Contents

About This Manual ................................................................. vii
   Audience .............................................................................. vii
   How This Manual is Organized ........................................... vii
   Recommended Reading ........................................................ viii
      Common Manuals ............................................................ viii
      Optional Manuals ............................................................ ix
      Syntax Conventions ......................................................... x

Chapter 1  Introduction ......................................................... 1
   Command Structure .......................................................... 1
      Variables ........................................................................ 1
   Session Identifiers ............................................................ 2
   Command Delimiters ........................................................ 3
   Wildcard Characters .......................................................... 4
   Escape String ..................................................................... 4
   PIE Exec Facility ............................................................... 5
   Transaction XCTL .............................................................. 6
   PIE Exec Command Execution .......................................... 7
      Command Access Authorization ........................................ 8
      Default PIE Exec Command Execution ........................... 8
   Command Queue Processing ............................................. 9
      FIFO Command Queue .................................................. 9
      RUN Command ............................................................ 9
      PROC Command .......................................................... 10

Chapter 2  Environment Commands .......................... 11
   ALIas ................................................................................. 14
   ALLocate ........................................................................... 15
   ALLOCATH ....................................................................... 16
   ALLOCATT ....................................................................... 18
   BULletin ............................................................................ 19
   BYe .................................................................................. 20
   CAncel .............................................................................. 22
   CANCELCl .......................................................................... 23
   CESF ................................................................................. 24
   CLose .............................................................................. 26
   CLOSEC .......................................................................... 27
   CREate ............................................................................. 28
   CSSF ................................................................................. 29
   CUT .................................................................................. 31
CUTEdit.................................................................33
DISC ........................................................................35
DISCHOLD .............................................................36
DISPLay .................................................................37
DROP .......................................................................38
END .........................................................................39
ESCAPE ..................................................................40
EXIT .........................................................................41
Free .........................................................................43
GOTO ......................................................................44
HELP .......................................................................46
Hide .........................................................................47
LOCK .......................................................................48
LOGOFF ...................................................................49
NAME .....................................................................51
NOTE .......................................................................52
NOTES .....................................................................53
OPEN .......................................................................54
PIE ...........................................................................55
PRINT ......................................................................56
PROFILE ...................................................................57
Query .......................................................................58
QUIT .......................................................................60
RESTART ..................................................................62
RETURN ...................................................................63
RETURNA ..................................................................64
SESSION ..................................................................65
SESSION ..................................................................66
SET .........................................................................67
SKIPnext ..................................................................70
START .....................................................................71
SWITCH ...................................................................72
SWITCHR ..................................................................75
TERM .......................................................................76
Title ........................................................................77
TOGGLE ....................................................................78
UNHIDE ....................................................................79
VIEW CAPTURE .....................................................80
VIEW .......................................................................82

Chapter 3  Application Commands .....................85

ACCESS (NetGate Only) ...........................................89
AutoEnd ...................................................................91
ATI .........................................................................92
ATTACH ...................................................................93
BROWSE ..................................................................94
BROWSEN ................................................................95
ConfirmEnd ...........................................................96
CICPRT ....................................................................97
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLRSt</td>
<td>98</td>
</tr>
<tr>
<td>CMD</td>
<td>99</td>
</tr>
<tr>
<td>Dsr</td>
<td>100</td>
</tr>
<tr>
<td>EC</td>
<td>101</td>
</tr>
<tr>
<td>EDIT</td>
<td>102</td>
</tr>
<tr>
<td>ENCP</td>
<td>103</td>
</tr>
<tr>
<td>ENV</td>
<td>104</td>
</tr>
<tr>
<td>GETC</td>
<td>105</td>
</tr>
<tr>
<td>GOTO</td>
<td>106</td>
</tr>
<tr>
<td>HELPDESK</td>
<td>107</td>
</tr>
<tr>
<td>KEY</td>
<td>108</td>
</tr>
<tr>
<td>LAUNCH (Dynamic Menus Only)</td>
<td>109</td>
</tr>
<tr>
<td>LIST</td>
<td>110</td>
</tr>
<tr>
<td>Load</td>
<td>112</td>
</tr>
<tr>
<td>LOGMode</td>
<td>113</td>
</tr>
<tr>
<td>MENU</td>
<td>114</td>
</tr>
<tr>
<td>MSG</td>
<td>115</td>
</tr>
<tr>
<td>NOAE</td>
<td>116</td>
</tr>
<tr>
<td>NoATI</td>
<td>117</td>
</tr>
<tr>
<td>NoATTach</td>
<td>118</td>
</tr>
<tr>
<td>NOCE</td>
<td>119</td>
</tr>
<tr>
<td>NOCICPRT</td>
<td>120</td>
</tr>
<tr>
<td>NOCLRRST</td>
<td>121</td>
</tr>
<tr>
<td>NODSR</td>
<td>122</td>
</tr>
<tr>
<td>NoENcP</td>
<td>123</td>
</tr>
<tr>
<td>NoMSG</td>
<td>124</td>
</tr>
<tr>
<td>NON3270</td>
<td>125</td>
</tr>
<tr>
<td>NoOPT (NetMizer Only)</td>
<td>126</td>
</tr>
<tr>
<td>NoPaSSP</td>
<td>127</td>
</tr>
<tr>
<td>NoRBS (NetMizer Only)</td>
<td>128</td>
</tr>
<tr>
<td>NOTSqsub (MultiCICS Only)</td>
<td>129</td>
</tr>
<tr>
<td>NOUCl</td>
<td>130</td>
</tr>
<tr>
<td>OPT (NetMizer Only)</td>
<td>131</td>
</tr>
<tr>
<td>Op1 (NetMizer Only)</td>
<td>132</td>
</tr>
<tr>
<td>OP2 (NetMizer Only)</td>
<td>133</td>
</tr>
<tr>
<td>OP3 (NetMizer Only)</td>
<td>134</td>
</tr>
<tr>
<td>PASS</td>
<td>135</td>
</tr>
<tr>
<td>PASSP</td>
<td>137</td>
</tr>
<tr>
<td>PERForm</td>
<td>138</td>
</tr>
<tr>
<td>PEXEC/EXEC</td>
<td>141</td>
</tr>
<tr>
<td>PRINTID</td>
<td>142</td>
</tr>
<tr>
<td>PROC</td>
<td>143</td>
</tr>
<tr>
<td>PROGRAM/PGM</td>
<td>144</td>
</tr>
<tr>
<td>PROMpt</td>
<td>145</td>
</tr>
<tr>
<td>PROMptN</td>
<td>146</td>
</tr>
<tr>
<td>RBS (NetMizer Only)</td>
<td>147</td>
</tr>
<tr>
<td>ReSeND (NetMizer Only)</td>
<td>148</td>
</tr>
<tr>
<td>RESETUSR</td>
<td>149</td>
</tr>
<tr>
<td>RETOFF</td>
<td>151</td>
</tr>
<tr>
<td>RETON</td>
<td>152</td>
</tr>
<tr>
<td>RETURN</td>
<td>153</td>
</tr>
</tbody>
</table>
About This Manual

This manual describes PIE/CICS commands available with Release 3.4.0. Each command is described separately. The description includes a syntax diagram, associated parameters, and examples of command usage.

Audience

This book is intended for system administrators and end-users of PIE/CICS. Readers are expected to understand CICS and MVS concepts.

How This Manual is Organized

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

- **Chapter 1  Introduction**
  Explains command elements and how PIE/CICS processes them to execute a command.

- **Chapter 2  Environment Commands**
  Lists commands to manage PIE/CICS sessions.

- **Chapter 3  Application Commands**
  Describes commands that set the operating conditions of PIE/CICS’s base and optional components.

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

- **Appendix B  Command Variables**
  Lists PIE/CICS command variables.

- **Appendix C  PIE/CICS Transactions**
  Lists PIE/CICS transactions.
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. These manuals need to be read only if these products 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 to PIE/CICS that are part of Release 3.4.0.
- PIE/CICS Command Reference
  Describes PIE/CICS Application and Environment commands.
- PIE/CICS Customization Guide
  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.
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.
- Monospace type indicates a screen field or an example of a PIE/CICS command entered on the screen.
Chapter 1 Introduction

This chapter explains some of the underlying concepts of PIE/CICS commands. It is an introductory chapter that gives an overview of PIE/CICS command elements and how they are interpreted as a command is executed.

PIE/CICS is a complex family of common and optional components. Together, these common and optional components provide an integrated environment that facilitates CICS transaction processing. Many commands are specific to optional components. Refer to individual product manuals within the PIE/CICS library for a more complete description of many of the commands described in this manual.

Command Structure

PIE/CICS commands are composed of constituent elements. Together, the command and these elements make up a command string. Although most commands do not use all elements, a PIE/CICS command string can be composed of the following:

- Variables
- Session identifiers
- Command delimiters
- Wildcard characters
- Escape strings

Each of these elements are described separately in the following sections.

Variables

Some PIE/CICS command parameters can be expressed as variables. A variable is a symbolic label that attributes the current value of data to a parameter.

Variables are used most often with dynamic data that changes frequently and is expressed in a consistent format. The date, time, user ID, or CICS job name are examples of consistent data types that can be expressed as variables. For example, the variable string &ZMONTHN &ZDAY, 19&ZYEAR generates the current date. Typical usage of PIE/CICS variables might be:

- Logon panels
- Dialog Manager menus
- Sessions menu
- Text files
- PIE Exec commands
- Session Manager command strings
A PIE/CICS variable is preceded with an ampersand (&) followed immediately by the variable name, which most often begins with Z.

Unresolved PIE/CICS variables may be passed to other PIE products by preceding the variable name with an additional ampersand (&). For example, to pass an unresolved &ZPSWD variable from a local PIE/CICS system to another PIE/CICS system, the password variable should be coded as &&ZPSWD. The local PIE/CICS system removes the first ampersand (&) and passes &ZPSWD to the second system to resolve the variable.

PIE/CICS variables may be concatenated with other items (including other PIE/CICS variables) by the following rules:

- A variable may be concatenated with other data by appending it as a suffix without intervening spaces.
  
  19&ZYEAR ==> 1996
  
  &ZMONTH&ZDAY ==> MAY21

- Variables may be concatenated by placing a period (.) between both variables.
  
  &ZDAY.PSWD ==> 21PSWD

The implied length of data expressed as a PIE/CICS variable may be changed by placing an integer between the leading ampersand (&) and the first letter of the variable name. The following rules apply to modifying the length of the data expressed as a PIE/CICS variable.

- Data is truncated to the right if it exceeds the modified length of the PIE/CICS variable.
  
  &ZMONTHN ==> Nov
  
  &2ZMONTHN ==> No

- Data is padded with blanks to the right if it less than the modified length of the PIE/CICS variable.
  
  MONTHN&ZDAY ==> Nov21
  
  &4ZMONTHN&ZDAY ==> Nov 21

Session Identifiers

Some commands must be entered with a parameter that identifies a PIE/CICS session. The parameter identifies a session by either a session number or name that appears on a PIE/CICS menu. In the following example of a Bulletin menu, the system news session can be selected by either session number 2 shown beneath the #field, or NEWS shown on the Name field.

#..Name...Title............................Arguments............
User=USR1 Terminal=PIE10005
Current Authorized View User=(NONE)

  1 HELPVIEW - Authorize Viewing of my screens by USER==>
  2 NEWS - System News on Tue 04/30/99 10:37:29

Session numbers range from 1 to 99 when PIE/CICS is used with MultiCICS. Otherwise, a single session is used and is identified as session 1.
Command Delimiters

Multiple commands can be entered as a single command string if they are separated by a command delimiter. After pressing ENTER, the commands are executed in the same order as they are entered in the string. A command delimiter is a single character placed between command strings.

A command delimiter is defined in a user's PIE profile. It is best to use a single delimiter for all profiles. The default command delimiter is a semi-colon (;).

The following example shows how a semi-colon command delimiter separates multiple commands in a string.

```
ae cmd cspk;ae prompte Screen Print Completed;sm return
```

A command delimiter is not required if a command does not have parameters. For example, AE does not require a delimiter placed after it because it has no parameters. On the other hand, CMD does have parameters and a delimiter must be placed between CMD and the next command in the string.

The following examples explain how command delimiters work with various types of PIE/CICS commands.

- **BROWSE PIEGNEWS;BROWSE PIELNEWS**
  
  This command string browses two news files consecutively; one for global news and the other for local news. When the user ends the first browse, the second BROWSE command automatically displays PIELNEWS.

  Stacked commands must be controlled by AutoEnd, either by default or with explicit AE declarations in the command string. In this example, BROWSE is a PIE command that autoends by default. An AE command is not required to end the first session before executing the second BROWSE command.

- **AE CMD CEDA;AE CMD CEMT**
  
  The CEDA and CEMT transactions are non-PIE commands. An AE command must be coded after the first command before executing the second transaction.

- **USERDATA &ARG;SCRIPT PLAY SYSTEM.APACCNT**
  
  If the first transaction is non-conversational, the next transaction is executed immediately. The user does not have to press a key to end the transaction. For example, the PIE command USERDATA is non-conversational. It accepts an argument from the user and places it in the &ZUDATA variable. The following execute string places the argument the user supplies in the &ZUDATA variable and the plays a script called SYSTEM.APACCNT, which requires that variable:
Wildcard Characters

PIE/CICS commands support wildcard characters as masks. Values are attributed to parameters based upon a match between data and the unmasked portion of the specified parameter.

PIE/CICS commands support the following wildcard characters:

%  Accept if a single number (0-9) occupies the same position as % in the data element.

?  Accept if a single non-numeric character occupies the same position as ? in the data element.

*  Accept all characters from the position occupied by * to the end of the data element.

Escape String

An escape string allows PIE/CICS commands to be entered from any menu. The command does not have to be entered from an application-specific menu to execute successfully. For example, Session menu commands can be entered from a NetMizer menu by using an escape string.

Typically, an escape string consists of a 1 to 5-character prefix placed immediately before a PIE/CICS command. The following list shows examples of commands with a variety of escape strings:

- ==SWITCH
- /EXIT LOGOFF
- ###PAYROLL$CEMT ITAS

The default escape string is ==.

PIE/CICS recognizes an escape string only if it is entered in the first position of a screen field. An escape string may be entered over the beginning of any field on a screen, followed by the command. The existing data within the field is not lost by over typing it with a command. Session Manager restores the data after executing the command.

The interpretation of the escape string and command is cursor sensitive. Any data left on the screen following the cursor is ignored. If a command is entered with an error and then corrected by back-spacing with the cursor, all data following the cursor is ignored. The cursor must be placed at the end of the command string before pressing ENTER.

PIE/CICS uses the following logic to process commands entered with an escape string:

- If a PIE environment command is entered with an escape string, the command is executed in the current session or in a special session, depending upon the requirements of the command.
- If a command is entered with a session ID and an escape string, MultiCICS switches to the session specified by the command parameter.
- If a PIE application command is entered with an escape string, the command is executed in a new session.
- If a valid CICS transaction is entered with an escape string, PIE/CICS opens a new session to execute the transaction.
The PIE Exec facility is a processor that provides the core functions of PIE/CICS command interpretation and execution. PIE/CICS commands either invoke or in some way manage the following functions:

- Transfer control to a transaction by emulating terminal input
- Start a new transaction
- Transfer control to a program
- Execute a FIFO queue containing command strings
- Invoke a menu
- Edit a text object
- Browse a text object
- Enter into native CICS environment (blank screen)
- Erase screen and display a message
- Pass a terminal to a remote VTAM application
- Switch a terminal to a remote VTAM application
- Request log off from PIE/CICS and CICS
- Request sign off from PIE/CICS

- Display the current status of specified aspects of a PIE system

PIE Exec also provides a set of commands that alter some conditions of a PIE/CICS session or the immediate transaction environment. They are called state commands because they set the state in which other commands or transactions are processed by the PIE Exec facility.

A PIE Exec command string contains one or more substrings. Each substring can contain one or more commands. Command substrings are delineated by command delimiters defined in user profiles. PIE Exec processes its command string one substring at a time and, for each substring, one command at a time. A PIE Exec command string can be of any length. The command string itself is free form, though certain commands have specific requirements on their parameters.

PIE Exec accepts commands from the following:

- Dialog Manager command strings
- Session manager commands

Any command that Session Manager does not recognize as a Sessions command is executed as a PIE Exec command in a new session.

You can only pass one PIE Exec command substring via Session manager. Any commands after the first command delimiter are ignored.

To execute a PIE Exec command in a new session, simply pass it to Session Manager in one of the following ways.

