VTAM Applications

You can log on to PIE/CICS from one VTAM application or CICS region and execute programs running in another session. You can execute programs belonging to different VTAM applications, such as TSO and IMS, in different sessions. These applications can run in unconnected CICS regions, such as stand-alone regions or multiple MRO TORs.

Using MultiCICS sessions or Dynamic Menus to run programs in other VTAM applications fosters the image of a single, unified system. Users do not have to locate an application before running a program. They do not have to manually log off and log on from one region to another to run different applications. They simply select sessions or menu lines to execute the programs they need to complete a task.

You can make the transfer process completely transparent to the user, which also enhances the perception of a single unified system. Users can automatically log on to a target region without ever seeing a log on screen. The application program executes as soon as the user logs on to the target region. From the user’s point of view, all applications are executing from the same CICS region.

PIE/CICS manages the transfer of control to the target application and back to the originating application. It automates the sign on procedure by passing user IDs, passwords, and profiles to the application. It also does error checking in the event of a target application being unavailable or a log on attempt is rejected. Instead of leaving the terminal at the VTAM logo, which could be confusing to users who did not know he was passing through VTAM in the first place, PIE/CICS returns user to their previous session—that is, the session he was in when he opened the VTAM session—or the menu from which the request was made.
Application Transfer Methods and Options

PIE/CICS uses three commands to transfer control to other VTAM applications: PASS, TSWITCH, and ACCESS. PASS and TSWITCH are available with all PIE/CICS products. ACCESS is available only with NetGate. Each command has a characteristic method to transfer control to the destination session.

There are several major steps in the application transfer process. You can control how each step is handled. The steps are:

- Leave the originating region
- Log on to the target region
- Start an application in the target region
- Return to the originating region

You can log on the user to the target region automatically. In CICS, the Logon Director can handle sign on automatically using the VTAM log on message or a script. This requires some customization. In most other VTAM applications you must use the ACCESS command and create a script that logs on users.

Finally, you can start an application in the target region automatically. If you are going to a PIE/CICS region, you can specify the initial transaction in the Execute field. If you are going to another type of VTAM application, you must use the ACCESS command and use the log on script to execute the desired program.

Differences Between PIE/CICS Transfer Session Commands

PASS and TSWITCH commands use CLSDST PASS to access a target region. TSWITCH always disconnects the user from the originating region. PASS disconnects the user by default. However, PASS can log off users from their originating region by specifying A in the Allow Disc Status field of the user’s entry in the User Directory.
The figure below shows the major difference between the three commands. With TSWITCH, users are automatically reconnected to the originating region when they exit from the target VTAM application. With PASS, users are sent to the VTAM log on screen. If PASS disconnected you, you can sign back on to PIE/CICS and resume your sessions. If you were logged off, you start from the beginning when you sign on.

TSWITCH can be regarded as a method to temporarily transfer to another VTAM application and then return to the original application. PASS, on the other hand, is intended to make a permanent switch to another VTAM application.

ACCESS, which is available only with NetGate, works entirely differently. It uses virtual terminals, just like a VTAM session manager. It connects to the target region and remains connected to the originating region.

The benefit to users is that they can switch between their target region and sessions running in the originating region. If you use PASS or TSWITCH, users cannot switch between regions. With PASS, they must either end out of the target region or PASS back to it. With TSWITCH, users must end out of the target region.

In addition, if you want to write a script to execute in the target region—for example, to log the user on and to start the desired application, you must use ACCESS. Scripts execute in the originating region. Because both PASS and TSWITCH disconnect from the originating region, you cannot write scripts that run other VTAM applications in other sessions. With ACCESS, the originating region retains control over the terminal and prevents the script from being interrupted.

When choosing between ACCESS, TSWITCH, and PASS, you may want to consider resource usage of each command. ACCESS creates a permanent link between the regions using virtual terminals that consumes VTAM resources. However, ACCESS does not use system resources to log the user on and off the originating region.
PASS and TSWITCH are exactly the opposite. They do not use virtual terminals that consume VTAM resources. However, they disconnect users from (and TSWITCH logs them back on) the originating region.

Usually, if the user needs to use the target region throughout the day, ACCESS is the preferred command. You may want to use PASS or TSWITCH instead if users simply need to execute a quick utility in a target region and then return to their primary work in the originating region.

If you would prefer to dynamically determine the routing method, you can do so with Dynamic Transaction Routing. See “Dynamic Transaction Routing,” on page 119, for details.
Customizing Transfer Commands

All three transfer commands require some CICS and VTAM customization before they can be used. For ACCESS customization procedures, refer to the NetGate Administration Guide. For PASS and TSWITCH, complete the following procedure.

1. **Edit VTAMLST.**
   
   Locate the APPL statement for the CICS regions from which you will be issuing the PASS or TSWITCH command. The AUTH parameter must specify PASS.

2. **Edit the SIT of the CICS regions from which you will be issuing the PASS or TSWITCH command.**
   
   Ensure the SIT includes the following statement: CLSDSTP=NOTIFY.
   
   The NOTIFY function enables a message to be returned to the originating CICS region when CLSDST PASS has completed. PIE/CICS needs this notification to finish housekeeping routines in the originating region.

3. **Edit the SIT of the target CICS regions.**
   
   Ensure the SIT specifies LGNMSG=YES. This allows the target region to accept log on data, such as user ID and password, from another region.

4. **For TSWITCH, edit the Terminal Control Tables (TCTs) of the target CICS regions.**
   
   Check the RELREQ option in your terminal entries. If you use CEDA, the RELREQ option is in the TERMTYPE resource type.
   
   - Specify RELREQ=NO for all terminals accessible by PIE/CICS.
   - If you use macro tables, the RELREQ option is in the TCT TYPE=TERMINAL macro. RELREQ’s first parameter should be NO. For example: RELREQ=(NO,YES).

Automating Log on Procedures

PIE/CICS is capable of automatic log on to target regions. The log on process depends on whether you are accessing a CICS or non-CICS application, and whether the ACCESS command transfers control to the destination session.

- **CICS regions**
  
  Run the PIE/CICS Logon Director as the good morning transaction in the target region. The Logon Director extracts the VTAM log on message and signs on users.

  - If the PIE/CICS region does not accept a log on request, ensure the statement, LGNMSG=YES, is entered in the SIT.

- **All other applications using PASS or TSWITCH**
  
  If the VTAM application receives VTAM log on message data, PIE/CICS automatically signs on the user. TSO accepts the user ID, but not the password. Most other VTAM applications do not accept the VTAM log on message. Users must manually sign on to the target application.

- **All other applications using the ACCESS command**
  
  You can create a script to sign on users.
For instructions to prepare the PIE/CICS Logon Director as the good morning message and pass the VTAM log on message, see “Logon Director,” on page 3. For instructions to create scripts, see Operation and Administration Guide, “Scripts,” on page 93, or the PIE/REXX User Guide. Both manuals include examples that demonstrate how to log on to a VTAM application.

**VTAM Logmodes**

When a user logs on to a VTAM application, the characteristics of their terminal are taken from an entry in the VTAM logmode table. Usually, VTAM takes the default logmode entry name from VTAMLST. However, a user or a VTAM menu or sessions product can specify a different logmode name. If that occurs, VTAM uses the default logmode name, not the new name, when users TSWITCH or PASS to the target region. Users do not receive any errors going to the target region, but their terminal characteristics are changed. When users exit from the target region and attempt to reconnect to the originating region, they receive errors because they cannot reconnect using a different logmode.

For example, suppose your default logmode is Mod 2. Your VTAM product changes it to Mod 4. When you TSWITCH to another region, VTAM uses the default Mod 2 logmode. You do not receive any errors, but your terminal screens look different. When you exit the target region and attempt to reconnect to the originating region, you will still be using Mod 2. An error occurs because you can only reconnect using Mod 4, the logmode you used to sign on originally.

If this occurs, perform the following procedure. It should be done by someone familiar with your CICS auto-install program.

*NetGate handles logmodes differently. These procedures are not necessary if you will use only the ACCESS command. See “Basic NetGate Customization,” on page 9 of the NetGate Administration Guide for NetGate procedures.*

1. **Refer to the example on page 115, modify the CICS auto-install program to retrieve the logmode name from the CINIT RU. The auto-install program has addressability to the CINIT RU and can find the LOGMODE name in it.**

2. **Pass the logmode name to the PIE/CICS PUXINIT exit. The easiest way is to write the logmode name to a CICS temporary storage record. Use the CICS termid as part of the queue ID to ensure the queue ID is unique.**

3. **In the PUXINIT exit, read the temporary storage record created in the auto-install program. Put the logmode name into field PUXILOGM in the exit commarea.**

4. **Delete the temporary storage queue when the auto-install program is invoked for the de-install of the terminal.**

This procedure sets the logmode for the duration of the user’s PIE/CICS log on, unless the user modifies it with the PIE LOGMODE command.

You can display your logmode with the PIE variable &ZLOGMOD. For more information on this variable, see the PIE/CICS Command Reference.
Example Routine to Extract the VTAM Logmode

This example assembler routine extracts the logmode from CINIT RU.

```
FINDLOG DS OH                   FIND LOGMODE IN CINIT RU ROUTINE
  XC XLOGMODE,XLOGMODE           CLEAR LOGMODE NAME
  *
  L  R6,1PARM4                  R6 = ADDRESS OF CINIT RU
  LA R6,2(R6)                   BUMP PASSED LENGTH
  SR R7,R7                      INITIALIZE
  ICM R7,'0011',10(R6)          LENGTH OF BIND
  *
  LA R6,13(R6,R7)               POINT TO SLU NAME LENGTH
  SR R7,R7                      SLU NAME LENGTH
  *
  LA R6,1(R6,R7)                POINT TO REQ ID LENGTH
  IC R7,0(R6)                   REQ ID LENGTH
  *
  LA R6,1(R6,R7)                POINT TO PASSWORD LENGTH
  IC R7,0(R6)                   PASSWORD LENGTH
  *
  LA R6,1(R6,R7)                POINT TO USER DATA LENGTH
  IC R7,0(R6)                   USER DATA LENGTH
  *
  LA R6,1(R6,R7)                POINT TO CHARACTERISTICS FIELD
  ICM R7,'0011',0(R6)           CHARACTERISTICS FIELD LENGTH
  *
  LA R6,2(R6,R7)                POINT TO SES CRYPTOGRAPHY LEN
  IC R7,0(R6)                   SESSION CRYPTOGRAPHY KEY LEN
  *
  LA R6,1(R6,R7)                POINT TO CONTROL VECTOR
  CLC LMODLIST,0(R6)            IS IT A CONTROL VECTOR?
  BNE EXIT                     ............NO EXIT
  *
  LA R6,1(.R6)                  POINT TO MODE LIST LENGTH
  CLI 0(R6),X'08                IF LENGTH = 8?
  BL EXIT                      ............NO, NOT LOGMODE
  *
  MVC XLOGMODE,1(R6)            SAVE LOGMODE NAME
  EXIT DS OH                    RETURN TO CALLER
  *
LMODLIST DC X'0D'               CONTROL VECTOR-MODE LIST ID
XLOGMODE DS CL8                 LOGMODE
```
Controlling Target Application Programs

After logging on to another session, an initial program is started by the target application. The procedures to start the initial program vary by the application running in the session and what PIE/CICS command was used to transfer to the session.

- **Non-CICS applications accessed by PASS or TSWITCH commands**
  The target VTAM application must accept the initial command from the VTAM log on message to start a program automatically after log on is complete. Most non-CICS applications cannot interpret an initial command from the log on message. In this case, users must start the program using the standard facilities of the VTAM application.

- **Non-CICS applications using the ACCESS command**
  Use a single script to sign on the user to the application and execute the desired program.

- **CICS regions**
  The initial program is passed to the VTAM log on message as the initial command parameter. Complete the following procedure to enable the target region to execute the initial program from the VTAM log on message.

  1. **Change the PIE profiles in the target region.**
     The Initial PIE Command field must begin with PROC. The PROC command processes the initial command passed to the region.

     After the word PROC, enter the default initial command. This becomes the default command that executes if no command is passed, or when the user logs on directly to the target region. You can code any PIE command after PROC. Begin PIE Environment commands with EC.

     **PROC** must run in a new session. If all your MultiCICS sessions are defined, this command cannot run. You can run PROC in a session, for example session 1, so that it becomes the active session to execute the command that was passed.

     **PROC** If you are reconnecting rather than performing an initial log on to the target PIE/CICS region, you are returned to the work in progress when the disconnect occurred. PROC and the initial command are not executed.

  2. **Use the log on Message exit to specify the type of data passed in the VTAM log on message.**
     By default, PIE/CICS passes the user’s ID, their password, and initial command. Several sample Logon Message exits are provided if you want to send other data with the log on message. You need only to reassemble these exits. See "PCTSPDCX—Logon Message Exit," on page 140 for more information.
Coding the Execution String

The session or menu’s execution string is similar for all three commands, ACCESS, PASS, and TSWITCH. However it differs depending on whether you are using the VTAM log on message or a script to sign the user on and execute a program in the target VTAM application.

Using the VTAM Log on Message

CICS regions running PIE/CICS accept the full VTAM log on message. TSO accepts the user ID, but not the password or initial command. Most other VTAM applications do not accept any part of the VTAM log on message. You can test your applications to determine what information they accept from the log on message.

The following syntax diagrams show the format of the ACCESS, PASS, and TSWITCH commands

<table>
<thead>
<tr>
<th>PASS</th>
<th>TSWITCH</th>
<th>vtamappl</th>
<th>userid</th>
<th>password</th>
<th>data</th>
</tr>
</thead>
</table>

Or:

| ACCESS vtamappl | DATA=userid | password | data |

vtamappl VTAM APPLID of the target CICS system. At times, you may want to specify a USERDATA variable to supply different applications for different users. USERDATA can be pre-assigned in the User Directory, entered from a sign on panel, or entered with the User Data Update Utility. See USERDATA in the Command Reference for more information.

userid User ID to log on to the target application automatically. You can use the PIE/CICS variable &ZUSER.

password Password to log on to the target application automatically. The password must follow the slash (/) with no intervening spaces. You can use the PIE/CICS variable &ZPSWD.

If you want to pass an encrypted password, the format is slightly different. Follow userid with .%, instead of /, and use the variable &ZEPSWD:

vtamappl &ZUSER.%&ZEPSWD

data Application you want to invoke in the target system. For PIE/CICS regions, you can specify either a PIE Application command or a CICS transaction. If you want to execute a PIE Environment command, begin it with EC. You can stack commands.

AE is not necessary with ACCESS, TSWITCH, or PASS. Because they are PIE/CICS commands, they all autoend automatically.
Examples

In this example, the user transfers to the CICSPROD region using ACCESS, logs on automatically, and executes CEMT I TAS.

ACCESS CICSPROD DATA=&ZUSER/&ZPSWD CEMT I TAS

Because we used ACCESS, users can switch between all MultiCICS sessions. When they log off, they are returned to the previous session.

In this example, we access the same region with the same parameters, except we use TSWITCH.

TSWITCH CICSPROD &ZUSER/&ZPSWD CEMT I TAS

With TSWITCH, users cannot switch to MultiCICS sessions that run in the originating region. To get to those sessions, they must log off the target region. When they log off, they will be returned to their previous session. (See “Differences Between PIE/CICS Transfer Session Commands,” on page 110, for details.)

Using Scripts to Log on to Target Applications

When you transfer to a session running a non-CICS application with the ACCESS command, you can use a script to complete the log on process and start the initial program.

When you create a script, you normally provide only the most basic elements of the command: the word ACCESS and the target applid. For example:

ACCESS CICSPROD

This command string takes the user to the applid’s log on screen. From that point, you can supply the remaining data needed to sign on to the application by running a play-back or a PIE/REXX script.

From the PIE Sessions or dynamic menu line, you will execute the REXX program or play-back script, not ACCESS. See the PIE/REXX Manual, or “Scripts,” on page 93 of the Operation and Administration Guide for details. Examples:

REXX TSOEDIT
SCRIPT PLAY IMSLOG

Dynamic Menus

For Dynamic Menus, the execution string requirements are the same. However, you may omit the ACCESS, PASS, and TSWITCH commands, if they are first in the execution string, and substitute the type codes N, Z, or W for them. For example:

<table>
<thead>
<tr>
<th>Execute</th>
<th>CICSPROD &amp;ZUSER/&amp;ZPSWD CEMT I TAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>W</td>
</tr>
</tbody>
</table>

Otherwise, code the command and use type code F. For example:

<table>
<thead>
<tr>
<th>Execute</th>
<th>TSWITCH CICSPROD &amp;ZUSER/&amp;ZPSWD CEMT I TAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>F</td>
</tr>
</tbody>
</table>
Dynamic Transaction Routing

Dynamic Transaction Routing provides two main functions:

- You can determine which region a transaction is executed from at run-time. Dynamic Transaction Routing overrides the transaction’s PCT entry. You can also use Dynamic Transaction Routing to execute a transaction that is not defined in the PCT. This allows you to route the same transaction to more than one AOR in a CICSplex.
- You can determine the run-time method to access the target region.

Dynamic Transaction Routing can be used to route both local and remote transactions. For local transactions, PIE/CICS executes the transaction under the P#RM umbrella transaction. For remote transactions, PIE/CICS does not use P#RM; transactions are executed with their own tranid.

When an umbrella transaction is used, CICS does not call external security for the target transaction in the TOR. Instead, CICS checks authorization for the P#RM transaction in the TOR and the target transaction in the AOR.

Dynamic Transaction Routing is controlled by the following PIE/CICS commands, exits, and facilities:

- SYSID command, which invokes Dynamic Transaction Routing
- TR, TP, TS, and TN Environment commands, which determine the access method to the region executing the transaction.
- Dynamic Transaction Routing exit, which specifies the target region and access method
- Network Monitor (NetGate), which can determine the physical name associated with a sysid and the routing method used to access it

Preparing Dynamic Transaction Routing

   
   See “Customization Options,” on page 65 for more information.

   The TP, TS, and TN commands perform PASS, TSWITCH, and ACCESS, respectively. To use these commands, you must set up the target regions as you would if you were using the PASS, TSWITCH, or ACCESS commands themselves. For information on required set up, see “Customizing Transfer Commands,” on page 113, and “Controlling Target Application Programs,” on page 116.

2. You may want to customize the Dynamic Transaction Routing exit.
   
   For more information on this exit, see the “PCMRPCRU—Dynamic Transaction Routing Exit,” on page 133.
Coding the Execution String: Determining the Target Region

Dynamic Transaction Routing is initiated with the SYSID command. When used with the default TR access method, SYSID is similar to the CICS CRTE transaction. There are four major steps to a CRTE transfer:

- Transfer control to the remote system
- Sign on
- Execute the transaction
- Cancel out of the region

With SYSID, the first three steps are accomplished simultaneously. Sign on is processed automatically. When you use AutoEnd with SYSID, users are canceled out automatically from the region.

The SYSID command has the following format:

```
SYSID target[; [PASSP] [ENCP] transaction [arguments]]
```

- **target**: Name of the system to run the transaction. Specify the SYSID if you are transferring to a region within the same CICSplex. If you are transferring to a separate VTAM application, specify the APPLID.
  
  You can hard code the target name, or you can supply the &ARG variable and have users supply the name when they open the session. Also, you can specify the target name with another PIE variable, such as USERDATA, that is assigned elsewhere.

  Use delimiters to separate commands after the target parameter.