- (MultiCICS only) Enter the command from the Command line of the Sessions menu.
- MultiCICS only) In the first position of any input field, enter the escape string, followed by the PIE Exec command and press ENTER.
Chapter 1  Introduction

- PIE Exec commands can be predefined in user profiles by entering them on the Default Application, Global PIE keys, and Initial Sessions fields of the profile menus.
- Menu Command line
- Native CICS screen
  When there are no tasks running on a terminal in system mode, enter PEXE or PEXEC, followed by a space, and the PIE Exec command string. The PEXE/PEEXEC commands start PIE Exec as a transaction to execute a command.
- From a program
  Programs can directly invoke PIE Exec by an application program interface.
- Logon message data
  Commands passed as a data parameter of commands to open or allocate another session, can be processed by the PIE Exec PROC command.

Transaction XCTL

With the exception of state commands, the majority of PIE Exec commands result in new tasks being created and given control. PIE/CICS employs a unique task initiation method that does not use the CICS Interval Control START facility. This method is analogous to the CICS transfer of control (XCTL) service that transfers control to another program. PIE/CICS ‘task initiation method can be viewed as transfer of control at the transaction level, hence the name transaction XCTL.

Part of the interval control start of a terminal owning task involves bidding for the terminal facility. Overhead can be especially high in MRO environments where the terminal facility and the task reside on different CICS systems. The PIE/CICS transaction XCTL service avoids this overhead, significantly reducing task initiation delay. Another benefit of transaction XCTL is the elimination of the waiting period between the end of one task and ATI initiation of a subsequent task.
PIE Exec Command Execution

Before processing each substring, PIE Exec checks the terminal entry uppercase translate byte setting to determine if lower-to-uppercase translation is required. This setting can be changed with the UCT and NOUCT state commands. If uppercase translation is active, the entire substring is translated into uppercase and hexadecimal zeros are translated into blanks.

After uppercase translation, the substring is parsed into tokens. The first substring token is assigned as the command verb. Remaining tokens are assigned to command parameters. PIE Exec allocates the requisite number of tokens for commands with an invariant number of parameters. After that, the next token in the sequence is assigned to the following command verb.

A token list consists of a contiguous sequence of token descriptors. Each descriptor in the list describes the corresponding token in the sequence. A token descriptor is six bytes long and consists of the offset, length, and flags:

- **offset**: Byte length added to the address of the parameter string that gives the address of the token word.
- **Length**: Byte length of the token word.
- **flag byte**: Flags that identify the type of data passed in a parameter. Three one-bit flags indicate the following: numeric character present, alphabetic character (upper or lower case) present, and special character present.

An input command string may contain PIE/CICS variables. Variable substitution occurs late in command processing, at a stage called parameter modification. Variables cannot be used to convey PIE Exec command verbs.

After the command verb has been identified and parameters have been extracted from a substring, PIE Exec invokes the Variable Substitution Exit to modify parameters and determine the variable substitution option.

The Variable Substitution Exit may do any or none of the following:

- Construct a new parameter string
- Set variable substitution options

This exit cannot change command code nor reject the command.

The parameters for this exit include the command code, the parameter string, and a matching token list. The command code is a single character code that identifies the command verb. The length and address of the parameter string are passed with a token list and the count of tokens. The token list facilitates the analysis of the parameter string, and the construction of a new parameter string, should that be necessary.

This exit can create a new parameter string in the 256 character work area passed, then return the length and address of the new string. This exit should also set one of the three PIE variable substitution options:

- No substitution
- Substitute only unsecured variables (at present, password is the only secured variable)
- Substitute all variables
On return from this exit, PIE Exec checks for the presence of the new string length and address. If a new string has been returned, it will be used as the parameter string for that command. Unless the variable substitution option returned indicates no substitution, variable substitution service is invoked next. The resultant parameter string is then passed to the Command Authorization Exit. If authorized, execution of the command will proceed.

**Command Access Authorization**

The modified and substituted parameter string are passed to the Command Access Authorization exit. Again, a matching token list is passed. This call gives the opportunity to examine the command and its parameters again. The command string can be accepted or rejected in its current form.

Also, the command string can be accepted with further modifications. Through the Command Authorization Exit, applications and PIE/CICS functions can be restricted to specific groups of users.

If a command is rejected by the Command Access Authorization exit, the following message appears in a popup window at the bottom of screen.

`Command rejected by user exit
Press ENTER to continue`

**Default PIE Exec Command Execution**

PIE Exec assigns default commands in the following situations:

- If PIE Exec receives control via XCTL without a commarea, it assigns the RUN command by default.
- If PIE/CICS receives control otherwise, but without input (like ATI STARTed without data, or failed to retrieve data), it assigns the END command by default.
- When PIE Exec encounters a token which is not a valid PIE Exec command verb, it will first attempt to interpret the first four characters of that token as a transid by looking in the PCT. If found, then TRAN is assigned as the default command. If not found in the PCT, PIE Exec assigns MENU as the default command, leaving it up to the menu processor to decide whether the menu actually exists or not.
- If PIE Exec receives control with a command string that is blank, or contains only place-holder and/or state variable manipulation commands, then the default application in the user’s profile will be processed. If the default application is blank, then the SYSTEM command is processed.
- PIE Exec honors the MENU command by passing control to the menu processor, irrespective of whether a menu name has been provided or not. If no menu name is passed to the menu processor, the default menu name (from User Directory) for that user will be assumed.
Command Queue Processing

Menu and command facilities are two proven end-user productivity facilities. Another productivity facility enables applications to dynamically construct a sequence of commands and then conditionally execute them. These commands are placed into an execution queue.

These commands can be considered as functional extensions of scheduling programs. Typically, network or hierarchical application programs can schedule commands into this unique session queue. PIE Exec command scheduling is done with the PIE Exec Link Interface.

FIFO Command Queue

When PIE Exec is linked to a commarea, the contents of the commarea are written to the command queue (a CICS Temporary Storage queue) of that session. Any application running in the session can schedule commands into the queue with a link to PIE Exec.

Command strings in this queue are always processed in the order they are created, hence the name FIFO command queue. Each command is processed once. When the last command string is executed, the queue is immediately deleted.

RUN Command

The PIE Exec RUN command processes command strings in the FIFO command queue. RUN sequentially reads an item from the queue and invokes PIE Exec with the XCTL interface to process the command string. Before transferring control to PIE Exec, a check is made to see if the last command string in the queue has been reached. If there are more command strings, the RUN command token remains. This guarantees the existence of a residual command and forces the current level to be pushed onto the process stack. The command string is handled in the new level just as any PIE Exec command string. When all the commands in the string have been processed and control is passed back from the new level, the RUN command is executed again to read the next command string in the FIFO command queue. This process repeats until all commands strings stored in the queue are processed.

After processing the last command string in the FIFO command queue, the queue is deleted and the RUN command token is discarded. If this RUN command is not followed by other commands, a new invocation of PIE Exec is not required to run at a new level.

Thus, the RUN command executes all commands in the queue unless the session is prematurely terminated or PIE/CICS had a problem reading the queue. In both cases, the queue is deleted. If a RUN command is never issued, the queue is deleted at the end of the session.

Any additional RUN commands encountered while there is a RUN level in the process stack are ignored to prevent looping due to recursive processing of the command queue.
Chapter 1  Introduction

**PROC Command**

The ability to schedule commands into a command queue for subsequent processing provides a method to execute a series of command strings when a session is initially opened. PIE/CICS Logon Director has the capability to extract and save a command string passed in a VTAM log on message. That command string can be scheduled into the FIFO command queue with the PROC command.

The PROC command reads the command string from an interim command queue and then links to PIE Exec to schedule the execution of the command string. It then XCTLs to PIE Exec with the RUN command.

The PROC command can be processed once for each sign on to PIE. After all commands from the VTAM log on message have been scheduled into the FIFO command queue, PROC deletes the interim command queue. The PROC command can be specified as the initial command of a profile.
# Chapter 2  Environment Commands

This chapter describes Environment commands that are used to manage PIE/CICS sessions. The following table lists the commands, their parameters, and a page reference for more information.

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameters</th>
<th>Page Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALIas</td>
<td>[ssssssss] [name]</td>
<td>“ALIas” on page 14</td>
</tr>
<tr>
<td>ALLocate</td>
<td>[ssssssss] [title]</td>
<td>“ALLocate” on page 15</td>
</tr>
<tr>
<td>ALLOCATH</td>
<td>[ssssssss] [title]</td>
<td>“ALLOCATH” on page 16</td>
</tr>
<tr>
<td>ALLOCATT</td>
<td>[ssssssss] [title]</td>
<td>“ALLOCATT” on page 18</td>
</tr>
<tr>
<td>BULletin</td>
<td>[ssssssss]</td>
<td>“BULletin” on page 19</td>
</tr>
<tr>
<td>BYe</td>
<td>\DISC|\DISCHOLD|\LOGOFF|\SIGNOFF|\LOCK|\ENDPIE]</td>
<td>“BYe” on page 20</td>
</tr>
<tr>
<td>CAncel</td>
<td>[ssssssss] [ALL]</td>
<td>“CAncel” on page 22</td>
</tr>
<tr>
<td>CANCELC</td>
<td>[ssssssss] [ALL]</td>
<td>“CANCELC” on page 23</td>
</tr>
<tr>
<td>CESF</td>
<td>\DISC|\DISCHOLD|\LOGOFF|\SIGNOFF|\LOCK|\ENDPIE]</td>
<td>“CESF” on page 29</td>
</tr>
<tr>
<td>CLoae</td>
<td>[ssssssss] [ALL]</td>
<td>“CLoae” on page 26</td>
</tr>
<tr>
<td>CLOSEC</td>
<td>[ssssssss]</td>
<td>“CLOSEC” on page 27</td>
</tr>
<tr>
<td>CReate</td>
<td>[ssssssss] [title]</td>
<td>“CReate” on page 28</td>
</tr>
<tr>
<td>CSSF</td>
<td>\DISC|\DISCHOLD|\LOGOFF|\SIGNOFF|\LOCK|\ENDPIE]</td>
<td>“CSSF” on page 29</td>
</tr>
<tr>
<td>Cut</td>
<td>[FROM=screen_position] [,TO=screen_position] [,SESSION=*|\nn|\ssession] [,|FIFO|\LIFO] [,QUEUE=|CICS_TSQ|\ZUDATA|\ZUDATA2]</td>
<td>“Cut” on page 31</td>
</tr>
<tr>
<td>CUTEdit</td>
<td>[FROM=screen_position] [,TO=screen_position] [,SESSION=*|\nn|\ssession] [,|FIFO|\LIFO] [,QUEUE=|CICS_TSQ|\ZUDATA|\ZUDATA2]</td>
<td>“CUTEdit” on page 33</td>
</tr>
<tr>
<td>DISC</td>
<td>None</td>
<td>“DISC” on page 35</td>
</tr>
<tr>
<td>DISCHOLD</td>
<td>None</td>
<td>“DISCHOLD” on page 36</td>
</tr>
<tr>
<td>DISPlay</td>
<td>[ssssssss]</td>
<td>“DISPlay” on page 37</td>
</tr>
<tr>
<td>DRop</td>
<td>[ssssssss] [ALL]</td>
<td>“DRop” on page 38</td>
</tr>
<tr>
<td>ENd</td>
<td>None</td>
<td>“ENd” on page 39</td>
</tr>
<tr>
<td>ESCape</td>
<td>pie_exec_command</td>
<td>“ESCapect” on page 40</td>
</tr>
<tr>
<td>EXit</td>
<td>\DISC|\DISCHOLD|\LOGOFF|\SIGNOFF|\LOCK|\ENDPIE]</td>
<td>“EXit” on page 41</td>
</tr>
</tbody>
</table>
## Chapter 2  Environment Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameters</th>
<th>Page Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Free</td>
<td>{ssssssss}[ALL]</td>
<td>“Free” on page 43</td>
</tr>
<tr>
<td>Goto</td>
<td>{ssssssss}[execution_text]</td>
<td>“Goto” on page 44</td>
</tr>
<tr>
<td>Help</td>
<td>[command_name]</td>
<td>“Help” on page 46</td>
</tr>
<tr>
<td>Hlde</td>
<td>{ssssssss}[ALL]</td>
<td>“Hlde” on page 47</td>
</tr>
<tr>
<td>LOCK</td>
<td>{ssssssss}[ALL]</td>
<td>“LOCK” on page 48</td>
</tr>
<tr>
<td>LOGOff</td>
<td>[DISC</td>
<td>DISCHOLD</td>
</tr>
<tr>
<td>Name</td>
<td>sssssss {name}</td>
<td>“Name” on page 51</td>
</tr>
<tr>
<td>NOTEc</td>
<td>sssssss {title}</td>
<td>“NOTEc” on page 52</td>
</tr>
<tr>
<td>NOTes</td>
<td>sssssss {title}</td>
<td>“NOTes” on page 53</td>
</tr>
<tr>
<td>OPen</td>
<td>sssssss {execution_text}</td>
<td>“OPen” on page 54</td>
</tr>
<tr>
<td>PASTE</td>
<td>{FROM=screen_position}</td>
<td>“PASTE” on page 55</td>
</tr>
<tr>
<td>PIE</td>
<td>{cccccccc}</td>
<td>“PIE” on page 56</td>
</tr>
<tr>
<td>PRINT</td>
<td>None</td>
<td>“PRINT” on page 57</td>
</tr>
<tr>
<td>PROfile</td>
<td>[terminal</td>
<td>1]</td>
</tr>
<tr>
<td>Query</td>
<td>sssssss</td>
<td>“Query” on page 59</td>
</tr>
<tr>
<td>QUIT</td>
<td>[DISC</td>
<td>DISCHOLD</td>
</tr>
<tr>
<td>RESTART</td>
<td>[pie_exec command]</td>
<td>“RESTART” on page 62</td>
</tr>
<tr>
<td>RETURN</td>
<td>None</td>
<td>“RETURN” on page 63</td>
</tr>
<tr>
<td>RETURNS</td>
<td>None</td>
<td>“RETURNS” on page 64</td>
</tr>
<tr>
<td>SESsion</td>
<td>None</td>
<td>“SESSion” on page 65</td>
</tr>
<tr>
<td>SESMenu</td>
<td>None</td>
<td>“SESSMenu” on page 66</td>
</tr>
<tr>
<td>SET</td>
<td>{SWSTR xxxx}</td>
<td>“SET” on page 67</td>
</tr>
<tr>
<td></td>
<td>{ESCTR xxxx}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>{Pfno fffff}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>{Pfno aaaaa}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>{ENTER}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>{ENDKEY Pfno}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>{RETKEY Pfno}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>{DELIM dddd}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>{TITLE [ON</td>
<td>OFF]}</td>
</tr>
<tr>
<td></td>
<td>{SMSG [ON</td>
<td>OFF]}</td>
</tr>
<tr>
<td></td>
<td>{KEYS [ON</td>
<td>OFF] [SON</td>
</tr>
<tr>
<td></td>
<td>{SCROLL [PAGE</td>
<td>HALF</td>
</tr>
<tr>
<td></td>
<td>{CMDTRACE [ON</td>
<td>OFF]}</td>
</tr>
<tr>
<td></td>
<td>{SBOND</td>
<td>SBOFF</td>
</tr>
<tr>
<td>SKIPnext</td>
<td>None</td>
<td>“SKIPnext” on page 70</td>
</tr>
<tr>
<td>STart</td>
<td>sssssss {execution_text}</td>
<td>“STart” on page 71</td>
</tr>
<tr>
<td>Command</td>
<td>Parameters</td>
<td>Page Reference</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>SWITCH</td>
<td>[ssssssss] [application]</td>
<td>“SWItch” on page 72</td>
</tr>
<tr>
<td></td>
<td>(NEXT</td>
<td>PREV) [OPENED</td>
</tr>
<tr>
<td>SWITCHR</td>
<td>ssssssss rexx_script rexx_args</td>
<td>“SWITCHR” on page 75</td>
</tr>
<tr>
<td>TERM</td>
<td>[ssssss] [ALL]</td>
<td>“TErm” on page 76</td>
</tr>
<tr>
<td>TITLE</td>
<td>sssssss [title]</td>
<td>“Title” on page 77</td>
</tr>
<tr>
<td>TOGGLE</td>
<td>[PA01</td>
<td>key]</td>
</tr>
<tr>
<td>UNHIDE</td>
<td>[ssssss] [ALL]</td>
<td>“Unhide” on page 79</td>
</tr>
<tr>
<td>VIEW CAPTURE</td>
<td>[userId] [terminal] [sessionId]</td>
<td>“VIEW CAPTURE” on page 80</td>
</tr>
<tr>
<td>VIEW</td>
<td>[FROM=screen_position]</td>
<td>“VIEW” on page 82</td>
</tr>
<tr>
<td></td>
<td>{CUT</td>
<td>CUTEDIT} [SESSION=[*</td>
</tr>
<tr>
<td></td>
<td>PASTE {LIFO</td>
<td>FIFO}</td>
</tr>
<tr>
<td></td>
<td>{QUEUE=[CICS_TSO</td>
<td>ZUDATA</td>
</tr>
</tbody>
</table>
**ALIas**

ALIas assigns a temporary name to a PIE/CICS session. The Name field of the Session menu displays the temporary name of the session assigned by the ALIas command. The name is retained for the period the session is active. The permanent name of the session is restored from the user’s profile when the session is reopened.

ALIas is a synonym of the NAme command. See “NAme on page 51.”

**Format**

```
ALIas [ssssssss] [name]
```

**Parameters**

- `ssssssss`  
  1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99, or a session name.
  
  If a session identifier is not specified, a new session is allocated and given the lowest available session number. The new session is assigned the default application defined in the user’s current profile.

- `name`  
  1 to 8- alphanumeric character name of the specified PIE/CICS session.
  
  If the name parameter is omitted from the ALIas command, the resulting Name field of the Sessions menu is blank.

**Examples**

- `ALIAS 5 DEVLPMNT`
  
  Session 5 is renamed DEVLPMNT.

- `ALIAS DEVLPMNT DEVELOP`
  
  The PIE/CICS session named DEVLPMNT is renamed to DEVELOP.
**ALLEcote**

ALLEcote creates a new session or assigns another application to an existing session. The session appears as a selectable option of the Sessions menu after being allocated.

If a new session is allocated:

- The session is allocated, designated as active, but not opened.
- The Name field remains blank on the Sessions menu.
- The Title field of the Sessions menu displays the application assigned to the allocated session.
- The session executes the command string assigned by the application parameter when it is opened.

If the allocated session currently exists:

- The session is allocated, designated as active, but not opened.
- The # and Name fields of the Sessions menu remain unchanged.
- The command string assigned to the session in the PIE Profile Sessions Configuration menu is replaced by the application parameter.

ALLEcote is an alias of the CREATE command. See “CREATE” on page 28.

**Format**

```
ALLEcote [nn] [application]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nn</td>
<td>Session number in a range from 1 to 99. If a session number is not specified, the session is assigned the lowest available session number.</td>
</tr>
<tr>
<td>application</td>
<td>Application assigned to the session or the command string that executes when the session is initially opened. Up to 66 characters in length. The Title field of the PIE/CICS Sessions menu shows the text of the application assigned to the allocated session. If the application parameter is not specified, the allocated session is assigned the default application defined in the user’s current profile.</td>
</tr>
</tbody>
</table>

**Examples**

- **ALLEcote 5 cemt i tas**
  
  Session 5 is allocated and the CEMT I TAS transaction executes when the session is opened.

- **ALL**
  
  A session is allocated and assigned the next available session number.
Chapter 2  Environment Commands

ALLOCATH

ALLOCATH creates a hidden PIE/CICS session that is excluded from the list of active sessions. The SWITCH or GOto commands bypass hidden sessions when they toggle through the user's active sessions. An H appears to the right of the Title field of the Sessions menu to indicate a session's hidden status. A specific reference must be made to a hidden session by either its number or name to switch to it.

If a new hidden session is allocated:

- The session is allocated, hidden, designated as active, but not opened.
- The Name field remains blank on the Sessions menu.
- The Title field of the Sessions menu displays the application assigned to the hidden session by the application parameter.
- The execute field of this session in the user's profile is assigned the command string specified by the application parameter.

If the hidden session is currently allocated:

- The session is hidden, designated as active, but not opened.
- The #, Name, and Title fields of the Sessions menu remain unchanged.
- The command string assigned to the session in the PIE Profile Sessions Configuration menu is replaced by the application parameter.

\[ ALLOCATH [ssssssss] [application] \]

**Parameters**

- **ssssssss** 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name. If a session number is not specified, the hidden session is assigned the lowest available session number.
- **application** Application assigned to the session or the command string that executes when the session is initially opened. Up to 66 characters in length. The Title field of the PIE/CICS Sessions menu shows the application assigned to the allocated session. If the application parameter is not specified, the hidden session is assigned the default application defined in the user's current profile.
Examples

- ALLOCATH 5 Test System
  
  Session 5 is allocated as a hidden session with 'Test System' as the description of the hidden session that appears in Title field of the Sessions menu. The title appears also in the Execute field of the user's session profile.

- ALLOCATH PROFILE menu piemsys
  
  The current PROFILE session becomes a hidden session that opens the PIE/CICS System Administrator menu when it is initially opened. The current session number and title of the PROFILE session remain unchanged.

- ALLOCATH
  
  A hidden session is allocated and assigned the lowest available session number. The Name and Title fields of the Sessions menu are blank. The session is assigned the default application defined in the user's current profile.
Chapter 2  Environment Commands

ALLOCATT

ALLOCATT creates a temporary PIE/CICS session that exists for the duration the application assigned to it is active and open. The temporary session is listed on the Sessions menu after it is allocated. The session is cancelled and no longer appears on the Sessions menu after exiting from the application.

After creating a temporary session:

- The session is allocated, designated as active, but not opened.
- The Name field remains blank on the Sessions menu.
- The Title field of the Sessions menu displays the application assigned to the temporary session by the application parameter.
- The execute field of this session in the user’s profile is assigned the command string specified by the application parameter.

Format

ALLOCATT [nn] [application]

Parameters

nn Session number in a range from 1 to 99.
If a session number is not specified, the session is assigned the lowest available session number.

application Application assigned to the session or the command string that executes when the session is initially opened. Up to 66 characters in length. The Title field of the PIE/CICS Sessions menu shows the text of the application assigned to the allocated session.
If the application parameter is not specified, the temporary session is assigned the default application defined in the user’s profile.

Example

- ALLOCATT 5 access applid=cicstest
  A temporary session is allocated that accesses a test CICS region when it is opened. The session is cancelled after exiting from the CICS region.
- ALLOCATT
  A temporary session is allocated and assigned the lowest available session number. The default application defined in the user’s profile is specified as the application for the temporary session.
BULletin

BULletin browses an electronic bulletin board assigned to a PIE/CICS user group. Normally, a bulletin board is a primary menu listing a variety of topics that can be selected and read by the user. BULletin provides parameters to select specific topics from the bulletin board menu.

Format

BULletin [ssssssss|n]

Parameters

ssssssss 1 to 8- character alphanumeric name of an option displayed on a bulletin board primary menu.
n Number of an option listed on a bulletin board menu.

Examples

• BULLETIN
  The designated bulletin board for a user group is displayed.
• BU NEWS
  The NEWS option is displayed when the bulletin board is opened.
• BU 5
  Option 5 of the bulletin board assigned to a PIE/CICS user group is displayed.
BYe

BYe provides several options to exit from PIE/CICS or disconnect sessions. A command parameter specifies the specific method to exit from PIE/CICS or stop session work.

If BYe is entered without a parameter, a pop-up window appears listing BYe parameters. Each parameter has an associated PF key. Pressing the key is equivalent to entering BYe and the parameter together.

BYe is an alias of the following PIE/CICS commands:

- CESF
- CSSF
- EXit
- LOGOFF
- QUIT

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

BYe [DISC|DISCHOLD|LOGOFF|SIGNOFF|END PIE|LOCK]

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISC</td>
<td>The user is disconnected from PIE/CICS and signed off from external security. The terminal is released from CICS and can be used for other applications. DISC temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time BYe was entered. Work can resume within these sessions at a later time by logging on to PIE/CICS again.</td>
</tr>
<tr>
<td>DISCHOLD</td>
<td>The user is disconnected from PIE/CICS and signed off from external security. The terminal returns to the Logon Director screen and remains attached to CICS. DISCHOLD temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time BYe was entered. Work can resume within these sessions at a later time by signing on to PIE/CICS again.</td>
</tr>
<tr>
<td>LOGOFF</td>
<td>All current user sessions are cancelled and the user is logged off PIE/CICS. Also, the user is signed off from external security. The terminal is released from CICS and can be used for other applications.</td>
</tr>
<tr>
<td>SIGNOFF</td>
<td>All current user sessions are cancelled and the user is signed off PIE/CICS and external security. The terminal returns to the Logon Director screen and remains attached to CICS.</td>
</tr>
<tr>
<td>END PIE</td>
<td>All current user sessions are cancelled and the user is logged off PIE/CICS. The terminal displays a blank screen and remains attached to CICS.</td>
</tr>
</tbody>
</table>
LOCK The terminal is disabled to prevent further keyboard input. The terminal is retained by CICS and sessions remain in their current state at the time the BYe command was issued. The terminal remains locked until the current user’s password is re-entered.

Examples

- **BYE**
  
  A pop-up window appears with a list of exit options that are selected with PF keys.

- **BYE DISC**
  
  All sessions are disconnected and the user is logged off PIE/CICS. The terminal is released from CICS and can be used for other applications.
CACancel

CACancel closes and deallocates a session. All system and application resources used by PIE/CICS sessions are freed and available for other work. The session is removed from the Sessions menu.

Active PIE/CICS transactions are cancelled when a session is closed by the CACancel command. Also, any unsaved work is lost that was being performed within the session when it closed.

If the ALL parameter is specified with CACancel, all allocated sessions for the current user are cancelled. All PIE/CICS resources are freed from the cancelled session(s).

If a session is designated as non-cancellable, a pop-up window appears in the session with a message indicating the session must be resumed and ended normally. The session remains inactive until it is resumed or ended normally.

CACancel is an alias of the DRop or FREE commands. See “DRop” on page 38 or “Free” on page 43.

Format

CACancel [ssssssss|ALL]

Parameters

ssssssss 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.

If a session identifier is not specified with CACancel, a pop-up window appears that lists all sessions. A PF key is assigned to each session. Pressing the associated key closes and deallocates the session.

ALL All allocated sessions are cancelled for the current user.

Examples

- CANCEL 5
  
  Session 5 is cancelled.

- CA ALL
  
  All PIE/CICS sessions allocated to the current user are cancelled.
CANCELC

CANCELC conditionally cancels a session. If the session exists, CANCELC closes and deallocates a session. All system and application resources used by PIE/CICS sessions are freed and available for other work. The session is removed from the Sessions menu.

If the session does not exist, CANCELC does nothing and RETURNS to the previous menu without issuing a message. On the other hand, the CANCEL command sends a message when an invalid session was specified with the command.

Format

```
CANCELC [ssssssss|ALL]
```

Parameters

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.
  
  If a session identifier is not specified with CANCEL, a pop-up window appears that lists all sessions. A PF key is assigned to each session. Pressing the associated key closes and deallocates the session.

- **ALL**: All allocated sessions are cancelled for the current user.

Examples

- **CANCELC 5**
  
  Session 5 is cancelled and deallocated if it exists. If it does not exist, a RETURN to the previous menu is made without issuing an error message.

- **CANCELC ALL**
  
  All PIE/CICS sessions allocated to the current user are cancelled.
Chapter 2  Environment Commands

CESF

CESF provides several options to exit from PIE/CICS or disconnect sessions. A command parameter specifies the specific method to exit from PIE/CICS or stop session work.

If CESF is entered without a parameter, a pop-up window appears listing the parameters. Each parameter has an associated PF key. Pressing the key is equivalent to entering CESF and the parameter together.

CESF is an alias of the following PIE/CICS commands:

- BYe
- CSSF
- Exit
- LOGOFF
- QUIT

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

| CESF [DISC|DISCHOLD|LOGOFF|SIGNOFF|END PIE|LOCK] |

Parameters

- DISC  The user is disconnected from PIE/CICS and signed off from external security. The terminal is released from CICS and can be used for other applications.
  
  DISC temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the CESF command was entered. Work can resume within these sessions at a later time by logging on to PIE/CICS again.

- DISCHOLD  The user is disconnected from PIE/CICS and signed off from external security. The terminal returns to the Logon Director screen and remains attached to CICS.
  
  DISCHOLD temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the BYe command was entered. Work can resume within these sessions at a later time by signing on to PIE/CICS again.

- LOGOFF  All current user sessions are cancelled and the user is logged off PIE/CICS. Also, the user is signed off from external security. The terminal is released from CICS and can be used for other applications.

- SIGNOFF  All current user sessions are cancelled and the user is signed off PIE/CICS and external security. The terminal returns to the Logon Director screen and remains attached to CICS.

- END PIE  All current user sessions are cancelled and the user is logged off PIE/CICS. The terminal displays a blank screen and remains attached to CICS.
LOCK The terminal is disabled to prevent further keyboard input. The terminal is retained by CICS and sessions remain in their current state at the time the CESF command was issued. The terminal remains locked until the current user’s password is re-entered.

Examples

- CESF
  A pop-up window appears with a list of exit options that are selected with PF keys.
- CESF dischold
  All active sessions are disconnected and the user is signed off PIE/CICS. The terminal returns to the Logon Director screen and remains attached to CICS.
Chapter 2  Environment Commands

**CLOSE**

CLOSE ends work occurring within a specified PIE/CICS session. All system and application resources used by a session are freed and available for other work. Active transactions are cancelled when the session is closed.

Permanently allocated sessions remain on the PIE/CICS Sessions menu after they are closed. A closed permanent session can be reopened to re-initiate work.

Temporary sessions are cancelled and removed from the Sessions menu. They must be reallocated after they are closed.

If the **ALL** parameter is specified with **CLOSE**, all allocated sessions for the current user are closed. All PIE/CICS resources are freed from the closed session(s).

If a session is designated as non-cancellable, a pop-up window appears in the session with a message indicating the session must be resumed and ended normally. The session remains inactive until it is resumed or ended normally.

**Format**

```
CLOSE [ssssssss][ALL]
```

**Parameters**

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.
- **ALL**: All PIE/CICS sessions are closed for the current user.

**Examples**

- **CLOSE 5**
  
  Session 5 is closed. The work occurring within the session is cancelled.

- **CLOSE all**
  
  All PIE/CICS sessions of the current user are cancelled. Temporary sessions are cancelled and removed from the user’s Sessions menu. Permanent sessions remain on the Sessions menu and can be reopened.
CLOSEC

CLOSEC conditionally ends work occurring within a specified PIE/CICS session. If a session exists, all system and application resources used by the session are freed and available for other work. Active transactions are cancelled when the session is closed.

Permanently allocated sessions remain on the PIE/CICS Sessions menu after they are closed. A closed permanent session can be reopened to re-initiate work.

Temporary sessions are cancelled and removed from the Sessions menu. They must be reallocated after they are closed.

If the ALL parameter is specified with CLOSE, all allocated sessions for the current user are closed. All PIE/CICS resources are freed from the closed session(s).

If a session is designated as non-cancellable, a pop-up window appears in the session with a message indicating the session must be resumed and ended normally. The session remains inactive until it is resumed or ended normally.

Format

```
CLOSEC [ssssssss|ALL]
```

Parameters

- `ssssssss` 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.
- `ALL` All PIE/CICS sessions are closed for the current user.

Examples

- **CLOSEC 5**
  
  Session 5 is closed. The work occurring within the session is cancelled.

- **Close all**
  
  All PIE/CICS sessions of the current user are cancelled. Temporary sessions are cancelled and removed from the user’s Sessions menu. Permanent sessions remain on the Sessions menu and can be reopened.
Chapter 2  Environment Commands

CReate

CReate allocates a PIE/CICS user session. After being allocated, the session appears as a selectable option on the PIE/CICS Sessions menu.

If a new session is allocated:

- The session is allocated, designated as active, but not opened.
- The Name field remains blank on the Sessions menu.
- The Title field of the Sessions menu displays the application assigned to the allocated session.
- The session executes the command string assigned by the application parameter when it is opened.

If the allocated session currently exists:

- The session is allocated, designated as active, but not opened.
- The # and Name fields of the Sessions menu remain unchanged.
- The execute field is replaced by the command string assigned by the application parameter.
- If the Title field of the Sessions menu is blank, it is replaced with the text specified with the application parameter.

CREATE is an alias of the ALLocate command. See “ALLocate” on page 15

Format

CReate [nn] [application]

Parameters

 nn  Session number in a range from 1 to 99.
     If a session number is not specified, the session is assigned the lowest available session number.

 application  Application assigned to the session or the command that executes when the session is initially opened. Up to 66 characters in length. The Title field of the PIE/CICS Sessions menu shows the application assigned to the allocated session.
     If the application parameter is not specified, the allocated session is assigned the default application defined in the user’s profile.

Examples

- CREATE 5 cemt i tas
  Session 5 is allocated. The CEMT transaction executes when session 5 opens.
- CREATE PAYROLL ACCESS APPLID=CICSPRD1
  A session named PAYROLL is allocated that accesses the CICSPRD1 region. The PAYROLL session is assigned the lowest available session number.
CSSF

CSSF provides several options to exit from PIE/CICS or disconnect sessions. A command parameter specifies the method to exit from PIE/CICS or stop session work.

If CSSF is entered without a parameter, a pop-up window appears listing the parameters. Each parameter has an associated PF key. Pressing the key is equivalent to entering CSSF and the parameter together.

CSSF is an alias of the following PIE/CICS commands:

- BYe
- CESF
- EXIt
- LOGOFF
- QUIT

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