- **PASSP**: User's profile sent with the data to sign on to the target region.

- **ENCP**: User's encrypted password sent with the data to sign on to the target region.

- **transaction [arguments]**: Transaction and arguments to be executed in the target region.

Examples of the SYSID Command

The following command string executes CEMT in AOR2 instead of the local region. Users return to the originating region after the transaction is finished.

```
SYSID AOR2; AE CMD CEMT
```

The same command string without AutoEnd leaves the user in the target region.

```
SYSID AOR2; CMD CEMT
```

After the CEMT transaction is finished, users are left with a CICS blank screen in the AOR2 region. If they enter another transaction, it executes from AOR2.

Suppose you want to execute the same transaction in two AORs of a CICSplex. To accomplish this, you can create two MultiCICS sessions. Each session accesses a different region.

```
SYSID AOR1; AE CMD CEMT
SYSID AOR2; AE CMD CEMT
```

Users can determine which region to use by selecting the proper session.
Including the Access Method in the Command String

There are four access methods available with Dynamic Transaction Routing.

- **TR**: Standard CICS transaction routing
- **TP**: Transaction pass, the same as the PASS command
- **TS**: Transaction switch, the same as the TSWITCH command
- **TN**: Transaction NetGate, the same as the ACCESS command

Use TR to access another region in the same CICSplex. Use TP, TS, or TN to access an unconnected region. The differences between these three access methods are described in “Differences Between PIE/CICS Transfer Session Commands,” on page 110.

You select the access method to the target by completing one of the following:

- Hard code the method in the command string. Place it before the SYSID command. For example, the following example executes CEMT in region AOR2 using the TSWITCH access method:
  
  ```
  TS SYSID AOR2; AE CMD CEMT
  ```

- The TS command sets the access method to TSWITCH.

- If you have NetGate, define the default access method for the CICS region in the PNAM file. See “Prepare a PNAM File,” on page 12 of the NetGate Administration Guide for details.


By default, PIE/CICS uses the TR access method. You may change the default using the Dynamic Transaction Routing exit.

You cannot use the PASS, TSWITCH, or ACCESS commands with SYSID. You must use the equivalent state commands; TR, TP, TS, or TN.

Dynamic Menus

For Dynamic Menus, command string requirements are the same. Use type code F. For example:

```
Execute  ==>  TS SYSID AOR2; AE CMD CEMT
Type     ==>  F
```
Network Monitor

Network Monitor tracks the performance of VTAM applications under its control. Network Monitor is used by NetGate, Availability Plus, and NonStop CICS. You need to customize the Network Monitor with procedures described in the NetGate, Availability Plus, and NonStop CICS manuals.

This section discusses special customization procedures to adapt the Network Monitor to unique site conditions. You should review this information to see if it is pertinent to your site.

Procedures to run the Network Monitor are presented in the chapter titled “Network Monitor,” beginning on page 103 of the Operation and Administration Guide.

Increasing the Size of the Network Monitor Systems Table

Some sites need only a few regions to process their applications. Other sites require more complex systems. If your site has more than 200 total regions defined in the systems Directory (NonStop CICS alone or Availability Plus and NetGate regions combined), you must increase the size of Network Monitor’s systems table.

Complete the following procedure if your site includes more than 200 regions in the Systems Directory.

1. **Edit the PNDAPPLS member of the SAMPLIB dataset.**
   
   Each slot in the Systems Directory is created by one @APPL (RES) macro statement.

2. **Add a @APPL (RES) statement for each target region.**
   
   You may have more @APPL statements than regions. The @APPL macro statement conforms to standard assembler coding conventions
   
   ```
   TITLE 'NETWORK APPLICATIONS DIRECTORY'
   @APPL   (RES)
   @APPL   (RES)
   @APPL   (RES)
   .
   .
   @APPL   (RES)
   @APPL   (RES)
   @APPL   (RES)
   END
   ```

3. **Edit member PNDAPPLS of the CNTL dataset to assemble the new systems table.**

4. **Update the substitution parameters (for help, see “JCL Parameters,” on page 197).**

5. **Run the PNDAPLS job.**
Changing Network Monitor Conditions with the Options File

The Network Monitor refers to the Options file to determine how certain tasks are completed. The default settings of the table are appropriate for most installations. However, you may review and change values set in the Options file.

1. Access member PNDMTOPT of the SAMPLIB dataset.

2. Modify the options.
   See “PNDMLOPT Macro,” on page 124 for more information about available options.
   For NonStop CICS, you must delete the APPLPOL option and add APPLID=PIENM001. Other changes are optional or installation dependent.

3. Access member PNDMTOPT of the CNTL dataset.

4. Update the substitution parameters (for help, see “JCL Parameters,” on page 197).

5. Run the PNDMTOPT job.
PNDMMOPT Macro

Network Monitor start-up options are specified with the PNDMMOPT macro. Only one PNDMMOPT macro statement can be coded in an assembly. PNDMMOPT conforms to standard macro coding conventions:

- Line continuations are indicated by a non-blank character in column 72
- Continued lines must end with a comma and a space
- Following lines must start in column 16

The following syntax diagram shows Network Monitor processing options specified with the PNDMMOPT macro. Default and valid values of each parameter are listed in the table beneath the PNDMMOPT syntax diagram:

```
PNDMMOPT TYPE=OPTIONS,
    {APPLPOL=pool-name
    APPLID=applid
    } VTAM={YES|NO},
    WRKINTL=nnnn,
    STAINT1=nnnn,
    STAINT2=nnnn,
    WEQNENT=n,
    GRPNENT=n,
    AVSTAT=description,
    UVSTAT=description,
    ISTAT=description,
    CLSTAT=description,
    MAXSTAT=description
    STAMSG=(code, text, code, text, ...code, text)
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Valid Value</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE</td>
<td>OPTIONS</td>
<td>OPTIONS</td>
</tr>
<tr>
<td>APPLPOL</td>
<td>1 to 7-characters</td>
<td>PIENM</td>
</tr>
<tr>
<td>APPLID</td>
<td>1 to 8-characters</td>
<td>None</td>
</tr>
<tr>
<td>VTAM</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>WRKINTL</td>
<td>0001 to 9959</td>
<td>0005</td>
</tr>
<tr>
<td>STAINT1</td>
<td>001 to 9959</td>
<td>0100</td>
</tr>
<tr>
<td>STAINT2</td>
<td>0001 to 9959</td>
<td>0500</td>
</tr>
<tr>
<td>WEQNENT</td>
<td>1 to 99</td>
<td>25</td>
</tr>
<tr>
<td>GRPNENT</td>
<td>1 to 999</td>
<td>20</td>
</tr>
<tr>
<td>AVSTAT</td>
<td>1 to 16-characters</td>
<td>AVAILABLE</td>
</tr>
<tr>
<td>UVSTAT</td>
<td>1 to 16-characters</td>
<td>UNAVAILABLE</td>
</tr>
<tr>
<td>ISTAT</td>
<td>1 to 16-characters</td>
<td>UNDEFINED</td>
</tr>
<tr>
<td>CLSTAT</td>
<td>1 to 16-characters</td>
<td>CLOSED</td>
</tr>
<tr>
<td>MAXSTAT</td>
<td>1 to 16 characters</td>
<td>MAX-USERS</td>
</tr>
<tr>
<td>STAMSG</td>
<td>code: 6 hex digits text: 1 to 16-characters</td>
<td>See text</td>
</tr>
</tbody>
</table>
Network Monitor

**TYPE**
Type of macro generation. It should always be OPTIONS.

For NetGate and NonStop, you must specify either APPLPOL or APPLID. (They are mutually exclusive.) Neither is needed for Availability Plus.

**APPLPOL**
Name of the ACB pool used to define the Network Monitor. The default is PIENM. When the Network Monitor starts in a region, PIE/CICS automatically searches for the first available ACB in the pool. Use APPLPOL if you are running it in more than one region.

For information about preparing an ACB pool, read the chapter titled “Basic NetGate Customization” of the NetGate Administration Guide.

**APPLID**
Single ACBNAME used to define the Network Monitor. The default is PIENM001. Use APPLID if you are running the Network Monitor in one region only.

**VTAM**
Choice (Yes or No) to monitor the status of an APPLID using VTAM INQUIRE and to control user sessions with VTAM SENDCMD.

For NonStop CICS and NetGate, specify the YES default.

For Availability Plus without NetGate, you can set this option to NO because Availability Plus does not reference APPLIDs. If you leave VTAM=YES, you must also leave the APPLPOL option coded as shipped.

The Network Monitor uses VTAM INQUIRE to verify the status of target regions at pre-set monitoring intervals. The WRKINTL, STAINT1, and STAINT2, parameters set the frequency or period between INQUIRE monitoring intervals. Each parameter sets the monitoring interval for a different group of target regions.

The period between monitoring intervals is expressed in minutes/seconds (mmss). For example, 0001 is 1 second, and 9959 is 99 minutes and 59 seconds.

VTAM INQUIRE monitoring uses CPU and real memory resources. You should set the period between intervals long enough to ensure adequate performance, but short enough to accurately monitor each region. A specific, recommended interval is included with the discussion of each parameter.

**WRKINTL**
Network Monitor verifies target regions that return users from the NSCR. This condition usually indicates the region may have crashed. The period between monitoring intervals should be fairly short; five seconds is the recommended interval.

WRKINTL does not affect Availability Plus. If you run Availability Plus but not NetGate, set WRKINTL to the same period as STAINT2 to minimize resource usage.

**STAINT1**
Network Monitor tracks the status of target regions that were requested with the PIE/CICS &ZAPPL(applid) variable. The period between monitoring intervals should be moderately short; 7 to 10 seconds.

STAINT1 does not affect Availability Plus. If you run Availability Plus but not NetGate, set STAINT1 to the same period as STAINT2 to minimize resource usage.
Network Monitor tracks target regions to verify if they are available. Set the period between STAINT2 monitoring periods short enough to ensure that regions are reinstated promptly after they become available again. If your site has plenty of regions available for routing, specify a longer period; possibly 15-30 seconds.

Number of entries in the Work Element Queue.

Number of entries in the Network Monitor Group Table. Make sure there is at least one entry for each balancing group in your NonStop CICS system or your Availability Plus and NetGate systems.

The following STAT fields indicate the current status of VTAM application under various conditions. An application’s status is displayed in the Status field of the Applid List. You can also obtain the status using the PIE variable &ZSYS(sysid).

Character string that indicates the region is running normally and is available for work. The default string is AVAILABLE.

Character string that indicates a region is unavailable because it is not active or Availability Plus has marked it unavailable because an attempt to route a transaction to it has failed. The default string is UNAVAILABLE.

Character string that describes the initial APPLID status. The initial status is displayed when the Network Monitor has not yet checked the application for availability. The default string is AVAILABLE.

Character string that indicates the region has been closed using the PNAM STOP command. The default string is CLOSED.

Character string that indicates an application is closed because the maximum number of users are currently logged on. The default string is MAX-USERS.

Links VTAM return codes to corresponding status messages. Network Monitor determines an application’s status by using the VTAM INQUIRE macro. When an INQUIRE request is completed, VTAM returns the:

- VTAM return code (RTNCD - two hex characters)
- Feedback-two code (FDBK2 - two hex characters)
- Feedback-three code (FDBK3 - two hex characters)

STAMSG code refers to these codes in concatenated form—6 hex characters in the order listed. STAMSG text is the status message corresponding to the code. PIE/CICS will display this message when you use the PIE variable &ZAPPL(applid). For more information on this variable, see the PIE/CICS Command Reference.

The default option lists all of the possible VTAM status values for an application and corresponding status messages. You can change the status message for a code, but do not change or delete any codes.
### Example of PNDMMOPT Macro Options

| PNDMMOPT | TYPE=OPTIONS,                       NG APPLID |
|----------|-------------------------------------|----------|
|          | APPLID=PIENM001,                   | NM APPLID |
|          | VTAM=YES,                          | DO VTAM INQUIRE |
|          | WRKINTL=0005.,                     | PERFORM WORK SCAN DELAY |
|          | STAIN1=0007.,                      | STATUS UPDATE INTERVAL 1 |
|          | STAIN2=0010.,                      | STATUS UPDATE INTERVAL 2 |
|          | WEQNENT=40.,                       | # ENTRIES IN W/E QUEUE |
|          | GRPNENT=20.,                       | # ENTRIES IN GROUP TABLE |
|          | AVSTAT=AVAILABLE,                  | DEFAULT 'AVAILABLE' |
|          | UVSTAT=UNAVAILABLE,                | DEFAULT 'UNAVAILABLE' |
|          | ISTAT=AVAILABLE,                   | INITIAL/DEFAULT STATUS |
|          | CLSTAT=CLOSED,                     | 'CLOSED' STATUS |
|          | MAXSTAT=MAX-USERS,                 | 'AT MAX USERS' STATUS |
|          | UASTAT=UNAVAILABLE,                | UNAUTHORIZED STATUS |
|          | STAMSG=(000000,AVAILABLE,         | APPLID ACT. & |
|          |                                  | ACCEPT.LOGONS |
|          | 000004,UNAVAILABLE,                | APPLID INACTIVE (CONCT) |
|          | 000008,UNAVAILABLE,                | APPLID ACT., NOT |
|          |                                  | ACCEPT.LOGONS |
|          | 000010,UNAVAILABLE,                | APPLID ACT., NOT |
|          |                                  | ACCEPT.LOGONS |
|          | 0000A0,UNAVAILABLE,                | APPLID NOT CONNECTABLE |
|          | 145300,UNDEFINED,                 | APPLID UNKNOWN TO VTAM |
|          | 144000,UNDEFINED.                 | APPLID UNKNOWN TO VTAM |

END
Chapter 10  PIE/CICS Exits

Exits can be used to customize PIE/CICS to suit your requirements. More than half of the PIE/CICS exits are stand-alone exits. These exits are assembled into their own load module. The source you assemble is the actual exit.

The remaining seven exits are assembled together into a single load module, PCTTPUX1 (PUX1 for short). The PUX1 source, itself, is not an exit. It simply names the seven exits to copy into the PUX1 load module.

All exits, whether stand-alone or PUX1, are called at different times and achieve different objectives.
Stand-alone Exits

The assembly JCL for all stand-alone exits is located in the CNTL dataset. Sample exit source members are located in the SAMPLIB dataset. The load module is located in the LOADLIB.

For PCSTPPVI and PC@PSEC, the member names of the assembly JCL and load module are the same as the exit name (PCSTPPVI or PC@PSEC). However, because these exits interact with external security, there is a separate source member for each supported security system. The source member names are shown in the sections that describe these exits. For all other exits, the member name of the assembly JCL, load module, and source are the same as the exit name.

When you upgrade to a new release of PIE/CICS, you need to know which exits have been modified to maintain the modifications in the new release. If you modify an exit, log the modification on the worksheet on page 177.

<table>
<thead>
<tr>
<th>Exit Name</th>
<th>Purpose</th>
<th>When Invoked</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC@UPVUD USERDATA</td>
<td>Inspects and modifies data entered by the user</td>
<td>After the user has entered data in the USERDATA command, but before the value is accepted</td>
<td>“PC@UPVUD—USERDATA Validation Exit,” on page 132</td>
</tr>
<tr>
<td>PC@XPSEC*</td>
<td>Works with the PUXMNUL exit to authorize the display of screen data to an external security system</td>
<td>If a menu line definition requests it, after the PUXMNUL exit and before the menu line is displayed</td>
<td>“PC@XPSEC—External Security Exit,” on page 132</td>
</tr>
<tr>
<td>PCMRPCRU Dynamic</td>
<td>Overrides the destination region of a transaction or the access method used to route the transaction</td>
<td>After PIE/CICS has made the transaction routing decision, but before the transaction is executed</td>
<td>“PCMRPCRU—Dynamic Transaction Routing Exit,” on page 133</td>
</tr>
<tr>
<td>Session Shutdown Exit</td>
<td>Performs system cleanup after an application ends</td>
<td>When a session ends—either normally or abnormally, by a user CLOSE or CANCEL or by AutoEnd</td>
<td>“PCSCPCAN—Session Shutdown Exit,” on page 134</td>
</tr>
<tr>
<td>PCSMPFIL</td>
<td>Suppresses PIE/CICS Environment commands or modifies their parameters</td>
<td>After a PIE Environment command is issued, before it is executed</td>
<td>“PCSMPFIL—PIE Environment Command Exit,” on page 135</td>
</tr>
<tr>
<td>PCSSPAEE Auto End Exit</td>
<td>Suppresses AutoEnd or changes it to Confirm End.</td>
<td>After the application is ended, before the session or menu line is AutoEnded</td>
<td>“PCSSPAEE—AutoEnd Exit,” on page 136</td>
</tr>
<tr>
<td>PCSSPTOE Time-out Exit</td>
<td>Changes the action taken by PIE/CICS in response to a terminal that has exceeded the idle time threshold</td>
<td>Just before PIE locks a terminal</td>
<td>“PCSSPTOE—Time-out Exit,” on page 137</td>
</tr>
<tr>
<td>Exit Name</td>
<td>Purpose</td>
<td>When Invoked</td>
<td>Reference</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PCSSSEXT TSSQ Substitution Exit</td>
<td>Overrides normal TSQ name substitution to permit data to be shared between multiple sessions</td>
<td>Before PIE/CICS changes the name of the temporary storage queue</td>
<td>“PCSSSEXT—TSQ Substitution Exit,” on page 138</td>
</tr>
<tr>
<td>PCSTPPVI* Sign on Verification Exit</td>
<td>Verifies user ID and password with an external security system</td>
<td>During PIE/CICS sign on</td>
<td>“PCSTPPVI—Sign on Verification Exit,” on page 139</td>
</tr>
<tr>
<td>PCTSPDCX Logon Message Exit</td>
<td>Interprets the VTAM log on message</td>
<td>Before the Logon Director reads the VTAM log on message</td>
<td>“PCTSPDCX—Logon Message Exit,” on page 140</td>
</tr>
<tr>
<td>PNC@PLPE PASS/TSWITCH Exit</td>
<td>Writes a message to the PIE message log.</td>
<td>Before CLSDST PASS is performed</td>
<td>“PNC@PLPE—PASS/TSWITCH Exit,” on page 142</td>
</tr>
<tr>
<td>PNDMPEXT Network Monitor Inquiry Exit</td>
<td>Overrides the choice of a virtual terminal or the availability of a VTAM application</td>
<td>When PIE/CICS needs to assign a virtual terminal or determine the status of a VTAM application</td>
<td>“PNDMPEXT—Network Monitor Inquiry Exit,” on page 142</td>
</tr>
</tbody>
</table>

* This exit interfaces with external security. There are different source members for each supported security system.
PC@UPVUD— USERDATA Validation Exit

The &ZUDATA variable is an 8-byte user data field that can be set by each user with the User Data Utility. Typically, the variable contains short data segments like billing codes or employee numbers. PC@UPVUD allows you to edit and validate &ZUDATA variable data.

PC@UPVUD is given control when the user invokes the USERDATA command. PO@UPVUD is given control after the user has entered a value, but before the value has been accepted. PC@UPVUD examines the data and sets a return code.

PC@UPVUD can take the following actions based upon the validity of data within the &ZUDATA variable.

- **Sign off immediately**
  
  If user data is not valid, you can specify the User Data Update Utility to sign off immediately or allow the user to end the user data update process.

- **Return error messages**
  
  You may return an error message up to 79 characters long or a short message up to 20 characters long. The short message is displayed at upper right-hand corner of the User Data Update screen.

- **End update with next input**
  
  If user data is valid, you may instruct the User Data Update Utility to end normally when the user presses ENTER.

Complete the following procedure to modify the User Data Validation exit:

1. **Edit the PC@UPVUD sample member of the SAMPLIB dataset.**

2. **Modify the exit.**

   It is invoked by a program LINK, may be written in either Assembler or COBOL, and can contain CICS command level code. The assembler definition of the communications area is located in member PC@UMVUD of the MACLIB.

3. **Assemble the exit using member PC@UPVUD of the CNTL dataset.**

   PC@UPVUD is an in-stream procedure that must be customized. Comments included in the JCL describe required and optional parameters. For a detailed description of these parameters, see “JCL Parameters,” on page 197.

   The exit is new-copyable.

PC@XPSEC— External Security Exit

The PC@XPSEC exit works with the PUXMNUL exit to determine which menu lines will be displayed for a particular user.

It is fully documented in “External Security,” on page 68 of the Dynamic Menus Administration Guide.
Stand-alone Exits

**PCMRPCRU— Dynamic Transaction Routing Exit**

The PCMRPCRU exits controls transaction routing. PCMRPCRU assumes control after PIE/CICS has selected the access method and region to run a transaction. PCMPPCRU assumes control before the transaction is executed.

You can use this exit to override either aspect of the decision: the region routed to or the access method used. You can substitute:

- AOR to route the transaction to
- Remote tranid to run in the AOR
- Remote transaction profile, if ISC routing
- Local transaction or PIE command for the transaction

For instance, you can select a different AOR based on user or terminal ID. Or, you can route some users to a production region and others to a test region. Refer to “Dynamic Transaction Routing,” beginning on page 119 for more information on dynamic transaction routing.

Complete the following procedure to change the Dynamic Transaction Routing exit.

1. **Edit member PCMRPCRU of the SAMPLIB dataset.**

2. **Modify the exit.**
   - It is invoked by branch entry, must be written in Assembler, and cannot contain any CICS command level code.
   - PCMRPCRU should not issue any CICS macro calls or operating system SVCs.
   - PCMRPCRU is invoked on every transaction route and could substantially increase the CICS TOR overhead if it does too much processing.
   - The assembler definition of the commarea is located in member PCMRFMUSR of the MACLIB dataset. This member also provides help on the exit’s possibilities.

3. **Assemble the exit using member PCMRPCRU of the CNTL dataset.**
   - Change the JCL to suit your environment, then run the job. The PCMRPCRU exit is link-edited with program PCMRPCRP, which is not new-copyable.

4. **To put your new exit into effect, you must cycle CICS.**
PCSCPCAN—Session Shutdown Exit

PCSCPCAN performs system clean-up operations when an application running in a PIE/CICS session is unable or unwilling to perform itself. For example, some applications do GETMAINs or create temporary storage queues without releasing these resources when they are no longer needed.

The PCSCPCAN exit is given control when a session ends either normally or abnormally. A session can end in the following ways:

- AutoEnd or a system end key.
- User-issued CLOSE, CANCEL, or LOGOFF commands
- If the user’s entry in the User Directory specifies Alloc Disc NO, and the user’s terminal is disconnected from CICS or the user enters the EXIT DISC command. All open sessions are cancelled.

There are several ways to prevent a session from being cancelled. (See “Closing and Cancelling Sessions,” on page 55 of the MultiCICS Administration Guide for details. PCSCPCAN exit is called even when a session is non-cancellable.

When a session is cancelled, the application running in the session cannot perform normal system cleanup. A PIE session cancel causes the same abnormal end of a transaction as a CICS lost session. You may want to put logic from your existing DFHZNEP into this exit. You can also use PCSCPCAN to override the decision to end or not end the session.

Complete the following procedure to change the Session Shutdown exit.

1. **Edit member PCSCPCAN of the SAMPLIB dataset.**
2. **Modify the exit.**
   - It is invoked by a program LINK, may be written in either Assembler or COBOL, and can contain CICS command level code. It may be invoked in a non-terminal attached mode from DFHZNEP.
   - The assembler definition of the commarea is located in member PCSCMCNE of the MACLIB dataset.
3. **Assemble the exit using member PCSCPCAN of the CNTL dataset.**
   - Change the JCL to suit your environment.
4. **Submit the PCSCPCAN job.**
PCSMPFIL— PIE Environment Command Exit

The PCSMPFIL exit assumes control before processing a PIE Environment command. PCSMPFIL can modify the command string before executing the command.

PCSMPFIL is the counterpart of the PUXCMMD exit that controls PIE Application commands (see “PUXCMMD— PIE Application Command Exit,” on page 148). Sometimes it is difficult to tell which of the two exits will be invoked. The rules are:

- All PIE Environment commands invoke the PCSMPFIL exit.
- All PIE Application commands invoke the PUXCMMD exit.
- A command such as ==CEMT invokes both exits. That is because this command has two possible meanings.

It can be a command to switch to the session named CEMT—a short form of SWITCH CEMT. As such, it is an Environment command and will invoke the PCSMPFIL exit.

It can also be a command to start a new session and execute CEMT in it—a short form of ==TRAN CEMT.

PIE/CICS will try the ==SWITCH CEMT version first; so the PCSMPFIL exit will always be called. If there is no session named CEMT, it will execute the ==TRAN CEMT version, and so invoke the PUXCMMD exit.

With PCSMPFIL, you can modify anything in the command string.

To change your PIE Environment Command exit, perform the following.

1. **Edit member PCSMPFIL of the SAMPLIB dataset.**
2. **Modify the exit.**
   
   PCSMPFIL is invoked by a branch entry and must be written in Assembler. The assembler definition of the commarea is located in member PCSMMFIL of the MACLIB dataset and cannot contain any CICS command level code.

3. **Assemble the exit using member PCSMPFIL of the CNTL dataset.**
4. **Change the JCL to suit your environment,**
5. **Submit the PCSMPFIL job.**
PCSSPAEE—AutoEnd Exit

The PCSSPAEE exit is executed during Auto End processing. PCSSPAEE has access to the contents of the final Terminal Input Output Area (TIOA) of the transaction (the last screen send when the transaction ends).

PCSSPAEE can override PIE/CICS’s decision to AutoEnd an application and return to the previous session or terminate the session. You can use PCSSPAEE to override AutoEnd and change the termination procedure to a Confirm End.

PCSSPAEE has the following uses:

- Disable AutoEnd for certain applications—for example, applications that pseudo-converse using the next tranid on the screen.
- Change AutoEnd to Confirm End for applications that send termination messages.
- Change AutoEnd to Confirm End when the application ends with an error message.
  
  AutoEnd automatically changes to Confirm End if the transaction abends. However, an application can end unexpectedly with an error message, but not abend. PCSSPAEE can examine the final output message and evaluate messages from different transactions differently.

There are two sample Auto End exits in the PIE SAMPLIB dataset:

- PCSSPAEE changes AutoEnd to Confirm End if the transaction ends and writes CICS message DFH2014 to the terminal. This message indicates an AOR processing a remote transaction has become unavailable.
  
  If you are using Dynamic Transaction Routing, the change to Confirm End occurs automatically.

- PCSSPAE2 changes AutoEnd to Confirm End if the ending transaction is tranid TEST and the message being written is not all blanks or nulls. The TEST transaction normally pseudo-converses or ends and clears the screen. Under certain abnormal conditions, TEST ends and writes an error message to the screen. This exit allows the message to be read by users.

Complete the following procedure to modify the PCSSPAEE exit:

1. Edit member PCSSPAEE of the SAMPLIB dataset.
2. Make any necessary modifications to PCSSPAEE.

   The assembler definition of the commarea is located in member PCSSMAEE of the MACLIB dataset.

3. Assemble the exit using member PCSSPAEE of the CNTL dataset.

   Change the JCL to suit your environment, then run the job.

   PCSSPAEE is link-edited with program PCSSPDET, which is not new-copyable.

4. To put your new exit into effect, cycle CICS.

   The assembler definition of the commarea is located in member PCSMMFIL of the MACLIB dataset.
Stand-alone Exits

PCSSPTOE—Time-out Exit

The PCSSPTOE exit is called when PIE is about to lock a terminal because it has been idle longer than the time-out period specified in the User Directory. Based upon a flag set in the PCSSPTOE exit, PIE/CICS can take one of three actions after an idle terminal has exceeded the time-out period:

- Ignore the time-out and leave the terminal unlocked
- Ignore the time-out but prevent switching from the current session (lock the user into the current session)
- Lock the terminal and display the Lock screen

Data from the following fields of the commarea is passed to the PCSSPTOE exit.

- TOETRAN Tranid running in the current session
- TOEPHYNA PIE logical name of the current session
- TOELOGNA PIE physical name of the current session
- TOEELAPS Elapsed time in seconds the terminal has been idle
- TOEFLAGS Flag that indicates the action to take after a terminal has been idle longer than the time-out period.

Complete the following procedure to modify the PCSSPTOE exit:

1. **Edit the sample exit in member PCSSPTOE of the PIE SAMPLIB dataset.**
2. **Modify the exit.**
   - It is invoked by a program LINK, may be written in either Assembler or COBOL, and can contain CICS command level code.
3. **See member PCSSMTOE of the PIE MACLIB for a map of commarea fields passed to the exit.**
4. **Set the TOERETC field of the commarea to one of the following values:**
   - TOENOLCK Do not lock the terminal
   - TOELCKSW Do not lock the terminal, but prevent session switching
   - TOELOCK Lock the terminal
5. **Reassemble the exit using member PSCCPTOE of the CNTL dataset.**
Chapter 10  PIE/CICS Exits

PCSSSEXT—TSQ Substitution Exit
PIE/CICS automatically modifies the name of each temporary storage queue (TSQ) created by an application. This ensures that TSQs have unique names for similar applications running in different sessions. This prevents data from an application overwriting another application’s data stored in a shared TSQ.

The PCSSSEXT exit permits you to override normal TSQ name substitution and allow multiple sessions to access the same TSQ. This is necessary for applications that share data for multiple users using TSQs.

The TSQFLDS and TERMASK tables also affect TSQ name substitution. See “TERMASK: Construct Unique Pseudo Termids,” on page 185, for more information.

Complete the following procedure to modify the PCSSSEXT TSQ Substitution exit.

1. Edit member PCSSSEXT of the SAMPLIB dataset.

2. Modify the exit.
   It is invoked by branch entry, must be written in Assembler, and cannot contain any CICS command level code.
   The assembler definition of the commarea is located in member PCSSMEXT of the MACLIB dataset.
   - Set the EXTRETC flag field to EXTBYP to bypass TSQ substitution.
     MVI EXTRETC,EXTBYP
   - Set EXTSUBID to the new TSQ ID and set the EXTRETC flag to EXTCHG to change the TSQ ID.
     MVC EXTSUBID,=CL8'NEWTSQID'
     MVI EXTRETC,EXTCHG

3. Assemble the exit using member PCSSSEXT of the CNTL dataset.

4. Change the JCL to suit your environment.

5. Submit the PCSSSEXT job.
   The exit becomes effective after CICS has been recycled.
**PCSTPPVI— Sign on Verification Exit**

The PCSTPPVI exit is invoked when a user logs on to PIE/CICS. PCSTPPVI assumes control after the Logon Director has verified the user ID and password with the Terminal and User Directories. PCSTPPVI completes the verification process by passing the user’s ID and password to an external security program. You can also use the PCSTPPVI exit to:

- Change the default action taken for terminals that are not listed in the Terminal Directory. Normally, unlisted terminals are sent to the CICS good morning transaction.
- Change the action taken for a USE or REQ terminal. Normally, the terminal is sent to the CICS good morning transaction.
- Change the action taken when the user presses PF3 or PF15 to end the Logon Director screen. Normally, PCSTPPVI issues CICS DISCONNECT, which does a VTAM log on from CICS.

Complete the following procedure to modify the PCSTPPVI Sign on Verification exit. The SAMPLIB dataset contains a separate PCSTPPVI sample member for each security system.

1. **Edit the appropriate security member of the SAMPLIB dataset:**
   - PCSTNONE Do not interact with external security
   - PCSTRACF RACF
   - PCSTTOPS TOPSECRET
   - PCSTACF2 ACF2
   - PCSTOMNI OmniGuard

2. **Modify the exit.**
   - It is invoked by a program LINK, may be written in either Assembler or COBOL, and can contain CICS command level code.
   - The assembler definition of the commarea is located in member PCSTMPVI of the MACLIB dataset.

3. **Assemble the exit using member PCSTPPVI of the CNTL dataset.**

4. **Change the JCL to suit your site’s environment.**

5. **Submit the PCSTPPVI job.**
Chapter 10  PIE/CICS Exits

PCTSPDCX—Logon Message Exit

The Logon Director reads and processes the contents of the VTAM log on message. This information is used to automatically log on users and bypass the Logon Director screen. Normally, the log on message is passed by a VTAM menu system or another PIE/CICS region.

PCTSPDCX determines how information within a VTAM log on message is parsed and interpreted. By default, PCTSPDCX accepts user ID, password, and initial PIE command from the message in a variety of formats. The exit must be changed if you want to pass the Logon Director a user profile or data in an unsupported format with a VTAM log on message.

You can also use PCTSPDCX to examine and change the incoming VTAM log on message before it is read. The user ID, password, or profile can be modified based on whatever criteria you like—terminal ID, for instance. In fact, this exit can even be used to create a log on message when none exists.

This feature requires the statement LGNMSG=YES be specified in the SIT table.

Default Formats

The default exit accepts messages in many formats: free form, field form, SSI, and encrypted form. Each type of format has several supported variations.

You may see the exit source for each variation. Here, we show only the most common variation of the most common format—the free form message:

```
<userid>[/[<password>] [<initial command>]]
```

You can use a vertical bar (|), instead of the slash (/), to indicate “Force Reconnect.” The encrypted password format is similar: simply use a percent sign (%) instead of the slash. To pass an encrypted password and force reconnect, use a question mark (?).

Examples

This VTAM log on message passes JJOHNS, APPLE, and CEMT I TAS as the user ID, password, and initial PIE command.

```
JJOHNS/APPLE CEMT I TAS
```

This message passes just the user ID and password:

```
JJOHNS/APPLE
```

This message passes only the user ID and initial command:

```
JJOHNS/ CEMT I TAS
```

Using a Sample Log on Exit

Four sample variations of the Logon Message exit are included as members of the SAMPLIB dataset. The sample exit members are:

- PCTSPDC1 allows you to examine and change the user ID and password passed.
- PCTSPDC2 allows you to pass user ID password, profile, and initial command.
  (Wanting to send the profile is the most common reason for changing the exit.)
- PCTSPDC3 writes data from the log on message to the TCTTE user area.
- PCTSPDC4 supports an alternate format: passing the user ID and password separated by a blank.
Complete the following procedure to use the sample log on exits:

1. **Edit member PCTSPDCX of the CNTL dataset.**

2. **Change the MEMBER parameter to the sample member you want to use. For example, specify MEMBER=PCTSPDC2.**

3. **Change other parameters to customize the procedure for your site.**
   - The comments in the JCL describe the required and optional parameters. For a detailed description of these parameters, see "JCL Parameters," on page 197.

4. **Run the job.**

### Changing the Exit Code

To change your Logon Message exit, do the following.

1. **Edit the PCTSPDCx member of the SAMPLIB dataset.**

2. **Modify the exit.**
   - It is invoked by a program LINK, may be written in either Assembler or COBOL, and can contain CICS command level code.
   - The assembler definition of the commarea is located in member PCTSMDCX of the MACLIB dataset.
   - You can edit and change the log on message. The message can be up to 255 characters and can contain the user ID, password, profile, and initial command.
   - The VTAM log on message is deleted after it is read. It can be read once only.

3. **Assemble the exit using member PCTSPDCX of the CNTL dataset.**
   - PCTSPDCX is an in-stream procedure that must be customized for your site. The comments in the JCL describe the required and optional parameters. For a detailed description of these parameters, see "JCL Parameters," on page 197.
**PNC@PLPE— PASS/TSWITCH Exit**

PNC@PLPE is invoked just before CLSDST PASS when executing a PASS or TSWITCH command. The user ID, node name, date, time, target APPLID, and whether PASS or TSWITCH is being executed, are passed to the exit.

PNC@PLPE can be used to write a message to the PIE log. For example:

```
NODE: LDEV001, USER: AD001 PASSED TO TSO
```

You can pass the message back to the exit after it is written to the PIE log.

To change the exit, do the following.

1. **Edit member PNC@PLPE of the SAMPLIB dataset.**
2. **Modify the exit.**
   
   It is invoked by a program LINK, may be written in either Assembler or COBOL, and can contain CICS command level code.
   
   The assembler definition of the commarea is located in member PNC@PLPE of the MACLIB dataset.
3. **Assemble the exit using member PNC@PLPE of the CNTL dataset.**

**PNDMPEXT— Network Monitor Inquiry Exit**

PNDMPEXT is executed whenever a virtual terminal needs to be assigned, or the status of a VTAM applid needs to be determined. In practice, this means the exit is invoked when a NetGate session is opened, a PASS or TSWITCH command is executed, or a pseudo-conversation is begun under Availability Plus.

You can use the exit to override the choice of virtual terminal or the reported status of an applid (AVAILABLE or UNAVAILABLE). For example, you can restrict access to a VTAM application by user ID or time of day, or you could select a virtual terminal pool based on user ID.

To change the exit, do the following.

1. **Edit member PNDMPEXT of the SAMPLIB dataset.**
2. **Modify the exit.**
   
   It is invoked by a program LINK, may be written in either Assembler or COBOL, and can contain CICS command level code.
   
   The assembler definition of the commarea is located in member PNDMPEXT of the MACLIB dataset.
3. **Assemble the exit using member PNDMPEXT of the CNTL dataset.**
## PUX1 Exits

Seven PUX1 exits are assembled into the PCTTPUX1 load module. This section describes

- Common features of the PUX1 exits, such as the User Exit table, which controls which PUX1 exits are invoked, and the token list used to analyze PIE Application command strings
- PUX1 source, which is used to assemble the PUX1 load module
- Each PUX1 exit

The names of the seven PUX1 exits begin with “PUX.” The exits are.

<table>
<thead>
<tr>
<th>Exit Name</th>
<th>Description</th>
<th>When Invoked</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUXCMMD</td>
<td>Prevents users from issuing specific Application commands or changing command parameters</td>
<td>After a PIE Application command is about to be executed. Its variables have already been substituted.</td>
<td>“PUXCMMD—PIE Application Command Exit,” on page 148</td>
</tr>
<tr>
<td>PUXINIT</td>
<td>Use it to override the values in the User Directory entry for a user. Example: to change the user’s profile. You may also use it to perform special security processing. Example: get the user’s group ID from your security database.</td>
<td>During sign on—after the user ID and password have been validated with CICS and PIE/CICS and before the profile is assigned. (PCSTPPVI exit gets control first. Then PIE/CICS does further processing. Then PUXINIT gets control.)</td>
<td>“PUXINIT—Post Sign on Initialization,” on page 149</td>
</tr>
<tr>
<td>PUXMNUL*</td>
<td>PUXMNUL works in tandem with the stand-alone exit PC@XPSEC. These exits tie menu line display to external security authorization. Use PUXMNUL to override display decisions, based on external or internal security. Example: make every menu line display for a system administrator, whether or not the administrator is authorized to execute the line.</td>
<td>If the menu line definition requests the exit, before a dynamic menu line is displayed</td>
<td>“PUXMNUL—Menu Line Security Exit,” on page 150</td>
</tr>
<tr>
<td>PUXOBJ</td>
<td>Use it to override normal PIE object access rules. Example: to permit Group Administrators to edit the User Directory.</td>
<td>After a PIE object has been selected for browse or update</td>
<td>“PUXOBJ—PIE Object Authorization Exit,” on page 150</td>
</tr>
</tbody>
</table>
### Exit Name | Description | When Invoked | Reference
--- | --- | --- | ---
PUXVARS | Use it to suppress variable substitution or to override the value chosen. (The PUXCMMMD exit is called after this.) | When a PIE Application command is issued, before its variables are replaced with their current values | “PUXVARS—Variable Substitution Exit,” on page 151
PUXVIEW | Use it to determine on the fly which users can use the VIEW command. | After the user issues the VIEW command | “PUXVIEW—View Command Access Exit,” on page 151
PUXVSCN | Use it to allow or disallow this View request based on the circumstances of the request. Example: prohibit a particular application from being viewed by most users. | After the user issues the VIEW command and after the PUXVIEW exit | “PUXVSCN—View Screen Access Exit,” on page 152

*This exit interfaces with external security. There are different source members for different external security systems. See the exit detail for more information.

When you upgrade PIE/CICS, you will need to know which exits have been modified, so that you can transfer the modifications to the new release. If you modify an exit, please note that on the worksheet on page 167.
Working With PUX1 Exits

Some information is common to all PUX1 exits. This section presents this common information.

Assembling the PUX1 Load Module

If you are using an exit with a different name—for example, you are using member P02INIT instead of member PUXINIT—you must change the name in the PUX1 source member. Complete the following procedure to reassemble the PUX1 load module.

1. **Edit the different PUX1 source members for each security system.**
   
   There is a different PUX1 source member for each security system:
   
<table>
<thead>
<tr>
<th>Security System</th>
<th>Source Member</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCTTNONE</td>
<td>Do not interact with external security</td>
</tr>
<tr>
<td>PCTTRACF</td>
<td>RACF</td>
</tr>
<tr>
<td>PCTTTOPS</td>
<td>TOPSECRET</td>
</tr>
<tr>
<td>PCTTACF2</td>
<td>ACF2</td>
</tr>
<tr>
<td>PCTTOMNI</td>
<td>OmniGuard</td>
</tr>
</tbody>
</table>
   
   Locate the COPY statement for the exit name you want to change and supply the new name.

2. **You may add code to the PUX1 source.**
   
   It must be written in CICS command level (Assembler or COBOL). The assembler definition of the commarea is found in member PCS@MUXP of the MACLIB dataset.

3. **Assemble the exit using member PCTTPUX1 of the CNTL dataset.**
   
   PCTTPUX1 is an in-stream procedure that must be customized for your site. The comments in the JCL describe the required and optional parameters. For a description of these parameters, see Appendix H "JCL Parameters," on page 197.

User Exit Table

P UX1 exits are activated by the PC@STEXT User Exit table. You can exclude unnecessary exits for better performance.

You can also specify whether an exit is invoked by CICS link or branch entry. If an exit does not need CICS services, specifying branch entry reduces the path length to call the exit. Currently, there is no sample program using a branch entry; however you may edit existing exits to do so.

The default table invokes only the PUXINIT and PUXMNUL exits. To change the exits invoked, perform the following.

1. **Edit the User Exit table: member PC@STEXT of the SAMPLIB.**

2. **Change the PUXEXIT macros.**
   
   There is one macro for each PUX1 exit. The PUX1 exits are identified by the EXIT parameter, using logical names. The logical names refer to the following exits:
   
<table>
<thead>
<tr>
<th>Logical Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MENULINE</td>
<td>PUXMNUL, the Menu Line Access exit</td>
</tr>
<tr>
<td>OBJ ACC</td>
<td>PUXOBJ S, the PIE Object Access exit</td>
</tr>
</tbody>
</table>
Chapter 10  PIE/CICS Exits

PEXECMD  UXCMMD, the PIE Application Command exit.
POSTINIT  PUXINIT, the Post Sign on Initialization exit.
VARSUB  PUXVARS, the Variable Substitution exit.
VIEWACC  PUXVIEW, the View Command Access exit.
VIEWSCRN  PUXVSCN, the View Screen Access exit.

Find the macro for the exit you want to change. Then specify either YES or NO on the INVOKE parameter.

3. **Reassemble the User Exit table using the PC@STEXT member of the CNTL dataset.**

PC@STEXT requires customization. The parameters are described in the J CL. For a detailed description of these parameters, see "J CL Parameters," on page 197.

**PUXEXIT Macro Format**