```
CSSF [DISC|DISCHOLD|LOGOFF|SIGNOFF|END PIE|LOCK]
```

Parameters

- **DISC**
  
  The user is disconnected from PIE/CICS and signed off from external security. The terminal is released from CICS and can be used for other applications.

  DISC temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the CSSF command was entered. Work can resume within these sessions at a later time by logging on to PIE/CICS again.

- **DISCHOLD**
  
  The user is disconnected from PIE/CICS and signed off from external security. The terminal returns to the Logon Director screen and remains attached to CICS.

  DISCHOLD temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the CSSF command was entered. Work can resume within these sessions at a later time by signing on to PIE/CICS again.

- **LOGOFF**
  
  All current user sessions are cancelled and the user is logged off PIE/CICS. Also, the user is signed off from external security. The terminal is released from CICS and can be used for other applications.

- **SIGNOFF**
  
  All current user sessions are cancelled and the user is signed off PIE/CICS and external security. The terminal returns to the Logon Director screen and remains attached to CICS.

- **END PIE**
  
  All current user sessions are cancelled and the user is logged off PIE/CICS. The terminal displays a blank screen and remains attached to CICS.
LOCK

The terminal is disabled to prevent further keyboard input. The terminal is retained by CICS and sessions remain in their current state at the time the CSSF command was issued. The terminal remains locked until the current user’s password is re-entered.

Examples

- CSSF

  A pop-up window appears with a list of exit options that are selected with PF keys.

- CSSF_LOGOFF

  All current user sessions are cancelled and the user is logged off PIE/CICS.
CUT

CUT copies a selected portion of a screen and stores it to the PIE clipboard or a designated storage area.

Format

```
{FROM=screen_position}
{TO=screen_position}
CUT
{SESSION=[*|nn|session_name]}
{LIFO|FIFO}
{QUEUE=[CICS_TSQ|ZUDATA|ZUDATA*n]}
```

Parameters

**FROM**  
Starting position of the screen area for a cut operation. The default is CURSOR. The starting screen-position of a cut or paste can be specified in the following ways:
- CURSOR: Current cursor position.
- (row,col): Row and column number of a 24 row by 80 column 3270 screen. For example, specifying (01,01) is the top left corner of a 3270 screen.
- position: Cursor position in the 3270 buffer. 0 would be the first position on the screen.
- BEGFIELD: Beginning of the current field that the cursor is located.
- ENDFIELD: End of the current field that the cursor is located.
- BEGLINE: Beginning of the current line that the cursor is located.
- ENDLINE: End of the current line that the cursor is located.

**TO**  
Ending position of the screen area for a cut operation. The default is ENDFIELD.

The ending screen position is specified with the same variables as the FROM parameter.

**SESSION**  
Session containing the screen image to be cut. The current session (*) is the default. The session can be identified by a two-digit session number nn, or a 1 to 8-alphanumeric character session name.

**LIFO FIFO**  
Organization of screen areas cut and stored in the clipboard. Screens may be stored in last in, first out (LIFO) or first in, first out (FIFO) order. This option is used only when the clipboard is empty. The default value is LIFO.
### QUEUE
Temporary storage area to store data copied from a screen with the CUT command.

The temporary storage area can be one of the following:

<table>
<thead>
<tr>
<th>QUEUE</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS_TSQ</td>
<td>ID of a CICS temporary storage queue. The default. It is an internally generated PIE queue name that is unique for each user. This queue can be edited using object name of PIE.CLIPBRD.</td>
</tr>
<tr>
<td>ZUDATA</td>
<td>PIE user data area. If this option is used, the length of the cut data is limited to 8 characters, and only one item can be cut at a time.</td>
</tr>
<tr>
<td>ZUDATA2 through ZUDATA5</td>
<td>PIE user data areas ZUDATA2 through ZUDATA5. If this option is used, the length of the copied screen data is limited to 16 characters. Only one item can be cut at a time. If the ZUDATA or ZUDATA_n parameters are specified, copied data can be identified by the &amp;ZUDATA and &amp;ZUDATA2-5 PIE variables. These variables can be used in a command string as part of a PIE/CICS script.</td>
</tr>
</tbody>
</table>

### Examples

- **CUT fifo,from=(01,01),to=endline**
  
  Data is copied from the current session's screen and stored into the user's CICS temporary storage queue. The portion of the screen that is copied is from the top left corner of the display to the end of the line that the cursor is currently positioned. Screen data is stored in the CICS temporary storage queue in a first in - first out manner.

- **CUT from=cursor,to=endfield,queue=zudata,session=3**
  
  8 characters of data are copied from the application screen running in PIE/CICS session 3 and stored as the ZUDATA variable. The copied screen area is from the cursor's present position to the end of the current field.
CUTEdit copies a selected portion of a screen and stores it to the PIE clipboard. Immediately afterwards, a PIE edit session is opened to view and modify the contents of the clipboard.

**Format**

```
FROM=screen_position
TO=screen_position
CUTEdit (,SESSION=[*|nn|session_name])
   [LIFO|FIFO]
   [.QUEUE=[CICS_TSQ|ZUDATA|ZUDATA*n]]
```

**Parameters**

- **FROM**
  - Starting position of the screen area for a cutedit operation. The default is CURSOR. The starting screen-position can be specified in the following ways:
    - CURSOR: Current cursor position.
    - (row,col): Row and column number of a 24 row by 80 column 3270 screen. For example, specifying (01,01) is the top left corner of a 3270 screen.
    - 0: Cursor position in the 3270 buffer. 0 would be the first position on the screen.
    - BEGFIELD: Beginning of the current field that the cursor is located.
    - ENDFIELD: End of the current field that the cursor is located.
    - BEGLINE: Beginning of the current line that the cursor is located.
    - ENDLINE: End of the current line that the cursor is located.

- **TO**
  - Ending position of the screen area for a cutedit operation. The default is ENDFIELD.
  - The ending screen position is specified with the same variables as the FROM parameter.

- **SESSION**
  - Session containing the screen image to be cut. The current session (*) is the default. The session can be identified by a two-digit session number nn or a 1 to 8-alphanumeric character session name.

- **LIFO FIFO**
  - Organization of screen areas cut and stored in the clipboard. Screens may be stored in last in, first out (LIFO) or first in, first out (FIFO) order. This option is used only when the clipboard is empty. The default value is LIFO.
Chapter 2  Environment Commands

**QUEUE**
Temporary storage area to store data copied from a screen with the CUTEdit command.
The temporary storage area can be one of the following:

**CICS_TSQ**
ID of a CICS temporary storage queue. The default.
It is an internally generated PIE queue name that is unique for each user. This queue can be edited using object name of PIE.CLIPBRD.

**ZUDATA**
PIE user data area. If this option is used, the length of the cut data is limited to 8 characters, and only one item can be cut at a time.

**ZUDATA_n**
PIE user data areas ZUDATA2 through ZUDATA5. If this option is used, the length of the copied screen data is limited to 16 characters. Only one item can be cut at a time.

If the ZUDATA or ZUDATA_n parameters are specified, copied data can be identified by the &ZUDATA and &ZUDATA2-5 PIE variables. These variables can be used in a command string as part of a PIE/CICS script.

**Example**

- **CUTEDIT fifo,from=(01,01),to=endline**

  Data is copied from the current session’s screen and stored into the user’s CICS temporary storage queue. The portion of the screen that is copied is from the top left corner of the display to the end of the line that the cursor is currently positioned. Screen data is stored in the CICS temporary storage queue in a first in - first out manner.
DISC

DISC disconnects all sessions and user exits from PIE/CICS. Sessions remain in their current state at the time the command is issued. The terminal is released from CICS and can be used to access another system.

DISC is intended to temporarily suspend PIE/CICS sessions and free the terminal for other tasks. Work can resume within suspended sessions by logging on to PIE/CICS again.

If a terminal is specified as logoff=no in the terminal directory, the terminal remains attached to CICS and is not released by the DISC command.

Format

| DISC |

Parameters

None

Example

- DISC

All active sessions of the current user are disconnected. The user exits from PIE/CICS. Sessions remain in their current state and can be resumed when the user logs on again to PIE/CICS.
DISCHOLD

DISCHOLD disconnects all PIE/CICS sessions and the user is returned to the Logon Director screen. Sessions remain in their current state when the command was issued. The terminal is released from CICS and can access another system.

DISCHOLD is intended to temporarily suspend PIE/CICS work from the current terminal. Work can resume within these sessions at a later time by logging on to PIE/CICS again.

Format

| Dischold |

Parameters

None

Example

- DISCHOLD

  All PIE/CICS sessions are disconnected and the terminal returns to the Logon Director screen. The sessions remain in their current state at the time the DISCHOLD command was entered. Work can resume again on these sessions by logging on to PIE/CICS.
DISPlay

DISPlay shows the current status of a PIE/CICS session. After entering the command, the Sessions menu is displayed with a message appearing beneath the Command line.

```
SES=2,TAS=0000003,ALOC=Y,OPEN=Y,KEYS=ON,HIDDEN=N,ACT=Y
```

The fields of the DISPLAY message are:

- **SES**: PIE/CICS session number
- **TAS**: 7-digit CICS task number (for conversational tasks only)
- **ALLOC**: Session allocated (Yes or No)
- **OPEN**: Session open (Yes or No)
- **KEYS**: PIE global keys active (ON or OFF)
- **HIDDEN**: Session hidden (Yes or No)
- **ACT**: Session active (Yes or No)

DISPlay is an alias of the Query command. See “Query” on page 59.

**Format**

```
DISPLAY sssssssssss
```

**Parameters**

- **ssssssss**
  
  1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name. An error message is displayed from the current menu if DISPLAY is entered without a session identifier.

**Examples**

- **DISPLAY 4**

  A message appears beneath the command line of the Sessions menu that shows the status of session 4 assigned to the current user.

- **DISPLAY CICSPROD**

  The status of a session named CICSPROD appears beneath the command line of the Sessions menu.
DRop

DRop closes and deallocates a session. All system and application resources used by the PIE/CICS sessions are freed and available for other work. The session is removed from the PIE/CICS Sessions menu.

Active PIE/CICS transactions are cancelled within a session closed by the DRop command. Also, any unsaved work is lost that was being performed within the session when it closed.

If the ALL parameter is specified with DRop, all allocated sessions for the current user are cancelled. All PIE/CICS resources are freed from the cancelled session(s).

If a session is designated as non-cancellable, a pop-up window appears in the session with a message indicating the session must be resumed and ended normally. The non-cancellable session remains inactive until it is resumed or ended normally.

DRop is an alias of the Cancel command. See “Cancel” on page 22.

Format

| DRop [ssssssss|ALL] |

Parameters

<table>
<thead>
<tr>
<th>sssssss</th>
<th>1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>All allocated sessions are cancelled for the current user.</td>
</tr>
</tbody>
</table>

Examples

- DRop 5

  Session 5 of the current user is closed and deallocated. Any active PIE/CICS transactions or work occurring within session 5 are cancelled.
- DR all

  All active sessions of the current user are closed and deallocated.
ENd

ENd returns to the preceding active PIE/CICS session. The current session is closed when the previous session is reopened. ENd can be invoked from the Command line, or by pressing the designated END key (PF3 by default).

ENd closes PIE/CICS sessions in the reverse order they were opened. Each invocation of the ENd command returns to a previous session until the last open session is closed. After that, ENd invokes the command specified in the Final Command field of the user’s current profile.

Format

| ENd |

Parameters

None

Example

- ENd

The current session is closed and the preceding active session is reopened.
ESCape

ESCape executes a PIE Exec command in session 0, the Sessions menu. Session 0 is unique and should be restricted to executing transactions that end by using the PIE Exec API to perform the END or RETURN commands.

If a conversational or pseudo-conversational transaction is executed in Session 0, no session switching should occur with either a PIE switch command or PF/PA key. This is because when switching out of session 0, the session is freed, and you cannot switch back to it. The PIE CANCEL process is not invoked for this session and unpredictable results may occur after switching out. Only non-conversational transactions should be executed in Session 0.

ESCape is useful for customers who are not licensed for MultiCICS and therefore restricted to a single session. ESCape gives them a second, utility session.

Format

```
ESCAPE pie_exec_command
```

Parameters

pie_exec_command

Examples

- ESCAPE RESEND RETURN

  Setting a PA key to ESCAPE RESEND RETURN effectively refreshes an application screen. This means that if any data is keyed or changed on the screen (via Erase EOF, for example), pressing the specified PA key before ENTER will cause the original screen to be redisplayed. You must use a PA key and no other key for this function.
EXit

EXit provides several options to exit from PIE/CICS or disconnect sessions. A command parameter specifies the particular method to exit from PIE/CICS or stop session work.

If EXit is entered without a parameter, a pop-up window appears listing the parameters. Each parameter has an associated PF key. Pressing the key is equivalent to entering EXit and the parameter together.

EXit is an alias of the following PIE/CICS commands:

- BYe
- CESF
- CSSF
- LOGOFF
- QUIT

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

EXit [DISC|DISCHOLD|LOGOFF|SIGNOFF|END PIE|LOCK]

Parameters

**DISC**

The user is disconnected from PIE/CICS and signed off from external security. The terminal is released from CICS and can be used for other applications.

DISC temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the EXit command was entered. Work can resume within these sessions at a later time by logging on to PIE/CICS again.

**DISCHOLD**

The user is disconnected from PIE/CICS and signed off from external security. The terminal returns to the Logon Director screen and remains attached to CICS.

DISCHOLD temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the EXit command was entered. Work can resume within these sessions at a later time by signing on to PIE/CICS again.

**LOGOFF**

All current user sessions are cancelled and the user is logged off PIE/CICS. Also, the user is signed off from external security. The terminal is released from CICS and can be used for other applications.

**SIGNOFF**

All current user sessions are cancelled and the user is signed off PIE/CICS and external security. The terminal returns to the Logon Director screen and remains attached to CICS.

**END PIE**

All current user sessions are cancelled and the user is logged off PIE/CICS. The terminal displays a blank screen and remains attached to CICS.
LOCK

The terminal is locked to prevent further keyboard input. The terminal is retained by CICS and sessions remain in their current state at the time the EXit command was issued. The terminal remains locked until the current user’s password is re-entered.

Examples

• EXit

A pop-up window appears with a list of exit options that are selected with PF keys.

• EXit LOCK

The terminal is locked, preventing any further input until the current user’s password is entered. Active sessions are disconnected until the terminal is unlocked.
Free

Free deallocates a session and makes it temporary rather than permanent. If the session is open when the Free command is issued, the session remains on the Sessions Menu but becomes an unallocated session. If the ALL parameter is specified with Free, all allocated sessions for the current user are deallocated.

Format

```
Free [ssssssss|ALL]
```

Parameters

- **ssssssss**  
  1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.
  
  If a session identifier is not specified with FRee, a pop-up window appears that lists all sessions. A PF key is assigned to each session. Pressing the key closes the associated session.

- **ALL**  
  All sessions are deallocated for the current user.

Examples

- **FREE CICSTEST**
  
  The CICSTEST session is deallocated. Any active PIE/CICS transactions or work occurring within session 5 are cancelled.

- **FREE ALL**
  
  All active sessions of the current user are deallocated.
**GOto**

GOto opens or resumes a previously opened session, or allocates and opens a temporary session. The temporary session is deallocated after it is closed.

If GOto is entered without parameters, a pop-up window lists all active sessions by their session number as they appear on the Sessions menu. A PF key is assigned to each session. Pressing a PF key opens the associated session.

GOto can be specified with positional keywords. These keywords define the search criteria to select a session to be opened. The search can be forward (NEXT keyword) or backward (PREV keyword) from the current open session where the GOto command is entered. Other keywords describe the required state of a session before it can be switched to and opened.

GOto is an alias of the SWItch command. See “SWItch” on page 72.

**Format**