```
PUXEXIT TYPE=ENTRY,
EXIT=exit_name,
PROGRAM=PCTTPUX1,
INVOKE=YES|NO,
METHOD=CICSLINK|BRENTRY
```

- **EXIT**  Logical name of the exit. Valid names are listed in step 2 of this procedure.
- **PROGRAM**  Exit load module name. Currently, the only valid name is PCTTPUX1.
- **INVOKE**  Choice (Yes or No) to invoke the exit. Specify YES to invoke the exit or NO to not invoke the exit.
- **METHOD**  Specify how the exit will be invoked. Specify CICSLINK to invoke it using a CICS LINK. This allows you to write the exit in either Assembler or COBOL and to use CICS command level code. Specify BRENTRY use a branch entry. Branch entry requires you to write in Assembler and prohibits CICS command level code.

**PUX1 Commarea**

The assembler definition of the PUX1 commarea is found in member PCS@MUXP of the MACLIB dataset. All PUX1 exits use this commarea. The first portion of the commarea contains parameters that are passed to all PUX1 exits. The second portion of the parameters are exit-specific—that is, only the parameters for a particular exit are sent.
The following parameters are passed to all exits.

<table>
<thead>
<tr>
<th>Address of TCTTE</th>
<th>PUXPTCTA</th>
<th>fullword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of userid</td>
<td>PUXPUSRL</td>
<td>1 char</td>
</tr>
<tr>
<td>Userid field</td>
<td>PUXPUSER</td>
<td>20 chars</td>
</tr>
<tr>
<td>Length of terminal nodename</td>
<td>PUXPNODL</td>
<td>1 char</td>
</tr>
<tr>
<td>Terminal Nodename</td>
<td>PUXPNODE</td>
<td>8 chars</td>
</tr>
<tr>
<td>Termid</td>
<td>PUXPTERM</td>
<td>4 chars</td>
</tr>
<tr>
<td>System administrator Indicator (Y/N)</td>
<td>PUXPSCTY</td>
<td>1 char</td>
</tr>
<tr>
<td>Group administrator Indicator (Y/N)</td>
<td>PUXPGRAD</td>
<td>1 char</td>
</tr>
<tr>
<td>Groupid for user</td>
<td>PUXPGPID</td>
<td>8 chars</td>
</tr>
<tr>
<td>Address of ACEE</td>
<td>PUXPACEE</td>
<td>4 chars</td>
</tr>
</tbody>
</table>

For exit-specific parameters, look for the exit name in member PCS@MUXP of the MACLIB.

**Base Registers**

PUX1 exits use the following general registers:

- Register 12 and 11 are program code base registers.
- Register 13 addresses dynamic storage
- Register 10 addresses the CICS exec interface block.
- Register 9 is reserved. It can be used as the third base register.
- Register 8 addresses PUXPMBLK, the PUX1 commarea.

Registers 0-7, 14, and 15 are available on entry to each PUX1 exit.

**Working with a Token List**

The PUXVARS, PUXCMMD, and PUXMNUL exits analyze PIE Application command strings. A token list is provided to parse the command string into individual elements. A token list is a sequence of fixed length entries. Each entry describes an element of the command string. Each token list entry can be mapped by the P@@MTOK DSECT.

The length of an entry is given by the symbol @@@MTOKL.

To address the word, take the offset value in the field PTOKOFFS and add it to the address of the command string.

The size of the word is in the PTOKLNGH field. Both PTOKLNGH and PTOKOFFS are halfwords.

You can create a program loop that analyzes the token list. In that case, the first token is addressed by the token list address (PUX@TOKA). You shift to the next token in the string by increasing the token address by the value of @@@MTOKL.

To directly address the nth token, multiply the value of @@@MTOKL by n-1, to get the offset. Then add that value to the address of the token list (PUX@TOKA).

The token count is passed in PUX@TOKN.
Chapter 10  PIE/CICS Exits

PUX1 Source
The PUX1 source is not an exit. It simply names the seven exits to copy into the PUX1 load module. However, there are several PUX1 source members, each is used with a different external security system.

You need only change the PUX1 source to change the name of an exit. For example, if you change a copy of the PUXINIT exit and want to assemble with the copy, you must edit the PUX1 source and supply the new PUXINIT member name.

For information on changing the PUX1 source, see “Assembling the PUX1 Load Module,” on page 145.

PUXCMMMD— PIE Application Command Exit
PUXCMMMD receives control after a PIE Application command has been parsed and variables have been substituted, but before the command is processed. PUXCMMD can suppress an Application command or change parameter values.

For example, you can use PUXCMMD to restrict the use of the SEND command to a few specified users. Or, your can prevent users from issuing the EXIT END PIE command, which logs users off PIE/CICS and drops them back to a CICS blank screen. You could add logic to PUXCMMD to change EXIT END PIE to something else, such as EXIT SIGNOFF or EXIT LOGOFF.

The PUXCMMD exit has several limitations. You cannot change the command verb. For example, you cannot change EXIT to SEND. Variable substitution has already occurred before PUXCMMD assumes control. If you place a variable in the string, it is not substituted.

The PUXVARS exit gives you the same control before command variables are substituted.

To change your Application Command exit, do the following.

1.  **Edit member PUXCMMD of the MACLIB dataset.**
   Specify whether to accept or reject the command in the PUXPRTCD field. Specify PUXPRTNR to accept the command or PUXCMREJ to reject it.
   If you want to change the parameter string, you can place it in the commarea work buffer or input string area or in program storage. Specify the new parameter string length in the PUXCNSTL field. Specify the parameter string address in the PUXCNSTA field.
   The exit uses a different number for each Application command. The Command verb, its number, and type code are shown in member PCMXMCMC of the MACLIB.

2.  **Reassemble the PCTTPUX1 load module as described in “Assembling the PUX1 Load Module,” on page 145.**

3.  **Tell PIE/CICS to invoke PUXCMMD, using the User Exit table.** See “User Exit Table,” on page 145, for more information.
PUXINIT—Post Sign on Initialization

PUXINIT is invoked during the sign on process after the User Directory entry is identified and validated, but before the profile is retrieved. PUXINIT has two primary uses; to override the values in the user’s User Directory entry and perform security-related processing.

For example, PUXINIT could be used to change the user’s profile or Lock Time-out value, based on time of day. Or you could use it to retrieve the user’s Group name from your external security database.

PUXINIT receives all of the fields from the User Directory entry and the Dial and Opt fields of the Terminal Directory. It also receives certain log on information—such as how the user logged on—through VTAM, through the Logon Director screen or without the Logon Director screen.

PUXINIT Samples

There are several sample exits in the MACLIB dataset.

- P01INIT gets user data from the log on screen. It takes any data in the log on screen Profile field and overrides the User Directory User Data field with it. (You can change the name of the Profile field to anything you like, such as Billing Code, so that this function is clear to users.) The user data is translated to uppercase. If there is no data in the Profile field, the User Data value will not be changed.

- P02INIT gets user data from RACF or TOPSECRET. It extracts the data pointed to by the users’ ACEE (Accessory Environment Element) and reformats it. It uses this value to override the User Data field. The RACF or TOPSECRET data is expected in the format described by DSECT UIDATA. If there is no data, the User Data value will not be changed.

- P03INIT extracts the ACF2 lock time-out value and replaces the User Directory Lock Timeout value with it.

- P04INIT gets the user’s Group ID from the group field in the RACF database.

Changing the PUXINIT Exit

To change your Post Sign on Initialization exit, do the following.

1. **Edit member PUXINIT of the MACLIB dataset.**
2. **Reassemble the PCTTPUX1 load module as described in “Assembling the PUX1 Load Module,” on page 145.**

   As shipped, the User Exit table invokes PUXINIT. If you do not want PUXINIT invoked, change the User Exit table. See “User Exit Table,” on page 145, for more information.
Chapter 10  PIE/CICS Exits

**PUXMNUL— Menu Line Security Exit**
PUXMNUL works in conjunction with the PC@XPSEC exit to determine whether menu lines can be viewed by a particular user. PUXMNUL is fully documented in “External Security,” on page 68 of the Dynamic Menus Administration Guide.

As shipped, the User Exit table invokes PUXMNUL. If you do not want PUXMNUL invoked, change the User Exit table. See “User Exit Table,” on page 145, for more information.

**PUXOBS— PIE Object Authorization Exit**
PUXOBS controls access to PIE/CICS objects. There is an extensive set of rules to govern who may retrieve and update PIE/CICS objects. (See “Object Security,” on page 75.) PUXOBS lets you override these rules.

For example, you could use it to allow a group administrator to update the User Directory. Or you could use it to disallow system administrator authority over a particular object.

Complete the following procedure to change your Object Authorization exit.

1. **Edit member PUXOBS of the MACLIB dataset.**
   The PUXPRTCD field tells you whether the access rules were passed. To reject access to the object, change the field to PUXOAREJ. To grant access, change the field to PUXPRTNR.

2. **Reassemble the PCTTPUX1 load module as described in “Assembling the PUX1 Load Module,” on page 145.**

3. **Tell PIE/CICS to invoke PUXOBS, using the User Exit table.** See “User Exit Table,” on page 145, for more information.
PUXVARS— Variable Substitution Exit

The PUXVARS exit receives control after a PIE Application command has been parsed and before variables have been substituted. PUXVARS can override normal variable substitution or suppress variable substitution altogether. You can, for instance, prevent the substitution of potentially sensitive variables, such as &ZUSER and &ZPSWD.

You can also use PUXVARS to suppress an Application command or change its parameters, just as with PUXCMMD. (The difference between them is when they are called. PUXVARS is called before variable substitution, PUXCMMD after.)

The PUSVARS exit has several limitations. You cannot change the command verb. For example, you cannot change EXIT to SEND. Variable substitution has already occurred when the exit takes control. If you place a variable in the string, it cannot be substituted.

To change your Variable Substitution exit, do the following.

1. **Edit member PUXVARS of the MACLIB dataset.**
   Specify how to substitute variables in the PUXPRTCD field. Specify PUX@SNON to prohibit variable substitution, PUX@SALL to substitute all variables, or PUX@SYES to substitute only unsecured variables (that is, all variables except &ZPSWD).
   If you want to change the parameter string, you can place it in the commarea work buffer or input string area or in program storage. Specify the new parameter string length in the PUXCNSTL field. Specify the parameter string address in the PUXCNSTA field.
   The exit uses a different number for each Application command. The Command verb, its number, and type code are shown in member PCMXMCMC of the MACLIB.

2. **Reassemble the PCTTPUX1 load module as described in “Assembling the PUX1 Load Module,” on page 145.**

3. **Tell PIE/CICS to invoke PUXVARS, using the User Exit table. See “User Exit Table,” on page 145, for more information.**

PUXVIEW— View Command Access Exit

PUXVIEW controls user access to the View utility. (Also see “PUXVSCN—View Screen Access Exit,” on page 152.)

To change your View Command Access exit, do the following.

1. **Edit member PUXVIEW of the MACLIB dataset.**
   Specify whether to allow use of View using the PUXPRTCD field. To deny access, change the field to PUXPSVRJ. To grant access, change the field to PUXPRTNR.

2. **Reassemble the PCTTPUX1 load module as described in “Assembling the PUX1 Load Module,” on page 145.**

3. **Tell PIE/CICS to invoke PUXVIEW, using the User Exit table. See “User Exit Table,” on page 145, for more information.**
PUXVSCN—View Screen Access Exit

Like the PUXVIEW exit, PUXVSCN controls the PIE/CICS View facility. While PUXVIEW allows you to control the feature by user ID alone, PUXVSCN grants or reject a View request based on other criteria. For example, you can use PUXVSCN to prevent users from viewing the screen of a particular application such as Payroll, but allow them to view other application screens.

The PUXVSCN exit modifies the action of the VIEW CAPTURE command if the user is not authorized to use the command. If an unauthorized user attempts to capture a screen with VIEW CAPTURE, the user is placed in the VIEW CAPTURE entry screen with the following message displayed on their terminal:

You are not authorized to capture the specified screen.

The VIEW CAPTURE entry screen is passed as the destination to the PUXVSCN exit in the PUXVVFLG field. PUXVVFLG can be set to PUXVVEND instead to cause PIE to perform an END rather than display the VIEW CAPTURE entry screen. Also, the PUXVVFLG field can be set to PUXVURET to cause PIE to do a RETURN after an attempt to use the VIEW CAPTURE command by an unauthorized user.

Complete the following procedure to change your View Command Access exit.

1. **Edit member PUXVSCN of the MACLIB dataset.**
   
The PUXPRTCD field tells you whether the access rules were passed. To reject access to the object, change the field to PUXVSREJ. To grant access, change the field to PUXPRTNR.

2. **Reassemble the PCTTPUX1 load module as described in “Assembling the PUX1 Load Module,” on page 145.**

3. **Tell PIE/CICS to invoke PUXVSCN, using the User Exit table. See “User Exit Table,” on page 145, for more information.”**
Chapter 11  Performance and Tuning

PIE/CICS is designed to be used by the general user population. Maintaining adequate PIE/CICS performance is essential to provide an effective work environment. This chapter provides tuning suggestions to maximize the performance of PIE/CICS and reduce its resource requirements. These tuning suggestions should be implemented before placing PIE/CICS into production.
Virtual Storage

The following section gives recommendations to assign transactions based upon their anticipated usage of DSA storage. Also, a separate section describes how to place queues in a separate temporary storage owning region to reduce the usage of DSA storage.

Conserving DSA

Use the CICS Class Max feature for the following PIE/CICS transactions. If your site runs CICS 4.1 and above, these transactions use DSA storage above the 16 Mb line. If your site runs an earlier version of CICS, these transactions use DSA storage below the 16 Mb line instead.

- PSGM
  This is the PIE log on transaction. Users tend to log on to CICS at approximately the same time each morning, or at system restart because of an outage. As a result, a fairly large number of PSGM transactions may run concurrently during short periods of the day. Each PSGM task can use up to 60K of DSA.

- PCDP
  On return from TSWITCH (which passes to another VTAM application and then returns to the originating region), users are reconnected to the originating session with the PCDP transaction. PCDP also may use up to 60K of DSA. Assign PCDR to the same class as the PSGM transaction.

- PSWC
  This is the SWITCH transaction. PSWC is used heavily because PIE switches to a special session to analyze and execute every command. The PSWC transaction executes every time the user enters an escape string, presses a PIE global key, or enters a command. Each PSWC transaction uses about 40K of DSA.

- P#NT
  This is the NetGate transaction. Since responses from various VTAM applications may be slow, many P#NT tasks may be active at the same time. Each P#NT task uses about 12K of DSA.

For CICS 3.1 and below, specify TSMGSET=4 in the CICS SIT table. A higher value needlessly consumes virtual storage and limits available DSA. (PIE/CICS uses many temporary storage queues and strongly affects the average. It frequently outweighs the affect of other products.)

If you are using AUX temporary storage for PIE/CICS, use a large CISZ for the AUX dataset. Otherwise, excessive TSUTs will be created.

If you are using MAIN temporary storage, you may need to increase the value for IEALIMIT. The default is 32 meg.
CICS Temporary Storage Table
You can place all PIE/CICS queues in a separate temporary storage owning region. A separate temporary storage owning region conserves DSA storage.

Complete the following procedure to create a separate temporary storage owning region.

1. **Edit member PIETST of the PIE SAMPLIB dataset.**
   
   PIETST creates a CICS Temporary Storage Table (TST). This table converts all PIE temporary storage queue requests to remote requests.

2. **Find the TYPE=INITIAL macro.**

3. **Supply a suffix for the DFHTSTxx table for the SUFFIX parameter of the TYPE macro.**

4. **Find the TYPE=REMOTE macro and change the DATAID parameter from @YZV to @YZ.**
   
   If you changed the PIE/CICS TSQ prefix using the Temp Storage Queue Prefixes field on the PIE Customization Options screen, specify the new prefix with the DATAID parameter.

5. **Assemble the TST using standard CICS procedures.**
   
   The values set in the TST become effective the next time you cycle CICS.
Real Storage—ELPA

Most PIE/CICS programs may be placed in the MVS Extended LPA. If you are running multiple PIE/CICS regions, placing programs in the ELPA can significantly reduce the demand for real memory and lower paging rates.

Refer to member PIEALT of the CNTL dataset for a list of eligible programs. Member PIEALT of the SAMPLIB contains these entries for programs in CICS ALT (Application Load Table) format.

Most PIE/CICS programs are ELPA-eligible. Programs in the ELPA run in protected storage and cannot be corrupted. Alternatively, you can use the storage protection feature of CICS 3.3 and above. All ELPA-eligible programs are loaded into read-only DSA and protected from corruption. However, using the ELPA also saves real memory and is the preferred method.
DASD I/O

The follow sections give recommendations to minimize the number of DASD I/Os with PIE/CICS.

VSAM Datasets

The PIE/CICS Options file is accessed only during PIE/CICS initialization and has no tuning requirements. The Repository database, which is a VSAM keyed dataset, stores all PIE objects. Because public objects are kept in temporary storage after they have been initially accessed, the Repository does not experience a significantly large number of I/O requests during normal operation.

However, the Repository experiences more I/Os if a large percentage of users have been granted private object authority; objects that have the user’s ID as their Group ID. Restricting private object authority to a limited number of users tends to reduce the number of I/Os to the Repository.

You may want to perform normal tuning for buffer/string waits for the dataset.

Temporary Storage

The Customization Options utility includes the Temp Storage on Auxiliary field. You should specify YES for this field to use AUX temporary storage rather than MAIN. While using auxiliary storage increases the average number of I/Os, it is preferred for the following reasons:

- In CICS version 2.1, AUX is preferred to avoid MVS GETMAIN requests for MAIN storage. GETMAIN requests issue SVC interrupts to CICS.
- In CICS version 3.1 and above, MAIN temporary storage is acquired with CICS GETMAINs above the 16 Mb line. However, AUX is still preferred because references to above the line DSA cause page faults that place the entire CICS address space in a wait state. An I/O to a file in auxiliary storage does not cause page faults.

CICS manages AUX temporary storage efficiently to minimize the number of I/Os to the VSAM cluster. Highly used data is kept in VSAM buffers. The AUX temporary storage cluster can be tuned to minimize I/O. CICS performance statistics can be used to assess the length of buffer waits. The number of buffers and strings can be adjusted using the CICS SIT table.

If you specify NO, you can improve performance by tuning the MVS paging subsystem.

PIE Session Shutdown Exit

Place the PCSPCAN Session Shutdown exit in the load library, even if it is just a dummy. CICS scans the entire DFHRPL each time a session is shut down if the PCSPCAN exit is not stored in the load library.
Network Optimization

If NetMizer is installed at your site, it should be used for all remote 3270 terminals to reduce the volume of network traffic and improve overall transaction response time. The maximum level of optimization, FUL, is recommended. FUL optimization includes repetitive character elimination, delta filtering, semantic imaging, and inbound suppression.

Non-3270 Optimization

The Opt field of the Terminal Directory determines the NetMizer data compression method to reduce the volume of data transferred over a network. If Netmizer is not installed, the value set on the Opt field still affects CPU usage. If the Opt field is set to N32, PIE/CICS does not maintain a screen image. PIE/CICS maintains a screen image in memory for any other NetMizer value placed on the Opt field.

Maintaining a screen image is required for all of the following PIE/CICS products or features:

- MultiCICS (enable session switching)
- SuperMessage
- PIE Lock
- Bulletin command
- NetMizer
- Print command
- Global PIE keys
- PIE/REXX programs and play-back scripts
- Disconnect/reconnect and session outage recovery.

If NetMizer nor any of the listed products or features are used, specify N32 on the Opt field to reduce CPU usage.

Local Terminals

Turn off NetMizer if you have a significant number of local terminals. Specify NO in the Opt field of the Terminal Directory for all local terminals.

Using NetMizer with Other Screen Managers

PIE/CICS must use a screen manager to maintain screen image for the products and features listed above. NetMizer uses a screen manager to optimize data sent over the network. If you use another optimization product, it will maintain a separate screen image. Converting all applications to the NetMizer screen manager reduces CPU usage and increases available real and virtual storage.
Short Log on Messages

You can use PIE/CICS to create a VTAM menu. As part of this set up, you will probably LOGAPPL a significant number of terminals to the PIE/CICS region and run the Logon Director as the good morning transaction. Terminals go directly to the Logon Director screen when they are turned on, or when the CICS region is started. After entering their sign on data from the Logon Director screen, users go directly to the VTAM menu.

When you start a CICS region, the Logon Director runs in every LOGAPPLed terminal at the same time. This can cause a network overload. You can significantly reduce this sudden load using the Short Logon Message feature. With Short Logon Messages, terminals do not receive the Logon Director screen immediately. Instead, they receive the following message:

Connected to cics-name, press Enter to continue.

When users press ENTER, the full Logon Director appears on their screen. Because users arrive at their terminals at different times, the network load for sign ons is distributed over a longer period of time.

Edit the Terminal Directory to activate Short Logon Messages. Specify YES in the SLM field of each LOGAPPLed terminal.
PIE/CICS Exits

There are seven PUX1 exits. These exits are enabled by the PC@STEXT exit table. Exclude any exits from the PO@STEXT table that are not being used. Only the PUXINIT and PUXMNULL exits are active by default.
Customization Options Utility

Implement the following recommendations with the Customization Options utility to minimize the amount of system resources required to run PIE/CICS:

- If MultiCICS is not installed at your site, specify NO on the Temp Storage Name Substitution field of the Customization Options utility. This prevents the CICS temporary exit from being called and saves significant CPU resources.
- If MultiCICS is installed, refer to “Using the Customization Options Utility,” beginning on page 66 for information about setting values for this field.
- The MRO/ISC Support field affects CPU use. Set it to the lowest value necessary for the support you require. See “Using the Customization Options Utility,” beginning on page 66, for more information.
- For the Dynamic Trans Routing field, specify YES only if you are running Availability Plus, or if you are using Dynamic Transaction Routing. Otherwise, specify NO. You can create a separate Options file for your Availability Plus AORs and run your TORs with the Dynamic Trans Routing field set to NO.
Appendix A Customer Support

This appendix describes how to get help from Customer Service when you experience a problem with a UNICOM Systems Software product. This appendix includes separate sections that describe several diagnostic suggestions to rule out common problems and the information you should have ready before reporting the problem.

Contacting Customer Service

UNICOM Systems Software Customer Service can be reached by the following methods:

- Phone  818-838-0606
- Fax     818-838-0776
- Email   support@unicomsi.com

A Support and Services web page provides Customer Service information about all of UNICOM Systems Software’s products. Use the following URL to browse the Support and Services web page:

http://www.unicomsi.com/support/index.html

The Support and Services web page provides an online form to report a problem with a UNICOM Systems Software product. Use the following URL to complete and submit a Technical Support Request form:

http://www.unicomsi.com/support/index.html

Normal business hours are from 6:00 a.m. to 4:30 p.m. Pacific Standard Time, Monday through Friday. Emergency customer service is available 24 hours a day, 7 days a week.

An answering service receives customer service calls beyond normal business hours. You may leave a message if it is not an urgent problem. A customer service representative will return your call at the start of the next business day.

Requests for urgent support outside of normal business hours are answered immediately. A customer service representative will be summoned to return your call. Leave a phone number where you can be reached. If you have not received a return call from a Customer Service representative within an hour of reporting the problem, please call back. Our customer service representative may be experiencing difficulties returning your call.

International customers should contact their local distributor to report any problems with a UNICOM Systems Software product.
Troubleshooting Suggestions

This section describes several troubleshooting suggestions to diagnose common errors that can cause PIE/CICS problems. Before calling Customer Service, follow these suggestions to rule out the possibility these errors are causing your PIE/CICS problem.

- Run the Installation Verification Program with the \#IV transaction. Browse the PIECIVP temporary storage queue to see if it contains error messages that suggest problems with the allocation of CICS programs, transactions, maps, or files.
- Verify any recent changes to your site’s operating system, CICS, or other products are fully compatible with PIE/CICS.
- Verify all load modules are at the same release level if a new release of PIE/CICS was installed over a previous release.
- Check that all modules were reassembled after upgrading PIE/CICS or applying maintenance to CICS.
- Verify all PIE/CICS system tables were reassembled after applying maintenance to CICS or upgrading to another release.
- Verify all users have current PIE/CICS passwords and entered them correctly.
- Examine your CICS logs, MVS console, and PIE/CICS logs for error messages from not only PIE/CICS, but any other product that runs concurrently with PIE/CICS.
Describing the Problem

Gather the following information about your system environment before reporting a problem to UNICOM Systems Software Customer Service:

Operating system release and PUT Level ________________________________
VTAM system release and PUT Level ________________________________
PIE/CICS release ________________________________________________
Date of PIE/CICS distribution tape ____________________________________

Gather the following information about your CICS system before reporting a problem to UNICOM Systems Software Customer Service:

CICS release and PUT Level _________________________________________
CICS configuration MRO/ISC etc. ______________________________________
Real or virtual terminal ______________________________________________

Get answers to the following questions before calling UNICOM Systems, Inc. Customer Service.

What PIE/CICS products were active when the problem occurred?
  - Availability Plus
  - Dynamic Menus
  - MultiCICS
  - NetGate
  - NetMizer
  - NonStop CICS

Is the problem occurring in the TOR or AOR? _____________________________
Is the problem occurring in a production or test region? _____________________
What is the severity of the problem? _____________________________________
What are the major symptoms of the problem? ______________________________

Is the problem re-creatable under specific conditions? ______________________
Has the problem occurred more than once? _______________________________
Were changes made to CICS or PIE/CICS immediately prior to the occurrence of the problem? ________________________________
What other software products were running when the problem occurred? ______

Is a diagnostic message produced when the problem occurs? If so, what is the ID and text of the messages? __________________________
Does an abend occur? If so, what are the abend and return codes? __________
Is a dump produced when the problem occurs? If so, what kind of dump is it? __

Please try to be as accurate and complete as possible in answering these questions. Your problem can be resolved more quickly if a customer service representative has all of the pertinent information needed to find a solution.
Appendix B Exit and System Table Worksheets

When you upgrade to new release of PIE/CICS, you need to know which exits and system tables were modified to continue using them with the new release. Use the following tables to record any modifications to PIE/CICS exits or system tables.

**Stand-alone Exits**

You can check the member of the same name in the PIE SAMPLIB dataset.

<table>
<thead>
<tr>
<th>Member</th>
<th>Name</th>
<th>Date Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC@UPVUD</td>
<td>USERDATA Validation Exit</td>
<td></td>
</tr>
<tr>
<td>PC@XPSEC</td>
<td>External Security Exit</td>
<td></td>
</tr>
<tr>
<td>PCMRPCRU</td>
<td>Dynamic Transaction Routing Exit</td>
<td></td>
</tr>
<tr>
<td>PCSCPCAN</td>
<td>Session Shutdown Exit</td>
<td></td>
</tr>
<tr>
<td>PCSMPFIL</td>
<td>PIE Environment Command Exit</td>
<td></td>
</tr>
<tr>
<td>PCSSPAEE</td>
<td>AutoEnd Exit</td>
<td></td>
</tr>
<tr>
<td>PCSSPTOE</td>
<td>Time-out Exit</td>
<td></td>
</tr>
<tr>
<td>PCSSSEXT</td>
<td>TSQ Substitution Exit</td>
<td></td>
</tr>
<tr>
<td>PCSTPPVI</td>
<td>Signon Verification Exit</td>
<td></td>
</tr>
<tr>
<td>PCTSPDCX</td>
<td>Logon Message Exit</td>
<td></td>
</tr>
<tr>
<td>PNC@PLPE</td>
<td>PASS/TSWITCH Exit</td>
<td></td>
</tr>
<tr>
<td>PNDMPEXT</td>
<td>Network Monitor Inquiry Exit</td>
<td></td>
</tr>
</tbody>
</table>

You can check the member of the same name in the PIE MACLIB dataset.

<table>
<thead>
<tr>
<th>Member</th>
<th>Name</th>
<th>Date Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUXCMMD</td>
<td>PIE Application Command exit</td>
<td></td>
</tr>
<tr>
<td>PUXINIT</td>
<td>Post Signon Initialization exit</td>
<td></td>
</tr>
<tr>
<td>PUXMNUL</td>
<td>Menu Line Access exit</td>
<td></td>
</tr>
<tr>
<td>PUXOBS</td>
<td>PIE Object Access exit</td>
<td></td>
</tr>
<tr>
<td>PUXVARS</td>
<td>Variable Substitution exit</td>
<td></td>
</tr>
<tr>
<td>PUXVIEW</td>
<td>View Command Access exit</td>
<td></td>
</tr>
<tr>
<td>PUXVSCN</td>
<td>View Screen Access exit</td>
<td></td>
</tr>
</tbody>
</table>
System Tables

You can check the member of the same name in the PIE MACLIB dataset.

<table>
<thead>
<tr>
<th>Member</th>
<th>Name</th>
<th>Date Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALIASTBL</td>
<td>Alias Table</td>
<td></td>
</tr>
<tr>
<td>ATTACHX</td>
<td>Attach Table</td>
<td></td>
</tr>
<tr>
<td>CANTABLE</td>
<td>Cancel Table</td>
<td></td>
</tr>
<tr>
<td>FIELDMAP</td>
<td>Field Map Table</td>
<td></td>
</tr>
<tr>
<td>TERMAK</td>
<td>Terminal Mask Table</td>
<td></td>
</tr>
<tr>
<td>TSQFLDS</td>
<td>TSQ Fields Table</td>
<td></td>
</tr>
<tr>
<td>USERFLDS</td>
<td>User Fields Table</td>
<td></td>
</tr>
</tbody>
</table>
Appendix C  Software Compatibility

This appendix describes PIE/CICS compatibility issues with third-party vendor software that operate on the same systems that PIE/CICS is installed.

CEDF from IBM

Do not use CEDF with any PIE program or transaction. This prohibition includes automatically initiated transactions, such as PEXE which is invoked when the users presses a system end key or when AutoEnd is invoked.

DADS from Computer Associates or Equivalent

If you use NetGate, NonStop CICS, or Availability Plus, doing MVS loads from the DFHRPL causes problems for software products like DADS.

Complete the following procedure if your site runs the Network Monitor and DADS, or an equivalent product.

1. Define the PIE/CICS LOADLIB twice in the CICS start up JCL, using another ddname for the second definition.

2. Specify the second DDNAME in the MVS Loaded Programs DDNAME field of the Customization Options utility.

FETCH from Axios

Initialize PIE/CICS before FETCH in your PLTPI.

GENER/OL from Pansophic

GENER/OL keeps a table of CICS operator IDs in a single temporary storage record. The CICS OPID field must be defined as SWAP in the PIE/CICS TSQFLDS table to support this table.

You must change the following line in the TSQFLDS table:

```
PG@RMFLD NAME=TCTTEOI,BASE=DFHTCTTE,COND=NOSWAP,CSECT=TSQID
```

to:

```
PG@RMFLD NAME=TCTTEOI,BASE=DFHTCTTE,COND=SWAP,CSECT=TSQID
```

See “TSQFLDS and TERMAK: Substitute Temporary Storage Queues,” on page 186 for more information.
IDEAL and ETC from ADR

IDEAL and ETC use the CICS operator ID to create temporary storage queue names. To support this naming convention, the CICS OPID field must be defined in the PIE/CICS TSQFLDS table as NOSWAP. The TSQFLDS table must include the following line:

```
PG@RMFLD NAME=TCTETI,BASE=DFHTCTTE,COND=NOSWAP,CSECT=TSQID
```

This is the default. See “TSQFLDS and TERMASK: Substitute Temporary Storage Queues,” on page 186 for more information.

INTERTEST from On-Line Software and EZTEST from Computer Associates

Under PIE/CICS, always monitor processes at the transaction or program level. Specify the transaction code or program name to monitor or set break points. Do not specify any PIE transaction codes or programs.

Do not monitor or set break points at the terminal level for a PIE/CICS terminal. This can cause unpredictable results with either product.

MSA

MSA software products use a proprietary security system. This security uses a disk file containing CICS termids. To use MSA products with PIE/CICS, you must use pseudo terminal IDs. Set up the TERMASK table to support the termids you require. See “TSQFLDS and TERMASK: Substitute Temporary Storage Queues,” on page 186 for more information.

NATURAL/NATURAL 2 from Software AG

NATURAL and NATURAL 2 use work buffers. You specify whether the buffers are allocated in CICS temporary storage or in the NATURAL database. If the buffers are allocated in the database, you must use pseudo terminal IDs. See “TSQFLDS and TERMASK: Substitute Temporary Storage Queues,” on page 186 for instructions.

OMEGAMON/CICS from Candle

The EMON transaction hangs if you start OMEGAMON/CICS in a PIE session. Candle PTF OC407R46 must be applied to OMEGAMON/CICS for EMON to function properly in a PIE session.

STABILIZE from Computer Associates

You must exclude four programs from monitoring to run Stabilize with PIE/CICS. Use the Stabilize exclude table and PPTEXCL command to exclude the four programs listed below.

```
PPTEXCL=(DFHCRP,DFHZNEP,PCMRPCRP,PCSSPNEP)
```

Please see the Stabilize documentation for more information.
Appendix D. PIE/CICS API Overview

PIE/CICS has several APIs, which are discussed in various PIE/CICS manuals. This appendix summarizes each API and gives a reference to other PIE/CICS manuals for more information.

- **Logon Director interface**
  Passes sign on data from a site-specific sign on system to the PIE/CICS Logon Director. Use this API to allow users to continue signing on with your local program rather than the PIE/CICS Logon Director screen. See “Receiving sign on Data From Another Application,” on page 8 for more information.

- **Issue PIE Command**
  Issues any PIE/CICS command from a program. This API has many uses. Refer to the PIE/CICS Command Reference for more information.

- **Use PIE/CICS variables**
  Allows you to use any PIE/CICS variable from a user program. See the PIE/CICS Command Reference for more information.

- **User data**
  Allows you to retrieve the value for USERDATA variables 1-5 and the language variable, along with other user information, such as the current session ID and the user’s unique PIE/CICS ID. See the PIE/CICS Command Reference for more information.

- **Repository Load**
  Loads and unload objects to and from the Repository. See page 77, “Repository Load Utility” of the Operation and Administration Manual for more information.

- **Change Sessions menu title**
  Change the title of the Sessions menu. See “Application Program Interfaces,” on page 65 of the MultiCICS Administration Guide for more information. (You can see the Sessions menu whether or not you have MultiCICS. You can also change its title whether or not you have MultiCICS.)

- **Reset users**
  Perform the PIE RESETU command. See PIE/CICS Command Reference for more information.

- **Revalidate a user’s password**
  Revalidates a password. Use this API with sensitive programs that require users to enter a password. When the user enters a password, the application can invoke the API to request PIE/CICS to revalidate it. For ACF2, the password is checked directly with ACF2. For other external security systems, it is done using PCSTPPVI, which normally checks PIE/CICS password data. See member PCSCMPWV of the MACLIB dataset for information on how to code the API.
Appendix D. PIE/CICS API Overview

- Send a SuperMessage
  Sends SuperMessages from CICS application programs. See "Sending a Message From a Program," on page 130 for more information.

- TCTTE User Area
  Extracts or updates the TCTTE user area for switched out PIE sessions. Use this API to perform clean up processing at log off time in DFHZNEP or in the PIE/CICS Shutdown exit. See member PCSSMUAA of the MACLIB dataset for more information.

- Multi-Lingual
  Your application programs can retrieve or change language characters using the Multi-Lingual API. See page 19, “Using the Multi-Lingual API”, of the Custom Menus Administration Guide for more information.
Appendix E. PIE/CICS Messages

PIE messages are stored in tables that are members of the SAMPLIB dataset. These tables can be edited to change the text of messages. You can also route messages to the console or PIE message log in addition to the terminal.

If your site needs to provide messages in several different languages, use PIE/CICS Custom Menus. In addition to messages, Custom Menus provides alternate language support to build a complete system of online screens in different languages. Refer to the Custom Menus Administration Guide for more information.

Changes to the PIE Repository are logged to the PIEL log defined in the DCT. A record is written to the log each time an update is made to a PIE menu, text file, log on screen, profile, User or Terminal directory. A log record shows:

- Object type and name
- Date and time when the change to the object was made
- User who made the change
- ID of the terminal from which the change was made

Network Monitor performance data and many PIE/CICS error messages are recorded to the PIEL log. The PIEL log is a critical debugging aid. See “Step 4: Add Entries to CICS Tables,” on page 6 of the PIE/CICS Installation Guide for information to prepare the PIEL log.

The name of the log can be changed by editing the Message Destination Id field of the Customization Options utility.
Locating Message Source Members

Before changing the text of a message, you must first determine the source member where the message is located and its numerical ID: Message tables are members of the SAMPLIB dataset. The format of the message member name is PtttTMSE.

1. Enter **SET SMSG ON** from the Sessions menu to set the short form of messages.

2. Invoke the message and have it appear on the screen.

   Short messages appear in the upper right corner of the screen (any session screen) in the following format:

   ttt nnn.pppooo

   The message table name is identified by the ttt variable. For example, if ttt is C@A, the message is located in the PC@ATMSE table. The message number is nnn. The final variable, pppooo, is for technical support use. The text of the message is not displayed using the short form of a message.

   For example, you could get the following message with short message on:

   CSM008.INI016

   Message 008 is located in the PCSMTMSE member of the SAMPLIB dataset.

3. **Enter SET SMSG OFF from the Sessions menu to turn off short messages.**
Editing Messages

PIE message tables are built with PIEMSG macros. All tables are new-copyable. Keep the following requirements in mind when you make changes to the tables:

- Follow assembler coding conventions and syntax.
- Do not add new messages or change the order of macros in the table.
- You may change only the text of a message. Do not change any of the following:
  - Message numbers
  - Message variables (variables begin with @)
  - Sequence of variables, although you may remove the last variable in a message
- If you increase the length of an existing message, the screen field may not be long enough to display the complete message. Messages are truncated on the right if they exceed the maximum length of the screen display field. The text for logon screen messages can be up to 60 characters. The maximum length for all other messages is 79 characters.

Complete the following procedure to edit a message table:

1. **Edit the appropriate PxxxTMSE member of the SAMPLIB dataset.**
2. **Change the text of the message.**
   
   Keep in mind the points above. Refer to “Example of a PIE Message Table,” on page 176 for examples of text changes to a message table.
3. **Add the DEST parameter to send the message to an additional destination. By default, messages are routed only to the terminal.**
   
   DEST=LOG Sends the message to the PIE message log.
   
   DEST=WTO Sends the message to the console.
   
   Multiple destinations can be coded with a single DEST statement. Include a comma between destinations in the form (LOG,WTO).
4. **Reassemble the table using JCL from the PSMGTAB member of the CNTL dataset.**
   
   PSMGTAB requires customization. The comments in the JCL describe required and optional parameters. For a detailed description of the parameters, see “JCL Parameters,” on page 197.
Appendix E. PIE/CICS Messages

Example of a PIE Message Table

The following example is a sample message table. Message number 1 displays on the screen as:

Invalid scroll default amount, please re-enter

Message 2 includes the "@X" variable; a two position hexadecimal number. If the return code is X'A2', the return code is inserted and the message appears on the screen as “Invalid return code=A2".

Message number 8 contains additional destinations. The message is sent to the PIE message log and the operator console.

PC@LPMSG PIEMSG TYPE=INITIAL
PIEMSG 1,'Invalid scroll default amount, please re-enter'
PIEMSG 2,'Invalid return code=@X'
PIEMSG 3,'Invalid scroll amount, please re-enter'
PIEMSG 4,'Invalid line command, please re-enter'
PIEMSG 5,'Scroll amount is limited to 3 digits'
PIEMSG 6,'Resource queue access failure'
PIEMSG 7,'Invalid command, please re-enter'
PIEMSG 8,'PIE @CCCCCCC initialization complete on region @CCX CCCCC’,DEST=(LOG,WTO)
PIEMSG TYPE=FINAL
Appendix F. Cutting and Pasting Across Regions

If you use either the TSWITCH or PASS commands to switch between CICS sessions, some customization is required before you can cut and paste data from one region to another. Complete the following procedure to cut and paste data between disconnected sessions opened with the TSWITCH or PASS commands. If you use the NetGate ACCESS command, cut and paste operations are functional and no further customization is required.

1. **Edit the PIETST member of the SAMPLIB dataset.**
   
   PIETST creates a CICS Temporary Storage Table (TST). This table converts all PIE temporary storage queue requests to remote requests.

2. **Find the TYPE=INITIAL macro statement.**

3. **Supply a suffix for the DFHTSTxx table in the SUFFIX parameter.**

4. **If you have changed the PIE/CICS TSQ prefix using the Temp Storage Queue Prefixes field of the PIE Customization Options utility, locate the TYPE=REMOTE macro and change the DATAID parameter.**

5. **Assemble the TST using standard CICS procedures.**

6. **Recycle CICS to initialize the Temporary Storage Table.**
Appendix G. PIE/CICS System Tables

The default values shipped with PIE/CICS system tables are often sufficient for most installations and do not require any further customization. However, these tables must be re-assembled whenever IBM maintenance is applied to CICS, or when you migrate to a new release of CICS.

PIE/CICS system tables are listed below. The tables are composed of macro statements assembled using the JCL in member PG@RPCSC. Each table is described separately in the following sections of this appendix.

<table>
<thead>
<tr>
<th>Name</th>
<th>MACLIB Member</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alias Table</td>
<td>ALIASTBL</td>
<td>Renames PIE programs. Renaming a PIE program is necessary only if a PIE program name conflicts with another program with the same name.</td>
</tr>
<tr>
<td>Attach Table</td>
<td>ATTACHX</td>
<td>Executes a different transaction or PIE command for a transaction. This is normally used to process CESN/CESF transactions properly under PIE.</td>
</tr>
<tr>
<td>Cancel Table</td>
<td>CANTABLE</td>
<td>Specify transactions that must be ended normally, not canceled by a user or PIE/CICS.</td>
</tr>
<tr>
<td>Field Map Table</td>
<td>FIELDMAP</td>
<td>Defines CICS control block fields to PIE that are subject to change with CICS maintenance (APARs and PTFs). Do not change this table.</td>
</tr>
<tr>
<td>Terminal Mask Table</td>
<td>TERMASK</td>
<td>Creates unique terminal IDs for each session.</td>
</tr>
<tr>
<td>Temporary Storage Queue Fields Table</td>
<td>TSQFLDS</td>
<td>Defines CICS fields that must be unique for each session, such as terminal ID and operator ID. For example, edit TSQFLDS to add fields that are used as part of a temporary storage queue name or in a work table. Define whether you want those fields to be changed in the TSQ name or in the CICS control block.</td>
</tr>
<tr>
<td>User Fields Table</td>
<td>USERFLDS</td>
<td>Defines CICS control block fields maintained for each session that are modified by programs.</td>
</tr>
</tbody>
</table>

When you upgrade to another release of PIE/CICS, you will need to know which tables have been modified so that you can transfer the modifications to the new release. If you modify a table, record the changes on the worksheet shown on page 168.
Modifying System Tables

Again, as shipped, the System Tables are appropriate for most PIE/CICS installations. However, if you need to change a table, complete the following procedure.

1. **Edit the table source member in the MACLIB dataset.**
   
   The MACLIB member name is the table’s short name—for example, ALIASTBL or CANTABLE. Refer to the section describing the table for a syntax diagram that shows the format of the macro.

   If you change the name of a table, you must change the table name referenced in the System Tables source member.

2. **Edit the PG@RPCSC member of the SAMPLIB dataset.**
   
   Locate the COPY statement for the table name you want to change and supply the new name.

3. **Reassemble the tables using the job within the PG@RPCSC member of the CNTL dataset.**
   
   PG@RPCSC requires some customization. Comments in the JCL describe required and optional parameters that may need to be changed. For a more detailed description of these parameters, see “JCL Parameters,” on page 197.

4. **Recycle CICS to initialize the modified tables.**
ALIASTBL: Rename PIE/CICS Programs

ALIASTBL can be used to rename most PIE/CICS programs. Normally, this table is used to rename a PIE/CICS program that has the same name as another program running on your system.

Change the Alias Table and rename the PIE/CICS module in the LOADLIB dataset to change a program name.

ALIASTBL is composed of PG@RMALI macros. The macro format is:

```
PG@RMALI NAME=old-name, ALIAS=new-name
```

- **NAME**: Current name of the PIE/CICS program.
- **ALIAS**: New program name.

**Example**

This example shows the PG@RMALI macro changing program module PCSSPTSQ to PIETSQXT. PCSSPTSQ must be renamed to PIETSQXT in the PIE/CICS LOADLIB.

```
PG@RMALI NAME=PCSSPTSQ, ALIAS=PIETSQXT
```
ATTACHX: Substitute Transaction IDs

ATTACHX substitutes one transaction for another. You can also substitute a command for a transaction.

The Attach table is used primarily to intercept sign on/sign off transactions (such as CESN and CESF) and substitute PIE/CICS functions. If you want to run PIE/CICS transparently, you must change the default table to prevent the sign on transaction from being substituted with the PIE/CICS sign on transaction.

You can also use the Attach table to create more recognizable aliases for transactions, such as LOGO for CESF or OPER for CEMT.

The Attach table cannot substitute ATI START transactions.

ATTACHX is composed of PG@RMATT macros. The format of the macro is:

```
PG@RMATT TRANID=tran +
 [,INT={YES|NO}] +
 [,SUB={YES|NO}] +
 [,NEWTRAN=tran] +
 [,COMMAND=command'] +
 [,NXTRAN={YES|NO}] +
 [,NOMSG={YES|NO}] +
 [,TIOA={YES|NO}] 
```

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRNID</td>
<td>Tranid of the transaction to be intercepted.</td>
</tr>
<tr>
<td>INT</td>
<td>Choice (Yes or No) to intercept a transaction for PIE/CICS command processing.</td>
</tr>
<tr>
<td>SUB</td>
<td>Choice (Yes or No) to substitute the TRNID transaction with another transaction.</td>
</tr>
<tr>
<td>NEWTRAN</td>
<td>Tranid of the substituted transaction if SUB=Yes.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>PIE Environment command to use if INT=Yes, or specify a new TIOA if INT=No. Remember, you can specify a CICS transaction and its parameters as a PIE Environment command with either CMD or TRAN.</td>
</tr>
<tr>
<td>NXTRAN</td>
<td>Choice (Yes or No) to specify whether the transaction uses the hidden or “next tranid on screen” method of pseudo conversing. If you specify YES, PIE/CICS ignores the system end key and passes the transaction to the application.</td>
</tr>
</tbody>
</table>

You can use this parameter to substitute a transaction and parameters for a transaction ID. For example, the following macro

```
PG@RMATT TRANID=TASK,INT=NO,SUB=YES,
   NEWTRAN=CEMT,COMMAND='AE CMD CEMT INQ TAS'
```

creates an alias, TASK, for the command string CEMT INQ TAS.

The command string is placed at the beginning of TIOA (terminal input/output area).
ATTACHX: Substitute Transaction IDs

**NOMSG** Choice (Yes or No) to exclude this tranid from receiving PIE/CICS SuperMessages.

**TIOA** Choice (Yes or No) to append the contents of TIOA (Terminal Input/Output Area) to the end of command string.

**Examples**

- The following macro substitutes the PIE RETURNA command for the CESF transaction.

  ```
  PG@RMATT TRANID=CESF,INT=YES,SUB=NO,
  COMMAND='RETURNA'
  ```

- The following macro excludes CEDA from receiving SuperMessages.

  ```
  PG@RMATT TRANID=CEDA,INT=NO,SUB=NO,
  NOMSG=YES
  ```

- The following macro changes the default operation of the MAIL transaction. MAIL needs to run without uppercase translation. To make this the default, the PIE attach exit is converted to a PIEXEC command string.

  ```
  PG@RMATT TRANID=MAIL,INT=NO,SUB=YES,
  NEWTRAN=PEXE,COMMAND='PEXE NOUCT NOATTACH AE CMD MAIL'
  ```

  NOUCT is a PIE state command that turns off uppercase translation. NOATTACH, also a PIE state command, prevents a recursive attach of the MAIL transaction.
CANTABLE: Prevent Transactions from Being Cancelled

CANTABLE prevents designated transactions from being cancelled. PIE/CICS cancels an application running in a session whenever:

- The terminal is disconnected from CICS and the value Allow Disc Status=NO is specified in the User Directory. All active sessions are cancelled.
- The user attempts to disconnect with either the EXIT DISC or DISCHOLD commands and Allow Disc Status=NO is specified in the User Directory. All active sessions are cancelled.
- The user logs off with either the EXIT LOGOFF, SIGNOFF, or END PIE commands. All active sessions are cancelled.
- The user ends a session with either the CLOSE or CANCEL command. Only the specified session is affected.

PIE/CICS takes one of the following actions if a transaction has been selected for cancellation, but is listed in CANTABLE as non-cancellable:

- If the user entered a CLOSE or CANCEL command, PIE/CICS switches to the session and displays a message indicating the application must end normally. The user can re-enter a command to end the session or exit PIE/CICS.
- If the terminal was disconnected, PIE/CICS can no longer communicate with the user. In that case, PIE/CICS disconnects rather than logs off the user. The disconnect occurs even if Allow Disc Status=NO is specified in the User Directory.

CANTABLE is composed of PG@RMCAN macros. The macro format is:

```
PG@RMCAN TRANID=tran,CANCEL={YES|NO}
```

- **TRANID** Tran id of the transaction to be cancelled. You can use wildcards to generically select a range of transactions. See “Wildcards,” on page 6.
- **CANCEL** Choice (YES or NO) to cancel transactions.

This value overrides the value set by the Allow Session Cancel field of the User Directory. If you specify CANCEL=NO, the user cannot cancel the transaction, even if the User Directory entry specifies Allow Session Cancel=YES. If you specify CANCEL=YES, the user can cancel the transaction, even if the User Directory entry specifies Allow Session Cancel=NO.

**Example**

The following CANTABLE macro entry prevents PIE/CICS from cancelling the CEDF transaction.