```
GOTO {NEXT|PREV} [ssssssss] [application]
```

**Parameters**

- **ssssssss** 1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or the session name. GOto opens the active session identified by the session number or name. Temporary sessions must be assigned by session number. The SWItch pop-up window appears instead if an attempt is made to allocate and open a temporary session by session name.

- **application** Application assigned to the session or the command string that executes when the session is initially opened. Up to 66 characters in length. The Title field of the PIE/CICS Sessions menu shows the text of the application assigned to the session when it initially opens.

  If the application parameter is not specified, the existing application defined for the session in the user’s current profile executes when the session is initially opened.

- **NEXT** Positional keyword that conditionally selects the next unhidden session in ascending session number sequence. The next session is relative to the session number of the current open session where the GOto command is entered.
A switch to the next session is based on positional keywords that describe the required state of the next session before it can be switched to and opened:

**ALL**  
All sessions can be switched to and opened. The default.

**ALLOCATED**  
Any allocated session can be switched to and opened.

**USED**  
Any session that has been previously used during the period that the user has been logged on to PIE/CICS can be switched to and opened.

**UNUSED**  
Sessions that have not been used during the period that the user has been logged on to PIE/CICS can be switched to and opened.

**FREE**  
Sessions that have been deallocated during the period that the user has been logged on to PIE/CICS can be switched to and opened.

**CLOSED**  
Currently closed sessions can be switched to and opened.

**OPENED**  
Currently open sessions can be switched to and opened.

**PREV**  
Positional keyword that conditionally selects the previous unhidden session in descending session number sequence. The previous session is relative to the session number of the current open session where the GOto command is entered.

A switch to the previous session is based on positional keywords that describe the required state of the previous session before it can be switched to and opened. PREV uses the same session state keywords as the NEXT keyword.

**BACK**  
Positional keyword that unconditionally switches to the previous session the user was in before the current session.

**Examples**

- **GOto 5 access applid=cicsqury**
  
  Session 5 is opened and the CICS query region named CICSQURY is the application running in it. If session 5 is a permanent session, the ACCESS command supersedes the current application defined in the user's current profile.

- **go next**
  
  The next available unhidden session in ascending session number order is switched to and opened. The number of the next session is relative to the current session where the SWItch command was entered.

- **GOto back**
  
  A session switch occurs to the previous active session.
HElp

HElp displays descriptions of PIE/CICS Environment commands in a series of scrollable ISPF panels. Each online help includes a description of the command and its parameters.

An example of a command’s syntax is enclosed within a box that appears on the first online help panel. The example includes the command, its parameters, and any notation that must be used to assign a value to a parameter.

PIE/CICS help panels adhere to standard ISPF navigational conventions. Press PF8/20 to scroll forward or PF7/19 to scroll backwards. Pressing the END key or entering the END command closes online help and returns the user to the session that was open when the HElp command was entered.

Format

```plaintext
HElp [cccccccc]
```

Parameter

- `cccccccc` 1 to 8-character name of a PIE/CICS command.

If HELP is entered without a command name, an online help index appears that lists commands in alphabetical order. Commands can be selected from the index with the S line command. Also, commands can be selected by entering the command number or name listed in the # and Name fields of the help index.

Examples

- **HElp**

  An online help index appears that lists PIE/CICS commands. A description of a particular command can be selected from the list with the S line command or by specifying the command number or name shown on the menu.

  **HE CLOSE**

  A description of the PIE/CICS CLOSE command appears in an ISPF panel. After scrolling through the panels to read all information about this command, pressing the END key closes online help and returns to the open session where the HElp command was entered.
HIde

HIde excludes an active session listed on the Sessions menu from being opened by making an implicit reference to it with the NEXT or PREV parameters of the SWITCH or GOto commands. A hidden session is designated with an H to the right of the Title field of the Sessions menu.

A hidden session cannot be opened by toggling through the active sessions listed on the Sessions menu in numerical order. For example, if session 3 is hidden, the SWITCH NEXT command entered from session 2 opens session 4, which is the next unhidden session. Session 3 must be opened by making an explicit reference to it by session number or name.

Infrequently used applications can be placed in hidden sessions to allow users to switch directly to sessions running more highly used applications. Users do not have to toggle through these rarely used sessions if they are hidden.

By default, PIE/CICS sessions are unhidden. After a session is hidden with the HIde command, the default can be restored with the UNHIDE command. Refer to “Unhide” on page 79 for instructions to change the status of a hidden session and make it selectable with the SWITCH or GOto commands.

Format

```
HIde [ssssssss|ALL]
```

Parameters

- **ssssss**
  1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.

- **ALL**
  All current allocated user sessions are hidden. Sessions that are subsequently allocated are unhidden by default.

Example

- **HIde MARKET**

  The MARKET session is hidden. It cannot be opened by making an implicit reference to it with the NEXT or PREV parameters of the SWITCH and GOto commands.
LOCK

LOCK disables a terminal and prevents further input from the keyboard. The terminal is retained by CICS and sessions remain in their current state at the time the LOCK command is issued. LOCK is intended to protect a user’s sessions for short periods of time when their terminal is left unattended.

After issuing the LOCK command, a message appears on the screen for the user to enter their password to unlock the session and resume work.

```
Lock entered at: 14:28:40
Enter password and press enter to resume work, or
Press PF3 or PF15 to disconnect terminal
```

The terminal remains locked until the password is re-entered. If an attempt is made to unlock a terminal with an invalid password, the sessions associated with the terminal enter a DISCHOLD state after a specified number of failed log on attempts. Authorized personnel must use the Protected Terminals list to reset the terminal.

**Format**

```
LOCK
```

**Parameters**

None

**Examples**

- LOCK

  All PIE/CICS sessions are locked that are managed by the current terminal.
LOGOff provides several options to exit from PIE/CICS or stop work from occurring within sessions. A command parameter specifies the particular method to exit from PIE/CICS or stop session work.

If LOGOff is entered without a parameter, a pop-up window appears listing EXit parameters. Each parameter has an associated PF key. Pressing the key is equivalent to entering LOGOFF and the parameter together.

LOGOff is an alias of the following PIE/CICS commands:

- BYe
- CESF
- CSSF
- EXit
- QUIT

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

**Format**

```
LOGOFF [DISC|DISCHOLD|LOGOFF|SIGNOFF|END PIE|LOCK]
```

**Parameters**

- **DISC**
  The user is disconnected from PIE/CICS and signed off from external security. The terminal is released from CICS and can be used for other applications.
  DISC temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the LOGOff command was entered. Work can resume within these sessions at a later time by logging on to PIE/CICS again.

- **DISCHOLD**
  The user is disconnected from PIE/CICS and signed off from external security. The terminal returns to the Logon Director screen and remains attached to CICS.
  DISCHOLD temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the LOGOff command was entered. Work can resume within these sessions at a later time by signing on to PIE/CICS again.

- **LOGOFF**
  All current user sessions are cancelled and the user is logged off PIE/CICS. Also, the user is signed off from external security. The terminal is released from CICS and can be used for other applications.

- **SIGNOFF**
  All current user sessions are cancelled and the user is signed off PIE/CICS and external security. The terminal returns to the Logon Director screen and remains attached to CICS.

- **END PIE**
  All current user sessions are cancelled and the user is logged off PIE/CICS. The terminal displays a blank screen and remains attached to CICS.
Chapter 2  Environment Commands

LOCK

LOCK disables a terminal to prevents further input. The terminal is
retained by CICS and sessions remain in their current state at the time
the BYe command was issued. The terminal remains locked until the
current user's password is re-entered.

Examples

- LOGOFF
  A pop-up window appears with a list of exit options that are selected with PF keys.
- LOGOFF signoff
  All current user sessions are cancelled and the user is signed off PIE/CICS. The
terminal returns to the Logon Director screen and remains attached to CICS.
NAme

NAme assigns a session name that appears on the PIE/CICS Sessions menu. An uppercase session name appears in the Name field of the PIE/CICS Session menu to the right of the session number. A PIE/CICS session can be selected by its number or name.

NAme is an alias of the ALias command. See “ALias” on page 14.

Format

\[ \text{NAme} \ [\text{ssssssss}] \ [\text{name}] \]

Parameters

- **ssssss**  
  1 to 8-alphanumeric character session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.
  
  If a session identifier is not specified with the NAme command, a new session is opened. It is assigned the lowest available session number and the name specified with the NAme command. The session is assigned the default application defined in the user’s current profile.

- **name**  
  1 to 8-alphanumeric character session name that appears on the PIE/CICS Sessions menu.
  
  If a session name contains more than eight characters, the name is truncated and only the first eight characters appear in the Name field. The Name field is blank if the name parameter is omitted from the command.

Examples

- **NAME MRKTNG MARKET**
  
  The current MRKTNG session is renamed MARKET.

- **NAME MARKETING**
  
  A new session is opened and assigned the default application specified in the user’s current profile. The Name field of the Sessions menu contains MARKETIN because the specified name was truncated to the first eight characters.
NOTEC

NOTEC assigns an execution string to a session. The next time the session is started, the execution string is invoked as the initial command.

The execution string is permanently assigned to the session in the user’s current profile. The execution string is placed on the Execute field of the Sessions menu of the Profile utility.

Format

```
NOTEC sssssss [execution text]
```

Parameters

- `ssssssss`   1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or the session name as it appears on the current PIE/CICS menu.
- `execution text`   1 to 42-alphanumeric character string that is invoked as the initial command.

- **NOTEC SYSTEM CEMT I TERM**

  The CEMT transaction is assigned to the SYSTEM session. The next time SYSTEM is opened, the CEMT transaction executes as the initial session command.
NOtes

NOtes assigns a descriptive title to a session listed on the PIE/CICS Sessions menu. The title is permanently assigned to the session in the user’s current profile.

NOtes is an alias of the TITle command. See “Title” on page 77

Format

\[ \text{NOtes sssssss [title]} \]

Parameters

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or the session name as it appears on the current PIE/CICS menu.
- **title**: 1 to 42 -alphanumeric character string that appears in the Title field of the Sessions menu. If title is omitted, the Title field of the specified session is blank.

Example

- **NOTES MARKETNG Western Region Market Planning**
  
  The phrase Western Region Market Planning appears in the MARKETNG session’s Title field of the Sessions menu.

- **NO 6 Customer Service Tech Support**
  
  The description Customer Service Tech Support appears in the Title field of session 6 of the Sessions menu.
Chapter 2  Environment Commands

OPen

OPen allocates a new PIE/CICS session or opens an active session listed on the Sessions menu. If the session is currently open, the command string specified as a parameter replaces the session’s existing command string.

OPen is an alias of the START command. See “START” on page 71.

Format

OPen [ssssssss] [command string]

Parameters

ssssssss 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.

If neither a session identifier nor a command string are specified with the OPEN command, a temporary PIE/CICS session is allocated and opened. This session executes the default application specified in the user’s current profile. The temporary session is cancelled after the default application is closed.

command string 1 to 66-character string interpreted as a command sequence that is executed when the session is initially opened.

The command string temporarily replaces the application or commands defined for the session in the user’s current profile. After the session is closed, the session reverts to the command string defined in the user’s profile.

Examples

• OP profile

An active session named PROFILE is opened.

• OPen 3 menu piemadm

Session 3 is opened and made active. The existing application or command sequence associated with session 3 is replaced with the MENU command that displays the PIE/CICS System Administrator menu. After closing the System Administrator menu, session 3 reverts to the application or command associated with it in the user’s profile.

• OP

A temporary session is opened and made active. The session displays the default application defined in the user’s current profile.
PASTE

PASTE places a previously cut screen area onto another screen. The data is taken from the PIE clipboard or a designated storage area.

Format

```
FROM=screen_position
,TO=screen_position
PASTE ,SESSION=[*|nn|session_name]
```

Parameters

FROM  
Starting position of the screen area for the paste operation. The default is CURSOR. The starting screen-position of a paste can be specified in the following ways:

- CURSOR: Current cursor position.
- (row,col): Row and column number of a 24 row by 80 column 3270 screen. For example, specifying (01,01) is the top left corner of a 3270 screen.
- position: Cursor position in the 3270 buffer. 0 would be the first position on the screen.
- BEGFIELD: Beginning of the current field that the cursor is located.
- ENDFIELD: End of the current field that the cursor is located.
- BEGLINE: Beginning of the current line that the cursor is located.
- ENDLINE: End of the current line that the cursor is located.

TO  
Ending position of the screen area for the paste operation. The default is ENDFIELD.

The ending screen position is specified with the same variables as the FROM parameter.

SESSION  
Session containing the screen image to be pasted. The current session (*) is the default. The session can be identified by a two-digit session number nn or a 1 to 8-alphanumeric character session name.

Example

- `PASTE from=cursor,to=endfield,session=3`

  Previously cut data is copied from the PIE clipboard and pasted to PIE/CICS session 3. The copied screen area is pasted from the cursor’s present position to the end of the current field.
**PIE**

PIE explicitly invokes a PIE command. The first word after the PIE command is regarded as a PIE command. An attempt is made to invoke the command. If the command is invalid, an error message and a PIE pop-up window appear on the screen. Users can select an Environment command from the pop-up window with an associated PF key shown within the window.

If PIE is entered without a command string, a pop-up window appears with a partial list of PIE/CICS Environment commands. Each command has an associated key. Pressing the key is equivalent to entering PIE and the command together.

PIE can be used to enter an Environment command from a CICS blank screen without entering the escape string.

**Format**

```
PIE [cccccccc]
```

**Parameters**

- `cccccccc` Valid PIE Environment command.

**Examples**

- **PIE**
  
  The PIE pulldown window is displayed with a subset of PIE commands that can be selected by a listed PF key.

- **PIE SESSLONG**
  
  The PIE command is entered from the CICS blank screen without an escape string. The user is sent to the SESSLONG session.
PRINT

PRINT issues a CICS screen print of the current session screen. Typically, PRINT is associated with either a PF key or a session with the PIE profile utility. The screen print is directed to the primary CICS printer defined with either the PRINITID command or through the User Directory.

Format

```
PRINT
```

Parameters

None

Example

- PRINT

  The current session screen is printed as a CICS screen print to the primary printer.
PROfile

PROfile starts the PIE Profile utility to modify individual user or group profiles. The PIE Profile editor consists of a series of ISPF menus. Each profile menu contains fields that can be modified to change a functionally related set of parameters. Profile parameters are organized into the following functional groups:

- Terminal characteristics
- Functions assigned to PF keys
- Default sessions and their initial command sequence

If PROfile is specified without a parameter, the PIE/CICS Profile menu lists individual options to modify specific groups of profile parameters.

Format

```
[TERMINAL|1]
PROfile [KEYS|3]
[SESSIONS|4]
```

Parameters

- TERMINAL or 1: Terminal Characteristics menu is displayed. This menu has fields to modify how commands, messages, and PF keys operate for all PIE/CICS terminals.
- KEYS or 3: Global PIE Keys menu is displayed. This menu has fields to assign commands or functions to session PF keys.
- SESSIONS or 4: Sessions Configuration menu is displayed. This menu has fields to assign default sessions and applications to a profile.

Example

- PROFILE KEYS

  The Global PIE Keys menu is displayed to assign commands or functions to session PF keys.
Query

QUERY shows the current status of a PIE/CICS session. After entering the command, the Sessions menu is displayed with a message appearing beneath the Command line.

\[ \text{SES}=2, \text{TAS}=0000003, \text{ALOC}=Y, \text{OPEN}=Y, \text{KEYS}=Y, \text{HIDDEN}=N, \text{ACT}=Y \]

The fields of the QUERY message are:

- **SES**: PIE/CICS session number
- **TAS**: 7-digit CICS task number (for conversational tasks only)
- **ALLOC**: Session allocated (Yes or No)
- **OPEN**: Session open (Yes or No)
- **KEYS**: PIE global keys active (ON or OFF)
- **HIDDEN**: Session hidden (Yes or No)
- **ACT**: Session active (Yes or No)

Query is an alias of the DISPlay command. See “DISPlay” on page 37.

**Format**

```
Query sssssssss
```

**Parameters**

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.

**Example**

- **Query 1**

  The Sessions menu is opened and a status message for session 1 appears beneath the Command line.
QUIT

QUIT provides several options to exit from PIE/CICS or stop work from occurring within sessions. A command parameter specifies the particular option to exit from PIE/CICS or stop session work.

If QUIT is entered without a parameter, a pop-up window appears listing QUIT parameters. Each parameter has an associated PF key. Pressing the key is equivalent to entering QUIT and the parameter together.

QUIT is an alias of the following PIE/CICS commands:

- BYe
- CESF
- CSSF
- Exit
- LOGOFF

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

QUIT [DISC|DISCHOLD|LOGOFF|SIGNOFF|END PIE|LOCK]

Parameters

<table>
<thead>
<tr>
<th>DISC</th>
<th>The user is disconnected from PIE/CICS and signed off from external security. The terminal is released from CICS and can be used for other applications. DISC temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the QUIT command was entered. Work can resume within these sessions at a later time by logging on to PIE/CICS again.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISCHOLD</td>
<td>The user is disconnected from PIE/CICS and signed off from external security. The terminal returns to the Logon Director screen and remains attached to CICS. DISCHOLD temporarily suspends PIE/CICS work from the current terminal. User sessions remain in their current state at the time the QUIT command was entered. Work can resume within these sessions at a later time by signing on to PIE/CICS again.</td>
</tr>
<tr>
<td>LOGOFF</td>
<td>All current user sessions are cancelled and the user is logged off PIE/CICS. Also, the user is signed off from external security. The terminal is released from CICS and can be used for other applications.</td>
</tr>
<tr>
<td>SIGNOFF</td>
<td>All current user sessions are cancelled and the user is signed off PIE/CICS and external security. The terminal returns to the Logon Director screen and remains attached to CICS.</td>
</tr>
<tr>
<td>END PIE</td>
<td>All current user sessions are cancelled and the user is logged off PIE/CICS. The terminal displays a blank screen and remains attached to CICS.</td>
</tr>
</tbody>
</table>
LOCK

LOCK disables a terminal to prevents further input. The terminal is retained by CICS and sessions remain in their current state at the time the BYe command was issued. The terminal remains locked until the current user’s password is re-entered.

Example

- QUIT

  A pop-up window appears with a list of exit options that are selected with PF keys.

- QUIT logoff

  All sessions are cancelled and the user is logged off PIE/CICS. The terminal is released from CICS and can be used for other applications.
Chapter 2 Environment Commands

RESTART

RESTART closes and reopens the current session. If a command string is specified with RESTART, the current session reopens and executes the command string as the initial session application. Otherwise, the initial command specified in the user’s current profile executes when the session reopens.

RESTART is intended to refresh a session with the default command string that normally executes when a session is initially opened. Also, the command gives the capability to dynamically start a session with another initial command string without having to make changes to the current user profile.

Format