```
PG@RMCAN TRANID=CEDF,CANCEL=NO
```
FIELDMAP: Define CICS Control Block Fields
The FIELDMAP table identifies CICS control blocks fields to PIE/CICS that are subject to change with CICS maintenance (APARs and PTFs).

⚠️ WARNING
Do not change this table unless you are directed to do so by UNICOM Systems, Inc.

TERMASK: Construct Unique Pseudo Termids
See “TSQFLDS and TERMASK: Substitute Temporary Storage Queues,” on page 186.
TSQFLDS and TERMASK: Substitute Temporary Storage Queues

The TSQFLDS and TERMASK tables determine how temporary storage queue (TSQ) names and CICS control block values are made unique for each session.

About TSQ Substitution

Most CICS applications maintain separate TSQs for each user to keep their work independent from other users. Applications create unique TSQ names by adding a user-specific value (almost always the CICS terminal ID or the operator ID) to a character string. For example, an application could add the user's CICS terminal ID to the prefix AAAA, to get queue names like AAAAL012 or T345AAAA.

With MultiCICS, a single user can run the same application in multiple sessions. Because a user has a single terminal and operator ID, the application would be forced to use the same temporary storage name for every session. Work in one session could corrupt the temporary storage of another session. In addition, PIE/CICS allows users to port their work from one terminal to another. If TSQ names are based on terminal ID, applications could not find their TSQs after users port their sign ons to different terminals.

PIE/CICS substitutes TSQ names that are unique for each session. PIE/CICS associates TSQ names to the user ID, not the terminal ID. This is called temporary storage queue substitution.

PIE/CICS uses two methods to substitute TSQ names:

- Change the CICS control block, itself. This is called SWAP substitution, because the control block in memory is swapped with a unique value. The control block is changed each time the user switches to a new session.
- Change the name of the queue when it is requested, before the queue is written or read. This is called NOSWAP substitution because the control block is not swapped; only the TSQ name is changed. This is the default method, which works for the vast majority of installations.

By default, PIE/CICS changes either the terminal ID, the operator ID, or both to a binary value that is unique for each user and session.

- For the four character terminal ID, PIE/CICS substitutes a three character user ID and a one character session ID.
- For the three character operator ID, PIE/CICS substitutes a two character user ID and a one character session ID.

  The operator ID is set by external security. Some security packages do this by default, others do not. The field contains all zeros if it is not set.
Reasons to Change the Default TSQ Substitution Method

There are two reasons to change the defaults:

- Your application uses something other than the terminal or operator ID to create TSQ names. For example, it may use a user-defined field from the TCTUA. In this case, NOSWAP substitution is fine. You simply need to notify PIE/CICS of the field used, so that it can substitute for that field also.

- Your application uses the terminal or operator ID in its processing, not just in the TSQ name. For example, GENER/OL keeps a table of CICS operator IDs in a temporary storage record. MSA software packages keep a work file containing CICS terminal IDs. In this case, SWAP substitution is required. The application needs the CICS control block changed in memory, so that a unique value is available to it directly.

The following sections describe how to support an alternative method of substituting TSQ names.
Defining a New NOSWAP Field

Complete this procedure if your application uses something other than the terminal or operator ID to create TSQ names.

1. **Edit the TSQFLDS table.**
2. **Add a PG@RMFLD macro to define the field.** The macro format is:

   ```
   PG@RMFLD NAME=field-name, BASE=control-block, COND=NOSWAP
   ```

   - **NAME**
     Name of the field used in the TSQ name. For example, specify TCTTETI for the terminal ID field.

   - **BASE**
     Name of the CICS control block (DSECT) that contains the field. For example, the DFHTCTTE control block contains the TCTTETI field.

     - On the first PG@RMFLD macro only, add the CSECT=TSQID parameter.

     Add a PG@RMLOC macro to define how to address the control block. The format of the macro is:

     ```
     PG@RMLOC NAME=control-block,
     ORIGIN=origin, CHAIN=offset
     ```

     - **NAME**
       Name of the control block you specified in the BASE parameter of the PG@RMFLD macro.

     - **ORIGIN**
       Name of the CICS control block. Specify one of the following:

       - **TERM**
         Terminal Control Table (TCT)

       - **SYSTEM**
         Common System Area (CSA)

       - **TASK**
         Task Control Area (TCA)

     - **CHAIN**
       Offset from the origin that points to the control block. For example, to point to the USERAREA in the TCT, specify

       ```
       CHAIN=(TCTTECIA-DFHTCTTE)
       ```

       That is, the offset is the value for TCTTECIA minus the value for DFHTCTTE.

3. **Add the DSECT for the control block to the TSQFLDS table.**

   - If you do not have a DSECT, see “Using the PG@RMFLD and PG@RMLOC Macros Without a DSECT,” on page 195, for instructions.

4. **Reassemble the table as described in “Modifying System Tables,” on page 180.**

5. **Edit the Customization Options utility.**

   Ensure that YES is specified for the Temp storage name substitution field. YES is
required in the Temp storage name substitution field if there is any NOSWAP macro in the TSQFLDS table.

There are two methods to implement SWAP substitution:

- The first method swaps binary, non-displayable characters into the field. This method is simpler, but it prevents systems personnel from using or looking up the field online. For example, they cannot access the terminal ID using CEMT. Use it solely for fields, like the operator ID, that are accessed only programmatically. This method is described in the following section.
  
  In CICS 4.1, you cannot use this method for the OPID. You must use NOSWAP substitution or the following method for swapped, displayable OPIDs.

- The second method swaps user-defined, displayable characters. Use this method for the terminal ID and any other field that will be displayed. This method is described in “Using Displayable Swap Substitution,” on page 191.
Using Non-Displayable Swap Substitution

To implement SWAP substitution for the operator ID or any other field that is not displayed, do the following.

1. **Edit the TSQFLDS table.**

2. **Add a PG@RMFLD macro to define the field.** The format of the macro is:

   ```
   PG@RMFLD NAME=field-name,BASE=control-block,COND=SWAP
   ```

   - **NAME** Name of the field to swap. For example, specify TCTTETI for the terminal ID field. The maximum length for this field is four characters. If the field is more than four characters long, break it up into multiple fields.
   - **BASE** Name of the CICS control block (DSECT) that contains the field. For example, the DFHTCTTE control block contains the TCTTETI field.
   - On the first PG@RMFLD macro only, add the parameter `CSECT=TSQID`.
   - Add a PG@RMLOC macro to define how to address the control block. Macro format is:

     ```
     PG@RMLOC NAME=control-block,ORIGIN=origin,CHAIN=offset
     ```

     - **NAME** Name of the control block you specified in the BASE parameter of the PG@RMFLD macro.
     - **ORIGIN** Specify TERM if the field is from the TCT, SYSTEM if it is from the CSA, or TASK if it is from the TCA.
     - **CHAIN** Offset from the origin that points to the control block. For example, to point to the USERAREA in the TCT, specify

       ```
       CHAIN=(TCTTECIA-DFHTCTTE)
       ```

       That is, the offset is the value for TCTTECIA minus the value for DFHTCTTE.

3. **Add the DSECT for the control block to the TSQFLDS table.**

   - If you do not have a DSECT, see “Using the PG@RMFLD and PG@RMLOC Macros Without a DSECT,” on page 195, for instructions.

4. **Reassemble the table as described in “Modifying System Tables,” on page 180.**

   - See the TSQFLDS table for examples.
Using Displayable Swap Substitution

This method implements SWAP substitution for the terminal ID. This is also called creating pseudo terminal IDs. The field will be displayable. You can use this method for any field you need to display or request online.

Complete the following procedure to implement displayable swap substitution.

1. **Edit the TERMASK table.**

   There are two PG@SMSUB macros coded in this table. One does no substitution; the other turns on substitution using the first position in the terminal ID. Because the first match is used and the first macro matches all terminal IDs, by default, no substitution is taking place.

   The easiest way to implement substitution is to comment out the first macro. Then, PIE/CICS uses the second macro to change terminal IDs.

   If you want to do the substitution differently or for a different field, change or add a macro using the instructions in “TERMASK Macro Format,” on page 191.

2. **Save your changes to the TERMASK table.**

3. **Edit the TSQFLDS table.**

4. **Remove the terminal ID from the TSQFLDS table by commenting out both of the following macros:**

   ```
   PG@RMFLD NAME=TCTTETI,BASE=DFHTCTTE,COND=NOSWAP, X
   CSECT=TSQID
   PG@RMLOC NAME=DFHTCTTE,ORIGIN=TERM
   ```

5. **Add the CSECT=TSQID parameter to the PG@RMFLD macro defining TCTTEOI:**

   ```
   PG@RMFLD NAME=TCTTEOI,BASE=DFHTCTTE,COND=NOSWAP, X
   CSECT=TSQID
   ```

6. **Save your changes to the TSQFLDS table.**

7. **Reassemble the tables as described in “Modifying System Tables,” on page 180.**

TERMASK Macro Format

The TERMASK table consists of PG@SMSUB macros. The format of the macro is:

```
PG@SMSUB SUBMASK=(real-termid,positions),SnMASK=values
```

- **real-termid** Terminal IDs to swap in the control block. You may use wildcards *, ?, and % to generically select a range of terminal IDs. See “Wildcards,” on page 6.

- **positions** Hex mask for the terminal ID field. Specify FF in the positions to modify. Specify 00 in the fields to leave as is.

  For example, the following macro substitutes all terminal IDs (*) and modifies bytes 1 and 4 (X'00FF0000FF').

  ```
  PG@SMSUB SUBMASK=(*,X'00FF0000FF')
  ```

- **SnMASK** Specify one SnMASK parameter for each MultiCICS session: S1MASK,
S2MASK, S3MASK, ..., S99MASK. For values, specify the values to substitute for x’FF’ in positions, above. If you do not want a position changed for a particular session, specify x’00’ for that position.

Example of the TERMAK Macro

This example changes the terminal ID ABCD as follows: unchanged for session 1, changed to ABC2 for session 2, to 3BCC for session 3, to 4BCD for session 4, etc.

S1MASK=X’0000’,  <= DON’T CHANGE FOR SESSION1
S2MASK=X’00F2’,  <= DON’T CHANGE FIRST POSITION FOR SESS2
S3MASK=C’3C’,    <= CHANGE FIRST TO 3, LAST TO C
S4MASK=C’4D’,    <= CHANGE FIRST TO 4, LAST TO D
S5MASK=C’5E’,    <= CHANGE FIRST TO 5, LAST TO E
S6MASK=C’6F’,    <= CHANGE FIRST TO 6, LAST TO F
S7MASK=C’7G’,    <= CHANGE FIRST TO 7, LAST TO G
S8MASK=C’8H’,    <= CHANGE FIRST TO 8, LAST TO H

Using this example, the complete PG@MSUB macro would look like the example shown below:

PG@MSUB  SUBMASK=(*,X’FF0000FF’)
  S1MASK=X’0000’,
  S2MASK=X’00F2’,
  S3MASK=C’3C’,
  S4MASK=C’4D’,
  S5MASK=C’5E’,
  S6MASK=C’6F’,
  S7MASK=C’7G’,
  S8MASK=C’8H’,
USERFLDS: Maintain Control Block Fields

Some applications use fields in the Terminal Control Table User Areas (TCTUA) and other CICS control blocks as a work area to pass data from one transaction to another. When applications run simultaneously in multiple sessions, there is a possibility that different applications could use the same field and corrupt each others' storage.

To prevent this from occurring, PIE/CICS can maintain separate control block fields for each session. PIE/CICS saves and restores the values stored in the fields each time the user switches between sessions.

While it is usually difficult to predict which applications and fields are affected, you may encounter some such situations during the testing phase. As you discover them, you should add these fields to the USERFLDS table. Most CICS control block fields are not changed by applications. They are, and should be, global for all sessions. Do not add these fields to the USERFLDS table.

The USERFLDS macros are similar to the TSQFLDS macros. However, they have a very different purpose. The TSQFLDS table controls values that must be unique from session to session but that are not altered by an application. The application simply expects them to be unique. So PIE/CICS, itself, creates unique values for each session. In contrast, the USERFLDS table controls values that are changed by the application. PIE/CICS does not set these values; it simply maintains them.

Adding Fields to the USERFLDS Table

USERFLDS can maintain any field in the TCT, TCA, or CSA, up to 4 bytes long. To add a field to the USERFLDS table, do the following.

1. Edit the USERFLDS table.
2. Add a PG@RMFLD macro for each field. This macro defines the field to maintain. The macro format is:

   ```plaintext
   PG@RMFLD NAME=field-name,BASE=control-block
   ```

<table>
<thead>
<tr>
<th>NAME</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>Name of the terminal ID field. The maximum field length is four characters. If the field is more than four characters long, break it up into multiple fields.</td>
</tr>
<tr>
<td>BASE</td>
<td>Name of the CICS control block (DSECT) that contains the field. For example, the DFHTCTTE control block contains the TCTTETI field.</td>
</tr>
</tbody>
</table>

   On the first PG@RMFLD macro only, add the parameter CSECT=USER.

   Add a PG@RMLOC macro to define how to address the control block. Macro format is:

   ```plaintext
   PG@RMLOC NAME=control-block,ORIGIN=origin,CHAIN=offset
   ```

<table>
<thead>
<tr>
<th>NAME</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>Name of the control block specified in the BASE parameter of the PG@RMFLD macro.</td>
</tr>
<tr>
<td>ORIGIN</td>
<td>Specify TERM if the field is from the TCT, SYSTEM if it is from the</td>
</tr>
</tbody>
</table>
Appendix G. PIE/CICS System Tables

CSA, or TASK if it is from the TCA.

CHAIN Offset from the origin that points to control block. For example, to point to the USERAREA in the TCT, specify

CHAIN=(TCTTECIA-DFHTCTTE)

That is, the offset is the value for TCTTECIA minus the value for DFHTCTTE.

3. Add the DSECT for the control block to the TSQFLDS table.

If you do not have a DSECT, see “Using the PG@RMFLD and PG@RMLOC Macros Without a DSECT,” on page 195, for instructions.

4. Reassemble the table as described in “Modifying System Tables,” on page 180.

Examples

The following PG@RMFLD macros specify that fields FIRST4 and SECND3 should be maintained for each session. They are located in the TCTUA control block.

PG@RMFLD NAME=FIRST4, BASE=TCTUA
PG@RMFLD NAME=SECND3, BASE=TCTUA

The following PG@RMLOC macro defines how to address the TCTUA control block. The ORIGIN=TERM parameter defines the beginning control block base is the terminal control table entry (TCTTE). The CHAIN=(TCTTECIA-DFHTCTTE) defines the offset of the control block within the TCTTE.

PG@RMLOC NAME=TCTUA, ORIGIN=TERM, CHAIN=(TCTTECIA-DFHTCTTE)
Using the PG@RMFLD and PG@RMLOC Macros Without a DSECT

The TSQFLDS and USERFLDS tables use the PG@RMFLD and PG@RMLOC macros. Normally, a DSECT is used to define fields and control blocks with these macros. If you do not have a DSECT, you can still define the fields with both macros.

The following syntax diagram shows alternative parameters used with the PG@RMLOC and PG@RMFLD macros to define table fields without a DSECT.

```
PG@RMFLD NAME=offset,LENGTH=n,BASE=0,ORIGIN=control-block

NAME     Offset from the beginning of the ORIGIN to the field. For example, specify 4 if the field begins four bytes into the control block.
LENGTH    Length of the field in bytes.
BASE      Specify 0.
ORIGIN    Name of the control block. For instance, if the field is in the TCTUA, specify TCTTEUA.

For the first PG@RMFLD macro only, add the parameter
CSECT=USER.

The PG@RMLOC macro looks the same. The only difference is in the
NAME parameter—specify the value you specified for the ORIGIN (not BASE) parameter in the PG@RMFLD macro. The format of the macro is:

```
PG@RMLOC NAME=control-block,
ORIGIN=origin,CHAIN=offset
```

NAME     Name of the control block specified in the ORIGIN parameter of the PG@RMFLD macro.
ORIGIN    Name of the CICS control block. Specify one of the following:
TERM      Terminal Control Table (TCT)
SYSTEM    Common System Area (CSA)
TASK      Task Control Area (TCA)
CHAIN     Offset from the origin that points to control block. For example, to point to the USERAREA in the TCT, specify
CHAIN=(TCTTECIA-DFHTCTTE)
That is, the offset is the value for TCTTECIA minus the value for DFHTCTTE.
See the USERFLDS table for an example.
Appendix H. JCL Parameters

This appendix describes common JCL parameters used by many of the PIE batch jobs described in this manual. These jobs are stored as members of the CNTL dataset. Normally, you edit these jobs and substitute the specific value used by your site for each of the parameters shown in the following list.

All jobs contain short descriptions of each parameter. See the job comments for more information, or contact Technical Support if you need information about a parameter not described in this appendix.

- **OUTC**: SYSOUT class for job output. The default is *.
- **WORK**: UNIT class of temporary datasets. The default is VIO.
- **ASMBLR**: Name of the assembler program to use. For Assembler H, it is normally IEV90. For standard Assembler, it is IFOX00. The default is IEV90.
- **CICSQ**: High-level qualifier of CICS library datasets. The default is SYS2.CICS410.
- **CICSLOD**: Low-level qualifier of the CICS load library.
- **CICSSRC**: Low-level qualifier of the CICS source dataset.
- **CICSMAC**: Low-level qualifier of the CICS MACLIB.
- **CICSMITS**: Low-level qualifier of the CICS MTS (SMP/Macro Temporary Store) dataset if you have applied maintenance to CICS but have not accepted it. If not, specify CICSMITS=MACLIB. The default is MTS.
- **Q1**: High-level qualifier of PIE/CICS datasets. The default is PIE.
- **Q2**: Mid-level qualifier of PIE/CICS datasets. The default is PC@310.
- **MACLIB**: Mid-level qualifier of the PIE/CICS macro library. The default is MACLIB.
- **SYSIN**: Low-level qualifier of the PIE/CICS sample library. The default is SAMPLIB.
- **LOAD**: Low-level qualifier of the PIE/CICS load library. The default is LOADLIB.
- **MEMBER**: Name of the source member to be assembled. The member must be in the PIE SAMPLIB. The default is usually the same as the job name.
- **SECMAC**: Full name of the security system macro library if you use ACF2 or OMNIGUARD. Otherwise, the default is SYS1.MACLIB.
- **SECURITY**: Name of the external security system: RACF, TOPS, ACF2, OMNI, or NONE. The default is RACF.
Index

Symbols
$$PIEPAN sample Logon Director screen 22, 23
% wildcard
   use with Terminal and User Directories 45
&INFO variable 19, 23
&NEWS variable 19, 23
&ZSYS variable 126
&ZSYS(sysid) variable 126
&ZUDATA variable 57, 132
* wildcard
   use with Terminal and User Directories 45
? wildcard
   use with Terminal and User Directories 45
@APPL (RES) macro statements 122

Numerics
16 Mb line
   DSA storage usage by transactions 154
3270 screen
   specifying terminal alarm 70
3270 terminal
   Logon Director fields 19
   7171 Device Attachment Control Unit 50

A
ACBNAME
   PIENM001 125
access
   exit ??– 78
   security rules 75
access method
   Dynamic Transaction Routing 121
ACF2
   menu line security 70
   securing PIE transactions 80
   sign on appendage program 79
ALIASTBL system table 181
AORs
   over 200 122
Application Program Interface (API)
   Logon Director 9
Application Programming Interface (API)
   overview 171–172
APPLID start-up option 123, 125
APPLPOL start-up option 123, 125
ASMBLR JCL statement 197
ATTACHX system table
   substituting transaction IDs 182–183
   use with transparent mode 92
   using to bypass transactions 15
audience ix
AutoEnd
   PCSSPAE2 exit 136
   PCSSPAEE exit 136
AutoLock
   disabling for transparent mode 91
automated sign on methods
   sign on
      automated 6
autoskip screen fields 20
Availability Plus
   TCTTE user area 16-byte field used for
      load balancing by CPU usage 68
AVSTAT start-up option 126

B
book conventions xiv
bulletin
   default 57

C
CANTABLE system table 184
CEDA transaction 183
CEDF 169
CEDF transaction 184
CEDF1 169
CESF transaction 92, 183
CESN transaction 93
CICS
   Class Max feature with PIE/CICS
      transactions 154
   defining control block fields with the
      FIELDMAP system table 185
   DFH2014 message 136
   DFHDCT table 69
modifying SIT for PASS and TSWITCH commands 113
modifying TCT for PASS and TSWITCH commands 113
temporary storage table 155
CICSLOD JCL statement 197
CICSMAC JCL statement 197
CICSMTS JCL statement 197
CICSQ JCL statement 197
CICSSRC JCL statement 197
Class Max feature 154
CLSTAT start-up option 126
commands
assigning delimiters with the
  Customization Options utility 70
authorization by security classes 59
changing parameters of Application
  commands with the PUXCMMD exit 148
classes 30
Customization Options utility 72
CUT 177
delimiters 33, 37
final PIE 32, 36
HELPDESK 57
initial PIE 32
LOCK 85
modifying Environment commands with
  the PCSMPFIL exit 135
PASTE 177
PRINTID 58
PROFILE 30
REFRESH 21
restricting for transparent mode 90
RETRIEVE 32
SET 37, 38
SKIPNEXT 39
SYSID 120
TOGGLE command
  commands
    TOGGLE 39
User Directory 62–63
USERDATA 117, 132
validate transaction IDs 71
contacting Customer Service 164
CRTE transaction 120
CSGM program 8
CSSN transaction 15
Custom Menus
  Multi-Language Characters field of
    Customization Options utility 71
    setting default language with the User
Directory 59
Customization Options utility
  commands 72
fields
Action on max session 71
Action on session close 72
AOR AE/ATI support-1 byte 68
Automatic PIE GMM Tran 68
Auto-start Network Monitor 69
Auto-start Terminal Security Monitor 69
Avg No. of Concurrent Users 67
Avg No. of Sessions per User 67
Changeable Token Delimiters 70
Create TCTTE Extension 68
Dynamic Trans Routing 70
Extract VTAM by Logon Dir 71
Generic Matching on Menus/Sessions 71
Generic matching on Menus/Sessions 71
Handle Unavailable Printer in NEP 71
Limit Transactions to One Session 71
Load Balancer 16-byte 68
Maximum Logon Attempt Count 66
Menu Auto-select Single Line 70
Menu Line Security Option 70
MRO/ISC support 67
Multi-Language Characters 71
MVS Loaded Programs DDNAME 70
Name of Logon Director Help 68
Signon Message Display Time 69
Temp Storage Name Substitution 67
Temp Storage on Auxiliary 68
Temp Storage on Auxiliary 68
Temp Storage Queue Prefixes 68
Terminal Alarm on Cut/Paste 70
Tranid Validation by Session Mgr 71
Transient TCTTEs 68
TSWITCH
  setting time-out period 67
Maximum Logon Attempt Count field 66
  setting security verification interval 70
  specifying command delimiters 70
CUT command 177
cut operation
    customizing for sessions opened with
    the TSWITCH or PASS commands 177
    setting PF keys 38
    specifying terminal beep 70