```
REStart [application]
```

Parameters

- **application** 1 to 66-character string interpreted as a command sequence that executes when the session is reopened.

Examples

- **RESTART**
  The current session closes and reopens. The initial command defined for the session in the user’s current profile executes when the session reopens.

- **RESTART PROF 3**
  The current session closes and reopens to the Profile Keys menu.
RETURn

Return closes the current session and opens the previous PIE/CICS session that preceded the current session. Return can be invoked from the Command line or pressing the designated RETURN key (PF4 by default).

Return closes PIE/CICS sessions in the reverse order that they were opened. Each invocation of the command returns to a previous session until the last open session is closed and the Sessions menu is displayed. After that, Return invokes the CLOSE pop-up window.

**Format**

| RETURn |

**Parameters**

None

**Example**

- RETurn

  The current session is closed and the preceding session is reopened.
Chapter 2  Environment Commands

RETURNA

RETURNA exits from the Sessions menu and returns to the most recent open PIE/CICS session. If there are no more open sessions, the final command defined in the user's profile is executed. Normally, the final command is the LOGOFF command that ends all PIE/CICS sessions.

Format

```
RETURNA
```

Parameters

None

Example

- RETURNA

  The user returns to the most recent open PIE/CICS session or executes the final command specified in the profile.
**SESqlon**

SESqlon switches to the PIE/CICS Sessions menu. If SESqlon is issued from the Sessions menu, the most recent open session is resumed.

**Format**

<table>
<thead>
<tr>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>SESqlon</td>
</tr>
</tbody>
</table>

**Parameters**

None

**Example**

- SESqlon
  
The user switches directly to the Sessions menu.
SESMenu

SESMENU switches to the Sessions menu. If you are currently at the Sessions menu, you remain there. This command differs from the SESsion command, which returns to the last used session if the Sessions menu is open when the SESsion command is issued.

**Format**

```
SESMenu
```

**Parameters**

- None

**Example**

- SESMenu

  The user switches directly to the Sessions menu. If the command is issued from the Sessions menu, the user remains at the Sessions menu and does not return to the most recent open session.
SET

SET temporarily assigns functions to PIE/CICS global keys for the duration of the current session. The default functions assigned to global keys are resumed after the session is closed and reopened. Functions are permanently assigned to global keys with the Profile utility.

**Format**

```
(SWSTR ssss)
(ESCTR eeee)
(PFnn fffff)
(PAnn aaaaa)
(ENTER)
(ENDKEY PFnn)
SET (RETKEY PFnn)
(DELIM ddddd)
(TITLE [ON|OFF])
(MSG [ON|OFF])
(KEYS [ON|OFF]SON nn|SOFF nn|SBON|SBOFF|SBASIS]sesid toggle_key)
(SCROLL [PAGE|HALF|DATA|CURSOR])
(CMDTRACE [ON|OFF])
(KEYS [])
```

**Parameters**

- **SWSTR**: 1 to 4-alphanumeric character escape string that designates PIE/CICS commands. Alias of the ESCSTR parameter.
- **ESCTR**: 1 to 4-alphanumeric character escape string that designates PIE/CICS commands. Alias of the SWSTR parameter.
- **PFnn aaaaa**: Function assigned to a PF key. The PF key must be designated with a two-digit number (e.g. PF01).
- **PAnn**: Function assigned to a PA key. The PA key must be designated with a two-digit number (e.g. PA01).
- **ENTER**: Sets the ENTER key to an executable statement.
- **ENDKEY**: System end function assigned to the specified PFnn key.
- **RETKEY**: Second system end key.
- **DELIM dddd**: Delimiter to separate commands within a string.
- **TITLE**: Title fields of PIE/CICS menus are displayed or left blank.
  - **ON**: Title fields of PIE/CICS menus are displayed. The default.
  - **OFF**: Title fields of PIE/CICS menus are not shown.
- **MSG**: Display short messages in addition to long messages. Used primarily for debugging problems. Short messages appear in the upper right corner of the screen.
  - **ON**: The message number appears with the text of the
Chapter 2  Environment Commands

message.

OFF Suppress the display of message numbers.

KEYS PIE/CICS global keys are active or inactive.

ON Sets PIE global keys on in all sessions. 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.

OFF Sets PIE global keys off in all sessions.

SON Sets PIE global keys on in session nn.

SOFF Sets PIE global keys off in session nn.

If SON and SOFF are specified without a session identifier, global key definitions are set for the current session.

If toggle-key is specified with the SOFF parameter, this key retains its PIE Global key definition in the new session. Usually, this key has the SWITCH command associated with it to facilitate switching out of a session.

Use the DISPLAY or Query commands to obtain the current state of PIE/CICS global keys.

SBON Sets PIE global keys on when switching back to session nn.

SBOFF Sets PIE global keys off when switching back to session nn.

SBASIS Maintain the current state of PIE global keys when switching back to session nn.

sesid Name or number that identifies a PIE session.

toggle_key Name of the key to toggle the state of PIE session commands to inactive or inactive.

SCROLL Default length that PIE/CICS menus and help panels are scrolled forward or backward.

PAGE Scroll page length composed of nn lines determined from ISPF session profile.

HALF Scroll one half page length composed of nn lines determined from ISPF session profile.

DATA Scroll nn lines set by the length of the data field within the panel.

CURSOR Scroll nn lines set by the length of the current cursor
position from the top of the panel.

CMDTRACE  Sets the PIE EXEC and MultiCICS and NetGate command trace facility on.
ON        Command tracing active.
OFF       Command tracing inactive. The default.

All commands executed from the current terminal are written to a trace Temporary Storage queue (TSQ). The contents of the TSQ can be viewed with the BROWSE PIE.TRACE command.
The TSQ ID is composed of the following elements:
1-3       PIE TSQ prefix
4         T
5-8       CICS termid

Examples

- Set swstr $$
  The command switch string is set to $$ for the duration of the current session.
- Set scroll page
  ISPF panels are scrolled forward or backward by page length.
- Set scroll page
  ISPF panels are scrolled forward or backward by page length.
- set delim *
  Multiple commands are separated by an asterisk (*).
- set keys sboff payrl pa1
  Set PIE global keys off when switching back to the PAYRL session. PA1 is the designated toggle key.
Chapter 2  Environment Commands

**SKIPnext**

SKIPnext disables PIE/CICS global key functions for the next n number of terminal inputs. Normally, SKIPNEXT is used to disable global keys in the case where an application uses the same key defined as a PIE global key.

**Format**

```
SKIPnext n
```

**Parameters**

- **n**  Number of pending terminal inputs with global key functions disabled.

**Example**

- `SKIPNEXT 4`
  PIE/CICS global keys are disabled for the next four terminal inputs.
**STart**

STart allocates a new PIE/CICS session or opens an active session listed on the Sessions menu. If the session is currently open, the command string specified as a parameter replaces the session’s existing command string.

If there is no title for this session the execution text will also become the title.

STart is an alias of the OPen command. See “OPen” on page 54.

**Format**

```
STart [ssssssss] [command_string]
```

**Parameters**

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a number in a range from 1 to 99 or a session name.

  If neither a session identifier nor a command string are specified with the STart command, a temporary PIE/CICS session is allocated and opened. This session executes the default application specified in the user’s current profile. The temporary session is cancelled after the default application is closed.

- **command string**: 1 to 66-character string interpreted as a command sequence that is executed when the session is initially opened. The command string appears in the Title field of the Sessions menu.

  The command string temporarily replaces the application or commands defined for the session in the user’s current profile. After the session is closed, the session reverts to the default application defined in the user’s profile.

**Examples**

- **STart MKTNG**

  The MKTNG session is opened and becomes the current session.

- **ST 6 access cicsprd**

  Start session 6 and access the CICSPRD region.
# SWItch

SWItch opens or resumes a previously opened session, or allocates and opens a temporary session. The temporary session is deallocated after it is closed.

If SWItch is entered without parameters, a pop-up window lists all active sessions by their session number as they appear on the Sessions menu. A PF key is assigned to each session. Pressing a PF key opens the associated session.

SWItch can be specified with positional keywords. These keywords define the search criteria to select a session to be opened. The search can be forward (NEXT keyword) or backward (PREV keyword) from the current open session where the SWItch command is entered. Other keywords describe the required state of a session before it can be switched to and opened.

SWItch is an alias of the GOto command. See “GOto” on page 44.

## Format

<table>
<thead>
<tr>
<th>sssssss</th>
<th>application</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACK</td>
<td></td>
</tr>
</tbody>
</table>

**SWItch {NEXT|PREV} [ssssssss] [application]**

**OPENED**

**ALLOCATED**

**USED**

**UNUSED**

**FREE**

**CLOSED**

**ALL**

## Parameters

- **ssssssss**
  - 1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or the session name.
  - SWItch opens the active session identified by the session number or name.
  - Temporary sessions must be assigned by session number. The SWItch pop-up window appears instead if an attempt is made to allocate and open a temporary session by session name.

- **application**
  - Application assigned to the session or the command string that executes when the session is initially opened. Up to 66 characters in length. The Title field of the PIE/CICS Sessions menu shows the text of the application assigned to the session when it initially opens.
  - If the application parameter is not specified, the existing application defined for the session in the user’s current profile executes when the session is initially opened.

- **NEXT**
  - Positional keyword that conditionally selects the next unhidden session in ascending session number sequence. The next session is relative to
the session number of the current open session where the SWItch
command is entered.

A switch to the next session is based on positional keywords that
describe the required state of the next session before it can be
switched to and opened:

**ALL** All sessions can be switched to and opened. The
default.

**ALLOCATED** Any allocated session can be switched to and opened.

**USED** Any session that has been previously used during the
period that the user has been logged on to PIE/CICS
can be switched to and opened.

**UNUSED** Sessions that have not been used during the period
that the user has been logged on to PIE/CICS can be
switched to and opened.

**FREE** Sessions that have been deallocated during the period
that the user has been logged on to PIE/CICS can be
switched to and opened.

**CLOSED** Currently closed sessions can be switched to and
opened.

**OPENED** Currently open sessions can be switched to and
opened.

**PREV** Positional keyword that conditionally selects the previous unhidden
session in descending session number sequence. The previous session
is relative to the session number of the current open session where the
SWItch command is entered.

A switch to the previous session is based on positional keywords that
describe the required state of the previous session before it can be
switched to and opened. PREV uses the same session state keywords
as the NEXT keyword.

**BACK** Positional keyword that unconditionally switches to the previous
session the user was in before the current session.
Chapter 2  Environment Commands

Examples

- **SWItch 5 access applid=cicsqry**
  
  Session 5 is opened and the CICS query region named CICSQURY is the application running in it. If session 5 is a permanent session, the ACCESS command supersedes the current application defined in the user’s current profile.

- **SWI next**
  
  The next available session in ascending session number order is switched to and opened. The number of the next session is relative to the current session where the SWItch command was entered.

- **SWITCH back**
  
  The previous session is switched to and reopened.
SWITCHR

SWITCHR opens a session and starts a supplied REXX script that operates according to parameters passed with the command.

Format

```
SWITCHR [ssssssss] [rexx_script] [rexx_args]
```

Parameters

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or the session name.
- **rexx_script**: Name of the REXX script that starts when the session is opened.
  
  **Note:**
  This script cannot be executed in a session that is currently executing another REXX or PIE script
- **rexx_args**: REXX program arguments passed to the REXX script.
Chapter 2  Environment Commands

TERM

TERM closes and deallocates a session. All system and application resources used by the PIE/CICS sessions are available for other work. The session is removed from the PIE/CICS Sessions menu.

If the ALL parameter is specified with TERM, all allocated sessions for the current user are cancelled. All PIE/CICS resources are freed from the cancelled session(s).

If a session is designated as non-cancellable, a pop-up alert window appears in the affected session containing a message that the session must be resumed and ended normally. The non-cancellable session remains inactive until it is resumed or ended.

TERM is an alias of the CANcel command. See “CANcel” on page 22.

Format

```
TERM [ssssssss|ALL]
```

Parameters

- **ssssssss**  
  1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or the session name as it appears on the current PIE/CICS menu.

- **ALL**  
  All PIE/CICS sessions are closed and deallocated.

Examples

- **TERM 5**  
  Session 5 is closed and deallocated.

- **TE ALL**  
  All sessions are closed and deallocated.
Title

Title assigns a descriptive title to a session listed on the PIE/CICS Sessions menu.
Title is an alias of the NOTE command. See “NOTes” on page 53.

Format

| Title ssssssss [title] |

Parameters

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or the session name as it appears on the current PIE/CICS menu.
- **title**: 1 to 79 alphanumeric character string that appears in the Title field of the Sessions menu. If title is omitted, the Title field of the specified session is blank.

Example

- **Title MARKETNG Western Region Market Planning**
  The phrase Western Region Market Planning appears in the MARKETNG session’s Title field of the Sessions menu.
- **Title 6 Customer Service Tech Support**
  The description Customer Service Tech Support appears in the Title field of session 6 of the Sessions menu.
TOGgle

TOGgle enables or disables the functions assigned to PIE/CICS global keys. PF and PA keys are enabled or disabled by switching to their current alternative state with the TOGgle command.

By default, global keys are active. At first invocation, TOGgle disables the function of all PF and PA keys except the designated toggle key. Reentering the TOGgle command, or pressing the toggle key, restores the function of PIE/CICS global keys.

Option 3, Keys, of the Profile menu shows the functions assigned to PIE/CICS global keys.

Format

```
TOGgle [PA01|toggle_key]
```

Parameters

- **toggle_key**: 4-character PF or PA key identifier. This key enables the functions assigned to global keys. PA01 is the default.

Example

- **TOGgle PA01**

  PIE/CICS global keys are disabled if they were active when the command was invoked. PA01 is assigned as the toggle key.
Unhide

Unhide returns a hidden session to the list of active sessions that can be opened with the SWITCH or GOto commands.

By default, a session is unhidden. A session must be hidden before it can be made accessible again with the Unhide command.

Format

| Unhide [ssssssss | ALL] |

Parameters

- **ssssssss**: 1 to 8-character alphanumeric session identifier. A session identifier can be a session number in a range from 1 to 99 or a session name as it appears on a PIE/CICS menu. The identified session is unhidden and can be made the active session by either the SWITCH NEXT or SWITCH PREV commands.

- **ALL**: All PIE/CICS sessions are unhidden and accessible by the SWITCH or GOto commands.

Examples

- **UNHIDE 4**

  Session 4 is unhidden and can be opened with an implicit reference by the SWITCH NEXT or SWITCH PREV commands.

- **U ALL**

  All sessions are unhidden and can be opened with an implicit reference by the SWITCH NEXT or SWITCH PREV commands.
VIEW CAPTURE

VIEW CAPTURE permits authorized users to display a current session screen of another user from their own terminal. VIEW CAPTURE is intended for helpdesk personnel to remotely diagnose user problems.

If VIEW CAPTURE is invoked without parameters, a menu appears with fields to select the captured session display by userid, terminal, or session identifier. Completing appropriate menu fields is equivalent to entering VIEW CAPTURE with parameters.

User screens are selected for viewing by a combination of userid and terminalid/sessionid parameters.

**Format**

```
VIEW CAPTURE [userid] [terminalid] [sessionid]
```

**Parameters**

- **userid**: 1 to 20-alphanumeric character user ID. The wildcard character (*) may be specified to select any user logged on at the terminal specified by the terminalid parameter.

- **terminalid**: 1 to 8-alphanumeric character VTAM node name or 4-character CICS termid. The View function is invoked for the user at this terminal. If userid is also supplied, the user must be signed on to PIE at that terminal. A colon (:) must be placed in front of the 4-character identifier to be interpreted as a CICS termid.

- **sessionid**: Number or name of a user's session containing the view to be captured. Specifying ALL or 0 selects the user's current Session menu, which displays the sessions available for viewing.
Examples

- **VIEW CAPTURE USER1**
  
  The active screen of USER1 is displayed.

- **VIEW CAPTURE USER:R0C8**
  
  The active screen of CICS terminal R0C8 is viewed. USER1 must be signed on to this terminal.

- **VIEW CAPTURE USER1 LDEV04F ALL**
  
  The Sessions menu for USER1 is viewed from the terminal with a VTAM node ID of LDEV04F.
Chapter 2  Environment Commands

VIEW

VIEW enables users to copy a selected portion of a screen and paste it into an unprotected area of another screen. Based upon the initial command parameter, screen areas are copied, edited, and pasted in separate operations.

Format

```
VIEW {CUT|CUTEDIT|PASTE} {,SESSION=[*|nn|session_name]} {,LIFO|FIFO} {,QUEUE=[CICS_TSQ|ZUDATA|ZUDATA_n]}
```

Parameters

The first VIEW parameter must be either CUT, CUTEDIT, or PASTE. After that, parameters can be specified in any order. If a parameter is not specified with the command, the default value is assigned.

**CUT**
Operation to copy a selected portion of a screen and store it to the PIE clipboard or a designated storage area.

**CUTEDIT**
Operation to copy a selected portion of a screen and store it to the PIE clipboard. Immediately afterwards, a PIE edit session is opened to edit the contents of the clipboard.

**PASTE**
Operation to place a previously cut screen area onto another screen.

**FROM**
Starting position of the screen area for a cut or a paste operation. The default is CURSOR. The starting screen-position of a cut or paste can be specified in the following ways:

- **CURSOR**  Current cursor position.
- **(row,col)**  Row and column number of a 24 row by 80 column 3270 screen. For example, specifying (01,01) is the top left corner of a 3270 screen.
- **position**  Cursor position in the 3270 buffer. 0 would be the first position on the screen.
- **BEGFIELD**  Beginning of the current field that the cursor is located.
- **ENDFIELD**  End of the current field that the cursor is located.
- **BEGLINE**  Beginning of the current line that the cursor is located.
- **ENDLINE**  End of the current line that the cursor is located.

**TO**
Ending position of the screen area for a cut or paste operation. The default is ENDFIELD. The ending screen position is specified with the same variables as the FROM parameter.
SESSION  Session containing the screen image to be cut or pasted. The current session (*) is the default. The session can be identified by a two-digit session number nn or a 1 to 8-alphanumeric character session name.

LIFO FIFO  Organization of screen areas cut and stored in the clipboard. Screens may be stored in last in, first out (LIFO) or first in, first out (FIFO) order. This option is used only when the clipboard is empty. The default value is LIFO.

QUEUE  Temporary storage area to store data copied from a screen with the VIEW CUT or CUTEDIT commands. The temporary storage area can be one of the following:

<table>
<thead>
<tr>
<th>QUEUE</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS_TSQ</td>
<td>ID of a CICS temporary storage queue. The default. It is an internally generated PIE queue name that is unique for each user. This queue can be edited using object name of PIE.CLIPBRD.</td>
</tr>
<tr>
<td>ZUDATA</td>
<td>PIE user data area. If this option is used, the length of the cut data is limited to 8 characters, and only one item can be cut at a time.</td>
</tr>
<tr>
<td>ZUDATAn</td>
<td>PIE user data areas ZUDATA2 through ZUDATA5. If this option is used, the length of the copied screen data varies from 8 to 16 characters, depending upon the variable used to store the data. Only one item can be cut at a time. If the ZUDATA or ZUDATAn parameters are specified, copied data can be identified by the &amp;ZUDATA and &amp;ZUDATA2-5 PIE variables. These variables can be used in a command string as part of a PIE/CICS script.</td>
</tr>
</tbody>
</table>
Examples

- **VIEW CUT fifo, from=(01,01), to=end line**
  
  Data is copied from the current session’s screen and stored into the user’s CICS temporary storage queue. The portion of the screen that is copied is from the top left corner of the display to the end of the line that the cursor is currently positioned. Screen data is stored in the CICS temporary storage queue in a first in -first out manner.

- **VIEW CUT from=cursor, to=endfield, queue=zudata, session=3**
  
  8 characters of data are copied from the application screen running in PIE/CICS session 3 and stored as the ZUDATA variable. The copied screen area is from the cursor’s present position to the end of the current field.

- **VIEW CUTEDIT fifo, from=(01,01), to=endline**
  
  Data is copied from the current session’s screen and stored into the user’s CICS temporary storage queue. The portion of the screen that is copied is from the top left corner of the display to the end of the line that the cursor is currently positioned. Screen data is stored in the CICS temporary storage queue in a first in -first out manner.

- **VIEW CUTEDIT from=cursor, to=endfield, queue=zudata4**
  
  8 characters of data are copied from the application screen running in PIE/CICS session 3 and stored as the QZUDATA4 variable. The copied screen area is from the cursor’s present position to the end of the current field.
Chapter 3  Application Commands

This chapter describes PIE/CICS Application commands. The following table lists the commands, their parameters, and a page reference for more information. Some commands are applicable to an entire PIE system. Other commands are provided by optional PIE/CICS components. These optional components must be installed before the commands can be used. The title of the command lists the product that must be installed to use an optional command.

Each command includes a syntax diagram. Refer to the Preface for a description of the syntax notation to indicate whether a command parameter, variable, or other values are required, optional, or included by default.

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameters</th>
<th>Page Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCesS</td>
<td>applid [VTERM=virtual_terminal] [VPOOL=virtual_terminal_prefix]</td>
<td>See page 89, “ACCesS (NetGate Only)”</td>
</tr>
<tr>
<td>AutoEnd</td>
<td>None</td>
<td>See page 91, “AutoEnd”</td>
</tr>
<tr>
<td>ATI</td>
<td>None</td>
<td>See page 92, “ATI”</td>
</tr>
<tr>
<td>ATTach</td>
<td>None</td>
<td>See page 93, “ATTach”</td>
</tr>
<tr>
<td>BROWSE</td>
<td>[groupid.]textname</td>
<td>See page 94, “BROWSE”</td>
</tr>
<tr>
<td>BROWSEN</td>
<td>[groupid.]textname</td>
<td>See “BROWSE” on page 95.</td>
</tr>
<tr>
<td>CEnd</td>
<td>None</td>
<td>See page 96, “ConfirmEnd”</td>
</tr>
<tr>
<td>CICPRT</td>
<td>None</td>
<td>See page 97, “CICPRT”</td>
</tr>
<tr>
<td>CLRRst</td>
<td>None</td>
<td>See page 98, “CLRRst”</td>
</tr>
<tr>
<td>CMD</td>
<td>transid [parameters]</td>
<td>See page 99, “CMD”</td>
</tr>
<tr>
<td>DSR</td>
<td>None</td>
<td>See “Dsr” on page 100.</td>
</tr>
<tr>
<td>EC</td>
<td>[session_command] [parameters]</td>
<td>See page 101, “EC”</td>
</tr>
<tr>
<td>EDIT</td>
<td>[groupid.]textname</td>
<td>See “EDIT” on page 102.</td>
</tr>
<tr>
<td>ENCP</td>
<td>None</td>
<td>See “ENCP” on page 103.</td>
</tr>
<tr>
<td>ENV</td>
<td>[session_command] [parameters]</td>
<td>See page 104, “ENV”</td>
</tr>
<tr>
<td>GETC</td>
<td>None</td>
<td>See “GETC” on page 105.</td>
</tr>
<tr>
<td>GOTO</td>
<td>[nn</td>
<td>name]</td>
</tr>
<tr>
<td>KEY</td>
<td>[3270_key]</td>
<td>See “KEY” on page 108.</td>
</tr>
<tr>
<td>LIST</td>
<td>[profiles</td>
<td>type] [ALL</td>
</tr>
<tr>
<td>LOAD</td>
<td>[function] [queue] [group] [member]</td>
<td>See “Load” on page 112.</td>
</tr>
</tbody>
</table>
### Chapter 3  Application Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameters</th>
<th>Page Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGMode</td>
<td>SNA=logmode, NONSNA=logmode, TERMINAL=nodename, EXCMSG RESET, TERMINAL=nodename, EXCMSG INQ, TERMINAL=nodename</td>
<td>See “LOGMode” on page 113.</td>
</tr>
<tr>
<td>MENU</td>
<td>[main] [groupid] menuname</td>
<td>See “MENU” on page 114.</td>
</tr>
<tr>
<td>MSG</td>
<td>None</td>
<td>See “MSG” on page 115.</td>
</tr>
<tr>
<td>NOAE</td>
<td>None</td>
<td>See “NOAE” on page 116.</td>
</tr>
<tr>
<td>NoATI</td>
<td>None</td>
<td>See “NoATI” on page 117.</td>
</tr>
<tr>
<td>NOATTach</td>
<td>sssssss [title]</td>
<td>See “NoATTach” on page 118.</td>
</tr>
<tr>
<td>NOCE</td>
<td>None</td>
<td>See “NOCE” on page 119.</td>
</tr>
<tr>
<td>NOCICPRT</td>
<td>none</td>
<td>See “NOCICPRT” on page 120.</td>
</tr>
<tr>
<td>NOCLRRST</td>
<td>None</td>
<td>See “NOCLRRST” on page 121.</td>
</tr>
<tr>
<td>NODSR</td>
<td>None</td>
<td>See “NODSR” on page 122.</td>
</tr>
<tr>
<td>NOEnCP</td>
<td>None</td>
<td>See “NoEnCP” on page 123.</td>
</tr>
<tr>
<td>NOMSG</td>
<td>None</td>
<td>See “NoMSG” on page 124.</td>
</tr>
<tr>
<td>NON3270</td>
<td>None</td>
<td>See “NON3270” on page 125.</td>
</tr>
<tr>
<td>NOOPT</td>
<td>None</td>
<td>See page 126, “NOOPT (NetMizer Only)”.</td>
</tr>
<tr>
<td>NOPASSP</td>
<td>None</td>
<td>See page 127, “NoPassP”.</td>
</tr>
<tr>
<td>NORBS</td>
<td>None</td>
<td>See page 128, “NoRBS (NetMizer Only)”.</td>
</tr>
<tr>
<td>NOTSQSUB</td>
<td>None</td>
<td>See page 129, “NOTSQSUB (MultiCICS Only)”.</td>
</tr>
<tr>
<td>NOUCt</td>
<td>None</td>
<td>See page 130, “NOUCt”.</td>
</tr>
<tr>
<td>OPT</td>
<td>None</td>
<td>See page 131, “OPT (NetMizer Only)”.</td>
</tr>
<tr>
<td>Op1</td>
<td>None</td>
<td>See page 132, “Op1 (NetMizer Only)”.</td>
</tr>
<tr>
<td>Op2</td>
<td>None</td>
<td>See page 133, “Op2 (NetMizer Only)”.</td>
</tr>
<tr>
<td>OP3</td>
<td>None</td>
<td>See page 134, “OP3 (NetMizer Only)”.</td>
</tr>
<tr>
<td>PASS</td>
<td>vtamappl [logonid[/password]] [data]</td>
<td>See page 135, “PASS”.</td>
</tr>
<tr>
<td>PASSP</td>
<td>None</td>
<td>See page 137, “PASSP”.</td>
</tr>
<tr>
<td>PERForm</td>
<td>[SYSID [applid</td>
<td>sysidnti] [TS</td>
</tr>
<tr>
<td>PEXEC</td>
<td>[command]</td>
<td>See page 141, “PEXEC/EXEC”.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameters</td>
<td>Page Reference</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------------------------------------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>PRINITID</td>
<td>[primary printer], [alternate printer]</td>
<td>See page 142, “PRINTID”.</td>
</tr>
<tr>
<td>PROC</td>
<td>{command_string}</td>
<td>See page 143, “PROC”.</td>
</tr>
<tr>
<td>PROGRAM</td>
<td>program_name [parameters]</td>
<td>See page 144, “PROGRAM/PGM”.</td>
</tr>
<tr>
<td>PROMpt</td>
<td>{parameters}</td>
<td>See page 145, “PROMpt”.</td>
</tr>
<tr>
<td>PROMPTN</td>
<td>{parameters}</td>
<td>See page 146, “PROMPTN”.</td>
</tr>
<tr>
<td>RBS</td>
<td>None</td>
<td>See page 147, “RBS (NetMizer Only)”.</td>
</tr>
<tr>
<td>RESEND</td>
<td>None</td>
<td>See page 148, “RESEND (NetMizer Only)”.</td>
</tr>
<tr>
<td>RESETUSR</td>
<td>{USERID=userid [.TERMINAL=nodename] [.APPLID=physical_name [.EXCMMSG=NOMSG]}</td>
<td>See page 149, “RESETUSR”.</td>
</tr>
<tr>
<td>REToff</td>
<td>None</td>
<td>See page 151, “RETOFF”</td>
</tr>
<tr>
<td>RETON</td>
<td>None</td>
<td>See page 152, “RETON”</td>
</tr>
<tr>
<td>RETURN</td>
<td>None</td>
<td>See page 153, “RETURN”</td>
</tr>
<tr>
<td>REXX</td>
<td>REXX [groupid.] progid args</td>
<td>See page 154, “REXX”.</td>
</tr>
<tr>
<td>RSTLOOP</td>
<td>None</td>
<td>See “RSTLOOP” on page 154.</td>
</tr>
<tr>
<td>RTRANID</td>
<td>tranid</td>
<td>See page 155, “RTRANID”.</td>
</tr>
<tr>
<td>RUN</td>
<td>None</td>
<td>See page 156, “RUN”.</td>
</tr>
<tr>
<td>SCRIPT</td>
<td>{AUTHOR plexec_command_string}</td>
<td>See page 157, “SCRIPT”.</td>
</tr>
<tr>
<td>SEND</td>
<td>See page 160, “SEND”.</td>
<td></td>
</tr>
<tr>
<td>SES</td>
<td>{session_command} [parameters]</td>
<td>See page 164, “SES”.</td>
</tr>
<tr>
<td>SIGNOFF</td>
<td>None</td>
<td>See “SIGNOFF” on page 165.</td>
</tr>
<tr>
<td>SM</td>
<td>{session_command} [parameters]</td>
<td>See “SM” on page 166.</td>
</tr>
<tr>
<td>STart</td>
<td>tranid [parameters]</td>
<td>See page 167, “STart”.</td>
</tr>
<tr>
<td>STORage</td>
<td>[CB=PIE_Control_block] [ADDRESS=XXXXX] [LENGTH=yyyyyyyy] [TRACE=OFF</td>
<td>ON]</td>
</tr>
</tbody>
</table>
### Chapter 3  Application Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameters</th>
<th>Page Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSId</td>
<td>{applid</td>
<td>sysidnt}</td>
</tr>
<tr>
<td>SYSTem</td>
<td>None</td>
<td>See page 171, “SYSTem”.</td>
</tr>
<tr>
<td>TERMDIR</td>
<td>None</td>
<td>See page 172, “TERMDIR”.</td>
</tr>
<tr>
<td>TN</td>
<td>None</td>
<td>See page 173, “TN (NetGate Only)”.</td>
</tr>
<tr>
<td>TP</td>
<td>None</td>
<td>See page 174, “TP”.</td>
</tr>
<tr>
<td>TR</td>
<td>None</td>
<td>See page 175, “TR”.</td>
</tr>
<tr>
<td>TRACE</td>
<td>TRACE {ON</td>
<td>OFF} ({TYPE=CMD|WTO|USER}) ([,\USERID=userid]), ([,\TERMINAL=nodename]) ([,\EXCM] ([,\NOMSG])</td>
</tr>
<tr>
<td>TRAN</td>
<td>transid [parameters]</td>
<td>See page 178, “TRANsaction”.</td>
</tr>
<tr>
<td>TS</td>
<td>None</td>
<td>See page 179, “TS”.</td>
</tr>
<tr>
<td>TSQSub</td>
<td>None</td>
<td>See page 180, “TSQSub (MultiCICS Only)”.</td>
</tr>
<tr>
<td>TSWITCH</td>
<td>vtamappl {logonid[/password]} [data]</td>
<td>See page 181, “TSWITCH”.</td>
</tr>
<tr>
<td>UCT</td>
<td>None</td>
<td>See page 183, “UCT”.</td>
</tr>
<tr>
<td>userdata</td>
<td>data</td>
<td>See page 184, “USERDATA, USERDATA2, USERDATA3, USERDATA4, and USERDATA5”.</td>
</tr>
<tr>
<td>USERDIR</td>
<td>None</td>
<td>See page 185, “USERDIR”.</td>
</tr>
</tbody>
</table>
**ACCesS (NetGate Only)**

ACCesS starts a NetGate session and logs on to a VTAM application. If the PIE/CICS Network Monitor is active, virtual terminal and logmode information can be retrieved from the PNAM file. If the Network Monitor is not available, the APPLID, LOGMODE and VTERM or VPOOL parameters must be specified with the ACCesS command.

**Format**