D
DADS 70, 169
DASD I/O
tuning 157
default application 32, 35
delimiters 33, 37
DFHGMM good morning message program 8
DFHGMMCICS good morning program 4
DFHSIT table
    setting good morning message 8
dial-up port indicator
    in the Terminal Directory 49
displayable TSQ swap substitution 191
Dynamic Menus
    auto-select single option menus 70
displaying protected menu lines to
e unauthorized users 70
generic name matching of menu options 71
    security verification interval 70
Dynamic Transaction Routing
description 119
functions 119
PCMRPCRU exit 133
preparation 119–120
specifying with the Customization
    Options utility 70
using the PCSSPAEE exit to change
    AutoEnd to Confirm End 136

E
editing authority 34
EMON transaction 170
encrypted password 10
escape string 33, 37
ETC 170
exits
    access authorization 150
    Dynamic Transaction Routing 119,
        133
    external security system 79
ICHRIX02 RACF post-processing 79
Logon Message 116
Logon Message Analysis 140
PCSSPAE2 136
PUX1
    PUXCMMD 148
    PUXINIT 149
    PUXMNU 150
    PUXOBJ 150
    PUXVARS 151
    PUXVIEW 151
    PUXVSCN 152
PUXCMMD 71, 90
PUXINIT 46, 85
resource access verification exit 132
stand-alone
    PC@UPVUD 132
    PC@XPSEC 132
    PMRPRCU 133
    PCSCPCAN 134
    PCSMPFIL 135
    PCSPCAN 157
    PCSSPAE 136
    PCSSPTOE 58, 137
    PCSSXEXT 67, 138
    PCSTPPVI 4, 79, 139
    PCCTSPDCX 10, 116, 140
    PCCTSPDCX exit 46
    PNC@PLPE 142
    PNDMPFIL 142
    PO@MNUL 79
    PO@XPSEC 79
    PUXINIT 4
    worksheet 167
    tuning 160
EZTEST 170

F
FETCH 169
FIELDMAP system table 185
final PIE command 32, 36

G
GENER/OL 169, 187
generic name matching of menu options 71
global PIE keys
    setting for transparent mode 92
    specifying with the Profile utility 33
temporary storage queue creation 33
good morning program 8
good morning transaction 68
use with automated log ons to target regions 113
Greenwich time 10
group administrator
description 75
granting authority through the User Directory 58
grupid mask 55
GRPNENT start-up option 126

H
help
PIETLDIR text member 27
specifying name of Logon Director help member 68
HELPDESK command 57
hidden sessions 40

I
IDEAL 170
initial PIE command 32
initial session configuration 32
INQUIRE
return codes 126
INTERTEST 170
ISC
routing 133
setting PIE/CICS support 67
ISTAT start-up option 126

J
J CL
parameters 197

L
last user to update an object 76
load balancing
TCTTE user area 16-byte field used by Availability Plus 68
LOAD J CL statement 197
LOCK command 85
log off
preventing 48
log on

message
extraction 141
logging messages 173
Logon Director
$$PIEPAN sample screen 22
actions at sign on 4
Application Program Interface (API) 9
assigning as the good morning transaction 8
assigning to terminals 27
authorization to customize screen 18
automated sign ons 6
default PIEMAIN screen 48
default screen 7
editing a log on screen 21–24
good morning message 8
maintenance
assigning a screen to a terminal 27
creating a screen 26
deleting a screen 26
editing online help 27
renaming a screen 27
news service 7–8
PIETLDIR help member 27
predefining sign on data 11
predefining sign on data for transparent operation 12
preparing for transparent mode 90
receiving a VTAM log on message 10
receiving sign on data from another application 8
screen commands 25–26
screen format 19
search method for sign on data 16
specifying screen with the Terminal Directory 48
use with automated log ons to target regions 113
user sign on retry limit 5
working with VTAM session managers 9

Logon Directory
sign on message 17
Logon Message Analysis exit 10, 140
Logon Message exit 116
Logon Panel
renaming 42, 51

M
MACLIB 197
macros
ALIASTBL 181
PG@RMALI 181
PG@RMATT 182–183
PG@RMCAN 184
PG@RMFLD 188–190, 193–195
PG@SMSUB 191
PIEMSG 175
PNDMOPT 124–126
MAIL transaction 183
Maintaining 25
MAXSTAT start-up option 126
MEMBER JCL statement 197
messages
controlling display 37
editing 175
locating source members 174
logging 173
password expiration 79
PIEL log 69
sending to additional destinations 175
setting profile option to display message IDs 37
short form 174
sign on message display time 69
MODEL entry
description 45
editing 47–50
MRO
specifying MRO/ISC support 67, 102–103
MSA 170, 187
MultiCICS
limiting transactions to one session 71
temporary storage queue name considerations 186

N
NATURAL 170
NATURAL 2 170
NATURAL/NATURAL 2 170
NetMizer
recommended optimization 158
specifying optimization with the Terminal Directory 49–50
use with local terminals 158
Network Monitor
ACBNAME 125
automatically starting during PIE/CICS initialization 69
Group Table 126
making changes to the systems table 122
monitoring over 200 regions 122
performance data recorded to the PIEL log 173
PNDMOPT macro 124–127
PNDMPEXT exit 142
specifying ddname of MVS program loads with the Customization Options utility 70
VTAM application status descriptions 126
news service 7–8
creating members with the Text utility 7
non-displayable TSQ swap substitution 189–190
NonStop CICS
changes to the Options file 123
noswap TSQ name substitution 186

O
object administrator
description 76
object owner
description 76
object security 75–78
OMEGAMON 170
OMNIGUARD
adding a TWA to PIE transactions 79
optimization
options for network 49
Options file
creating a separate file for each region 103
description 65
sharing between regions 102, 103
use with Network Monitor 123
OUTC JCL statement 197

P
P#IV transaction 164
P#NT transaction 154
P#RM transaction 119
P#TM transaction 86, 87
P@@@MTOK DSECT 147
PA key
assigning 39
PASS command examples 118
passwords
encrypted in the VTAM log on message 10
expiration messages 79
mask 55
reverification 83
specifying for automated sign ons 48
specifying preset passwords with the PCTSPDCX exit 46
paste operation 177
setting PF keys 38
specifying terminal beep 70
PC@STEXT 146
PC@UPVUD exit 132
PC@XPSEC exit 132
PCDR transaction 154
PCMRPCRU exit 133
PCSCPCAN exit 134
PCSMPPFIL exit 135
PCSPCAN exit 157
PCSSPAEE exit 136
PCSSPTOE exit 58, 137
PCSSSEXT exit 67, 138
PCSTPPVI exit 4, 79, 139
PCTSPDCX exit 10, 46, 140
PCTSPGM program 9
PCTTPUX1 148
performance
   DASD I/O 157
   exits 160
   log on message 159
PG@RMALI macro 181
PG@RMFLD macro 188– 190, 193– 195
PG@SMSUB macro 191
Phone and fax numbers, UNICOM Systems, Inc.
   Software 163
PIE objects
   access authorization 150
PIECVP TS queue 164
PIEL log 173
PIE message log 69
PIELMAIN screen 11, 48
PIENM001 ACBNAME 125
PIETLDIR help member 27, 68
PIETST program 155, 177
PIEUSER user ID 48
PNC@PLPE exit 142
PNDAPPLS 122
PNDMMPOPT macro statement 124
PNDMPEXT exit 142
PNDMTOPT macro 123
PNDMTOPT member 123
PO@MNUL exit 79
PO@XPSEC exit 79
pop-up menus
   preparing for transparent mode 90
printer
   queueing screen print requests to unavailable printers 71
   setting alternate with the User Directory 58
   setting default with the User Directory 57
PRINTID command 58
private object authority
   description 31
   specifying with User Directory 59
private objects
   authorization with the User Directory 59
private profile 30
PROFILE command 30
Profile utility
   description 32
   Global PIE Keys screen 38– 39
   invoking 34
   List screen 35
   main screen 35– 36
   Sessions Configuration screen 40
   Terminal Characteristics screen 37– 38
profiles
   assigning at user sign on 30
   attributes of the default user profile 30
   creating 33
   disabling global PIE keys 38
   editing 34– 40
   maintenance
      deleting profiles 42
      renaming profiles 42
      screen commands 41
   private 30
   private object authority 31
   run-time 30
   setting appearance of screen messages 37
   setting default application 35
   setting final PIE command 36
   setting initial PIE command 35– 36
   setting length of a screen scroll 38
   setting option to display titles of screen fields 37
   setting PIE escape string 37
   setting system END key 37
   setting user default 55
programs
  CSGM  8
  excluding from monitoring to run
  STABILIZE  170
  PCTSPGMM  9
  PIETST  155
  renaming with ALIASTBL system table
  181
Protected Terminal list
  resetting a terminal to unprotected
  mode  5,  66
pseudo terminals
  NATURAL  170
PSGM transaction  3,  8,  68,  154
PSMTAB  175
PSWO transaction  154
PUX1 exits
  assembling  145
  base registers  147
  commarea  146– 147
  PO@STEXT User Exit table  145– 146
  PUXCMD  148
  PUXINIT  149
  PUXMNUL  150
  PUXOBJS  150
  PUXVARS  151
  PUXVIEW  151
  PUXVSCN  152
  source members  148
  using a token list  147
  PUXCMD exit  71,  90,  148
  PUXCMD user exit module  146,  167
  PUXEXIT exit macro  145
  PUXINIT exit  4,  46,  85,  149
  PUXMNUL exit  150
  PUXOBJS exit  150
  PUXVARS exit  151
  PUXVIEW exit  151
  PUXVSCN exit  152
  PUXVVFLG field  152

Q
  Q1 JCL statement  197
  Q2 JCL statement  197

R
  RACF
    ICHR X02 post-processing exit  79
    password expiration message  79
  securing PIE transactions  80
  recommended reading  xii
  REFRESH command  21
  regions
    over 200  122
    status  125
  related reading  xiii
  Repository database
    limiting private object authority to re-duce DASD I/Os  157
    sharing between regions  103
  Repository database
    recording changes to the PIEL log  173
  resource access exits
    menu line  132
  RETRIEVE command  32
  reverifying passwords  83

S
  screens
    assigning a log on screen to a terminal  27
    creating a log on screen  26
    deleting a log on screen  26
    PIELMAIN  11
    print requests  71
    renaming a log on screen  27
    scroll default  38
    setting default  55
    setting field titles with Profile utility  37
    titles display  37
    TSWITCH return message  50
  SECMAC JCL statement  197
  security
    access rules  75
    authorizing user commands by class
      with the User Directory  59
    automatically starting the Terminal
      Security Monitor at PIE initialization
      69
    bypassing with PIEUSER feature  15
    changing an external security system
      82
    displaying protected menu lines to
      unauthorized users  70
    displaying sign on messages  58
    external security system exits  79
    LOCK command to secure an
      unprotected terminal  85
    MSA  170

205
object 75–78
OMNIGUARD 79
password re-verification 83
PC@XPSEC exit 132
protecting unguarded terminals 84–87
securing a terminal with the LOCK command 85
setting terminal time-out action with the PCSSPTOE exit 137
setting the maximum number of log on attempts 66
transaction security class 75–82
TSWITCH
setting time-out period 85
TSWITCH auto-signon time-out period 67
unauthorized attempts to capture a screen with the VIEW CAPTURE command 152
using RACF CHRIX02 exit to send password expiration messages 79
verification interval of Dynamic Menus 70
verifying log on information with the PCSTPPVI exit 139
SECURITY JCL statement 197
SECURITY parameter 15
sessions
anticipated average number of open sessions per user 67
assigning a name 40
assigning a title 40
authorizing maximum number with the User Directory 59
authorizing users to cancel through the User Directory 58–59
characteristics specified with the default user profile 30
default action on session close 72
default action taken if users attempt to create a new session when additional sessions are not available 71
disabling disconnected sessions for transparent mode 91
generic name matching 71
hidden 40
initial command 40
maximum number of log on attempts to restore a locked terminal 66
running multiple sessions in transparent mode 92
system clean-up operations with the PCSCPCAN exit 134
Sessions menu
preparing for transparent mode 90
SET command 37, 38
short log on message
auto signon terminals 50
specifying with the Terminal Directory 50
short log on messages
using to reduce network load 159
sign on
ACF2 appendage program 79
automating 6
customized for applications 9
Logon Director message 17
receiving data from another application 8
retry limit 5
search strategy used by the Logon Director 16
SIT
Logon Message Analysis exit 140
SKIPNEXT command 39
software compatibility
CEDF 169
DADS 169
ETC 170
EZTEST 170
FETCH 169
GENER/OL 169
IDEAL 170
INTERTEST 170
MSA 170
NATURAL 170
OMEGAMON 170
STABILIZE 170
STABILIZE 170
STAMSG start-up option 126
stand-alone exits
PC@#UPVUD 132
PCSCPCAN 134
worksheet 167
status of a target region 125
SuperMessage
authorization to receive 60
using the PG@RMATT macro to exclude CEDA from receiving messages 183
swap substitution of TSQ names 186
syntax conventions xiv
SYSID command 120
SYSIN JCL statement 197
system administrator
   authority to modify the User Directory  56
description  75
system end key
   specifying with the Profile utility  37
system end keys
   specifying with the Profile utility  33
   use in transparent mode  91
System Initialization Table
   LOGNMSG statement  10
   TSMGSET parameter  154
system tables
   ALIASTBL  181
   ATTACHX  92, 182– 183
   CANTABLE  184
description  179
   FIELDMAP  185
   modifying  180
   TERMAK  170, 185– 191
   TSQFLDS  169, 170, 186– 191
   USERFLDS  193– 194

UNICOM Systems, Inc., phone and fax numbers  163
temporary storage
   conserving by allocating a global key TSQ when a key has been defined.  33
tuning
   temporary storage  157
temporary storage owning region  155
temporary storage queue name substitution  67, 186
Temporary Storage Table  177
TERMAK system table
   substituting TSQ names  186– 191
TERMAK table  170
TERMAK terminal mask table  179
Terminal Control Table  113
Terminal Control Table User Area  193
Terminal Directory
   authorization to modify commands  44
   description  43
   editing  47– 50
   entries  45
   creating  51
   deleting  51
   renaming  51
   reorganizing  51
   search order  45
   fields
      Dial  49
      Logoff  48
      Logon  48
      Opt  49
      Password  48
      PIE  48
      RST  50
      SLM  50
      Terminal  47
      TRM  50
      Userid  47
   invoking  47
   overriding defaults with PUXINIT exit  46
   required customization  44
Terminal Input Output Area  183
terminal reset option  50
Terminal Security Monitor
   authorizing users  59
   automatically starting at PIE initialization  69
description  85
disconnecting or logging off users  85
excluding users and terminals from being disconnected  85
running  86– 87
starting  87
stopping  87
terminals
   action after invoking the Logon Director  48
   assigning a log on screen  27
   assigning Logon Director screens  27
dial-up ports  49
excluding from being disconnected by TSM  85
NetMizer optimization  49
protected mode  5
pseudo terminal IDs with MSA  170
pseudo terminal IDs with NATURAL or NATURAL 2  170
reset port if used with the IBM 7171 control unit  50
setting maximum log on retry number  66
setting VTAM logmode  114
supported types by PIE/CICS  47
Text utility
   creating news members  7
token delimiters
specifying with Customization Options utility 70
TOPSECRET
securing PIE transactions 80
transactions
CEDA 183
CEDF 184
CESF 15, 92, 183
CESN 15, 93
CRTE 120
CSSF 15
CSSN 15
DSA usage 154
EMON 170
identifiers ??– 82
MAIL 183
P#IV 164
P#NT 154
P#RM 119
P#TM 86, 87
PCDR 154
prevent cancellation with the
  CANTABLE system table 184
processing invalid transactions 71
PSGM 3, 8, 68, 154
PSWO 154
security classes ??– 82
security groups 80–??
specifying Dynamic Transaction Routing 70
substituting IDs with the ATTACHX
  system table 182–183
transient TCTTEs 68
transparent mode
description 89
disabling AutoLock 91
disabling disconnected sessions feature 91
logging off 92
running multiple sessions 92
setting global PIE keys 92
system end keys 91
TSQ name substitution 186–191
TSQFLDS system table
  modifying for use with GENER/OL 169
  modifying for use with IDEAL or ETC 170
substituting TSQ names 186–191
TSWITCH command
comparison with PASS and ACCESS
  commands 110–112
customizing to support cut and paste
  operations 177
DSA usage 154
examples of usage 118
extract sign on data from the VTAM log
  on message 71
required CICS and VTAM customization 113
return message 50
select disconnect status with the
  Customization Options utility 58
setting Auto-Signon time out period 67, 85
syntax 117
terminal characteristics set in the VTAM
  logmode table 114
use with a dial-up port 49
use with the TS access method 121
writing data to the PIE log with the
  PNC@PLPE exit 142
tuning
  DASD I/O 157
  datasets 157
  exits 160
  log on message 159
  NetMizer 158
TYPE start-up option 125

U
uppercase
changing default operation of the MAIL
  transaction 183
User Directory
  access limitations 53
  authorization to modify 44
description 43
editing 54–60
entries 45
fields
  Allow Disconnect Status 58
  Groupid 55
  Log Sign-on MSG 58
  Menu Name 55
  Password 55
  Profile 55
  Terminal 55
  Userid 54
maintenance 52
creating entries 60
deleting 60
renaming entries 60
reorganizing entries 61
required customization 44
scrolling 62
search order 45
user exits
  access authorization 150
USERDATA command 117, 132
USERFLDS system table 193–194
userid
  in the Terminal Directory 47, 48, 54
types 56–57
userid mask 54
UVSTAT start-up option 126

V
variables
  &INFO 19, 23
  &NEWS 19, 23
  &ZSYS 126
  &ZUDATA 132
assigning default user data to &ZUDATA 57
including in the news member 7
overriding substitution with the
  PUXVARS exit 151
ZLOGMOD 114
VIEW utility
  authorizing users 57
  controlling access with the PUXVSCN
    exit 152
VSAM
  tuning datasets 157
VTAM
  determining status of an applid with the
    PNDMPEXT exit 142
  extracting log on message by Logon
    Director 71
  format of log on messages 9
  good morning message 8
INQUIRE 125, 126
log on message 10
PNDMLOPT start-up option 125
reading contents of the log on message
  with the PCTSPDCX exit 140
return codes 126
SENDCMD 125
session managers 9
specifying terminal logmodes 114

W
WEQNENT start-up option 126
wildcards
  %
    use with Terminal and User directories 45
  *
    use with Terminal and User directories 45
  ?
    use with Terminal and User directories 45
    use with Terminal and User Directories 45
Work Element Queue 126
WORK J CL statement 197

Z
ZLOGMOD variable 114