```
ACCesS applid [VTERM=virtual_terminal]
    [VPOOL=virtual_terminal_prefix] [LOGMODE=logmode]
    [DATA=[userid[/old_password/password]|data]
```

**Parameters**

- **applid**: Logical name of the target application. If the Network Monitor is used, `applid` is the logical system name. Otherwise, it is the APPLID of the target VTAM application.
- **VTERM**: Virtual terminal name. If the Network Monitor is not used, either VTERM or VPOOL must be specified. Mutually exclusive with VPOOL.
- **VPOOL**: Virtual terminal pool name. If the Network Monitor is not used, either VTERM or VPOOL must be specified. Mutually exclusive with VTERM.
- **LOGMODE**: VTAM logmode used to access the VTAM application. The Network Monitor provides the logmode valued defined by the PIE administrator.
- **DATA**: Initial data passed to the application as part of the VTAM log on message. DATA must be specified as the last parameter of the ACCesS command. The `LOGMSG=YES` parameter must be specified in the SIT to enable data to be passed by the ACCesS command.

User ID and password can be sent by the DATA parameter to automate logging on to the application active in the session.

If the user changes their password on the PIE Logon screen, the old password can be sent also to allow an automated sign on to the target application.
Examples

- **PEXE ACCesS CICS1**
  Region CICS1 is accessed from the CICS blank screen.

- **ACC CICS1 DATA=USR1/PSSWRD**
  If the target application accepts the VTAM Logon message, the user ID and password can be passed with the DATA parameter to automate the log on procedure. The user ID and password are separated with a forward slash (/) without blanks.

- **ACCS CICS1 VPOOL=PIE1 LOGMODE=M3278X**
  An active session named PROFILE is opened. The virtual terminal and logmode must be specified with the ACCess command if the NetWork monitor is not used with NetGate. Also, the current virtual terminal and logmode table definitions specified in the PNAM file can be overridden by including the VPOOL and LOGMODE parameters with ACCess.

- **ACCESS CICSPROD DATA=&ZUSER/&ZOPSWD/&ZPSWD**
  CICSPROD region is accessed after the user changed their current password during the sign on process. The user ID is passed with the &ZUSER variable and the old and new passwords are passed with the &ZOPSWD and &ZPSWD variables respectively.

- **ACCESS CICSPROD DATA=&ZUSER/&ZEPSWD/&ZPSWD**
  CICSPROD region is accessed after the user changed their current password during the sign on process. The user ID is passed with the &ZUSER variable and the old password is encrypted with the &ZEPSWD variable. In this case, the target application must be either PIE/CICS or PIE/TSO.

- **ACCS CICS1 DATA=&ZUSER/&ZPSWD CEMT I TAS**
  CICS1 region is accessed and the CEMT I TAS transaction is executed after log on is complete. The previous session becomes active again after exiting from the session started with the ACCesS command.
AutoEnd

AutoEnd returns control to the preceding session after an application running in the current session ends, or a transaction is completed. If there are pending stacked commands, they are executed after ending the current session.

AutoEnd suppresses the final send from the ending application and the user is presented with a PIE/CICS menu rather than a CICS blank screen. If there are no more active sessions, the final PIE command is executed instead.

AutoEnd works for conversational and pseudo-conversational transactions. There are some restrictions. AutoEnd does not work under the following conditions:

- AutoEnd works for pseudo-converse transactions that use EXEC CICS RETURN TRANSID(). It does not work with pseudo-converse transactions that retrieve the next tranid from the screen or screen buffer. PIE/CICS cannot detect the next tranid and will interrupt the application.
- System mode is active.
- The terminal is in Basic Mapping Support (BMS) paging mode.
- An ATI started task is pending.

AutoEnd is invoked automatically for all PIE/CICS processing: all menus, utilities, and commands. Other applications require an explicit AE to suppress messages and the CICS blank screen after ending an application or transaction.

AutoEnd is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

```
AutoEnd
```

Parameters

None

Example

- AE CMD CEMT I TAS

AE must be coded before CMD, TRAN, or the transaction ID in the command string to assure that control is returned to the preceding session after the transaction is complete.
Chapter 3  Application Commands

**ATI**

ATI enables automatic transaction initiate processing at the terminal for the current session. This allows a CICS started task to run.

ATI is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
ATI
```

**Parameters**

None

**Example**

- ATI AE CMD CEMT I TAS
ATTach

ATTach re-enables the PIE ATTACHX exit in the current session that had been previously disabled with the NOATTCH command. By default, the ATTACHX exit is active to intercept and modify transactions.

ATTach is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

| ATTach |

**Parameters**

None

**Example**

- `ATTach CMD CEMT I TAS`

  The CEMT transaction is intercepted by the ATTACHX exit to give it an automatic end attribute.
BROWSE

BROWSE starts the PIE/CICS Text Utility to view a text member stored in the PIE Repository or the contents of a CICS temporary storage queue (TSQ). Although a text member or TSQ can be viewed, they cannot be modified. Use the EDIT command instead to modify a member or temporary storage queue.

An error message appears on the current screen if an invalid text member or temporary storage queue name is specified with the BROWSE command. Pressing ENTER restores the screen and erases the error message. Refer to “BROWSEN” on page 95 for an alternative command to browse text members without producing error messages.

Format

```
BROWSE [groupid.]textname
```

Parameters

- **groupid**
  - Name of the group the text member or storage queue belongs.
  - If groupid is not specified, a standard search strategy is used to select the group for the text member or TSQ. The search is by userid as groupid, then the user's groupid. SYSTEM is assigned as the default group name if it cannot be found by userid or groupid.
  - For security purposes, the groupid of a CICS Temporary Storage queue is PIE. For example, BROWSE PIE.PIEZABCD browses a temporary storage queue called PIEZABCD.

- **textname**
  - Name of the text member or temporary storage queue to be browsed.
  - An error message appears if the textname parameter specifies an invalid text member or temporary storage queue name.

Examples

- **BROWSE USRPROF**
  - A text member named SYSTEM.USRPROF is available for viewing. SYSTEM is assigned as the default group name of the text member because the groupid parameter was not specified as part of the BROWSE command.

- **BROWSE USR1.PIECSTM**
  - The contents of a text member named USR1.PIECSTM are displayed from a PIE/CICS browse session.

- **BROWSE PIE.ZZZW**
  - The contents of a TSQ named PIEs.ZZZW are displayed from a PIE/CICS browse session.
BROWSE

BROWSE starts the PIE/CICS Text Utility to conditionally browse a text member stored in the PIE Repository, or the contents of a CICS temporary storage queue (TSQ). Although the a text member or temporary storage queue can be viewed, they cannot be modified. Use the EDIT command instead to modify a member or temporary storage queue.

Unlike the BROWSE command, BROWSE does not produce an error message if an invalid text member or temporary storage queue name is specified with the command. If an invalid name is used, no browse is attempted and the END command executes to restore the previous menu.

Format

```
BROWSE [groupid.]textname
```

Parameters:

- **groupid**: Name of the group the text member or storage queue belongs.
  - If groupid is not specified, a standard search strategy is used to select the group for the text member or storage queue. The search is by userid as groupid, then the user's groupid. SYSTEM is assigned as the default group name if it cannot be found by userid or groupid.
  - For security purposes, the groupid of a CICS Temporary Storage queue is PIE. For example, BROWSE PIE.PIEZABCD browses a temporary storage queue called PIEZABCD.

- **textname**: Name of the text member or temporary storage queue to be browsed.
  - An error message appears if the textname parameter specifies an invalid text member or temporary storage queue name.

Examples

- **BROWSE USRPROF**
  - A text member named SYSTEM.USRPROF is available for viewing. SYSTEM is assigned as the default group name of the text member because the groupid parameter was not specified as part of the BROWSE command.

- **BROWSE USR1.PIECSTM**
  - The contents of a text member named USR1.PIECSTM are displayed from a PIE/CICS browse session.

- **BROWSE SYSTEM.TSONEWS;ACCESS TSO**
  - The SYSTEM .TSONEWS text member is browsed if it exists and contains data. Otherwise, TSO is accessed immediately.
ConfirmEnd

ConfirmEnd returns control to the preceding level or session after an application running in the current session ends or a transaction is completed. Unlike AutoEnd, ConfirmEnd requires the user to press an AID key before returning to the preceding menu. This gives users the opportunity to read messages that may appear as a result of recently completed work.

CEnd is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
CEnd
```

**Parameters**

None

**Example**

- `CEnd CMD PYRL EJONES`

  The user must press an AID key to return to the current menu after the transaction is completed.
CICPRT

CICPRT passes the current defined CICS print key of the originating region to an application running under NetGate. When the print key is pressed, CICPRT allows a print request to be intercepted by the originating CICS region running within a NetGate session to perform a screen print.

CICPRT is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

```
CICPRT
```

Parameters

None

Example

```
• CICPRT ACC APPLID=CICSPROD
```

A NetGate session is started with the CICSPROD region as the active VTAM application running within it. If the CICS print key is pressed, the current CICS region intercepts the command and performs a screen print.
CLRRst

CLRRst enables the CLEAR key to restore the primary size of the current screen. By default, the CLEAR key can restore the default size of a screen that have been set to an alternative size. CLRRst restores the screen size function of the CLEAR key after it has been disabled with the NOCLRRST command. Refer to "NOCLRRST" on page 121 for information about disabling the screen size function of the CLEAR key.

CLRRst is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
CLRRst
```

**Parameters**

None

**Example**

- CLRRst AE CMD MSAX

  The MSAX transaction is executed. The CLEAR key can restore the default screen size defined for the session.
CMD

CMD starts a new transaction through transaction XCTL.

Transaction parameters are passed to the Variable Substitution exit to substitute PIE/CICS variables with run-time variables. After that, transaction parameters are passed to the Command Authorization exit, which can reject or accept this command. The Command Authorization exit may modify the parameters even further. If the command is accepted, the transaction is given control via transaction XCTL.

Format

```
CMD transid [parameters]
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tranid</td>
<td>Transaction ID.</td>
</tr>
<tr>
<td>parameters</td>
<td>Parameters specified after the transid in the command string are regarded as TIOA transaction parameters. The parameters are passed to the Variable Substitution exit. On return from this exit, PIE/CICS variables are substituted with run-time values.</td>
</tr>
</tbody>
</table>

Example

- AE CMD CEMT I TAS

The CEMT transaction is given control through transaction XCTL after the parameters have been converted to run-time variables by the Variable Substitution exit.
Dsr

Dsr enables NetMizer error recovery handling for output data sent from the current session. If output data contains an error, a popup window appears with a description of the error. This is the default NetMizer error recovery method.

Dsr is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

| Dsr |

**Parameters**

None

**Example**

- DSR OPT SYSID CICSPRD1 AE CMD ACNT JJONES

The ACNT transaction is sent to the CICSPRD1 by the PIE/CICS Dynamic Transaction Routing facility. The output data stream is fully optimized and has data stream error recovery handling enabled.
EC

EC executes a session command within a PIE/CICS application session. EC permits users to manage their PIE/CICS sessions from an application without being required to return to the Sessions menu.

If EC is entered without a command, a PIE pop-up window appears that lists common session commands that can be selected with PF keys.

Commands and parameters are passed to PIE/CICS Session Manager. The command is executed if it is valid. If the command is invalid, a PIE window appears with a list of common environment commands. Each command is associated with a PF or PA key. Pressing the appropriate key executes the command.

EC is an alias of the following PIE/CICS commands:
- ENV
- SES
- SM

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

```
EC [session_command] [parameters]
```

Parameters

- **session_command**: PIE/CICS environment command executed from the Sessions menu.
- **parameters**: PIE/CICS environment command parameters.

Example

- EC
  
  A PIE pop-up window appears with a list of common session commands that can be selected with PF keys.

- EC CLOSE PAYROLL
  
  The session named PAYROLL is closed.

- EC BUL
  
  The bulletin board is displayed.
EDIT

EDIT starts a PIE/CICS Text Utility session to modify a text member stored in the PIE Repository. EDIT creates a new, empty member if there is no member by that name stored in the PIE Repository. An error message appears in a popup window if EDIT is entered without parameters.

CICS Temporary Storage queues also can be edited. For security, temporary storage queues have a PIE groupid and the queue name has a PIE prefix. For example, EDIT PIE.PIEZABCD edits a temporary storage queue named PIEZABCD.

⚠️ WARNING

Editing a Temporary Storage queue changes the record length to 75 bytes. Lines greater than 75 bytes are truncated and lines less than 75 bytes are padded with blanks to the 75 byte record length.

The PIE /CICS Text Utility started by EDIT is similar to the ISPF EDIT facility. Many of the same commands are supported. Refer to the PIE/CICS Operation and Administration Guide for a description of all Text Utility commands.

Format

```
EDIT [groupid.] textname
```

Parameters:

- **groupid**: Name of the group the text object belongs.
  If groupid is not specified, a standard object search strategy is used to select the group for the text member. The search is by userid as groupid, then the user’s groupid. SYSTEM is assigned as the default group name if it cannot be found by userid or groupid.

- **textname**: Name of the text member or temporary storage queue to be edited. An error message appears if the textname parameter is missing and a default member name is not specified.

Examples

- **EDIT PIE.PIEZZZS**
  The Text Utility is started to edit a CICS temporary storage queue named PIE.PIEZZZS.

- **EDIT CICSPROF**
  The Text Utility is started to edit a member named SYSTEM.CICSPROF. The SYSTEM groupid is assigned to the member name by default.
ENCP

ENCP enables an encrypted password to be passed as a parameter of several PIE/CICS commands. Typically, these commands transfer control of the current terminal to a remote region. Encrypted passwords provide security if the command string is transmitted over a public telecommunications system to connect to a remote application. A PIE environment must be established at the remote application to decrypt the password.

ENCP can be specified with the PERFORM command to pass an encrypted password as a parameter. If Dynamic Transaction Routing is used, ENCP can be used with the TS, TP, or TN access method commands.

ENCP is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
ENCP
```

**Parameters**

None

**Example**

- `ENCP TS SYSID CICSPRD1 CEMT I TAS`

  The current user’s encrypted password is sent as a parameter of the TSWITCH command to log on to the remote CICSPRD1 region and execute the CEMT transaction.
Chapter 3  Application Commands

ENV

ENV executes a session command within a PIE/CICS application session. ENV permits users to manage their PIE/CICS sessions from an application without being required to return to the Sessions menu.

If ENV is entered without a command, a PIE pop-up window appears that lists common session commands that can be selected with PF keys.

Commands and parameters are passed to PIE/CICS Session Manager. The command is executed if it is valid. If the command is invalid, a PIE window appears with a list of common environment commands. Each command is associated with a PF or PA key. Pressing the appropriate key executes the command.

ENV is an alias of the following PIE/CICS commands:

- EC
- SES
- SM

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

ENV [session_command] [parameters]

Parameters

- session_command: PIE/CICS environment command executed from the Sessions menu.
- parameters: PIE/CICS environment command parameters.

Example

- ENV
  A PIE pop-up window appears with a list of common session commands that can be selected with PF keys.
- ENV CLOSE PAYROLL
  The session named PAYROLL is closed.
- ENV BUL
  The bulletin board is displayed.
GETC

GETC retrieves data from a queued binary commarea previously stored by the PIEXEC API. Data from the commarea is passed to the next program or transaction scheduled for execution.

Format

GETC

Parameters

None

Example

- GETC
  
  Data from the binary commarea is retrieved and passed to a pending program or transaction scheduled for execution.
GOTO

GOTO executes an option listed on the current menu. Options are selected by their menu number or name as they appear in the # and Name fields. Immediate control is given to the option and all pending commands are discarded.

If the current menu does not list the option, control reverts to the immediately preceding menu and the option executes if it is listed. This backward search process continues to each preceding menu until the option is found or the highest level menu is reached in the current session. A popup window appears with an error message if the search fails to find the menu option.

Format

GOTO [nn|name]

Parameters

nn Two-character option number listed on a menu
name 1 to 8-character name of menu option listed on the Name field of the current menu.

The name parameter is processed as a PIE primary menu command. Therefore, name can be a menu command such as END or DOWN.

Examples

- GOTO 5
  The function associated with option 5 of the current PIE/CICS menu is invoked.
- GOTO MARKTNG
  The MRKTNG option is selected from the current menu.
- GOTO DOWN
  DOWN is invoked from the current menu as a primary menu command to scroll forward.
HELPDESK

HELPDESK grants access to view a user's screen by authorizing the helpdesk ID or mask. This gives authorized personnel with the helpdesk ID the capability to view the requesting user's current screen with the PIE CICS VIEW Capture facility.

The userid specified with HELPDESK replaces the user's current helpdesk ID specified in the Generic User Directory. This change affects all sessions.

The helpdesk ID may be specified with the PIE &ZHDESK variable.

**Format**

```
HELPDESK [userid]
```

**Parameters**

- **userid**
  - Helpdesk user ID of the user whose current screen can be viewed.
  - If the userid parameter is not specified, the user's current helpdesk ID is reset or cleared to blank in the Generic User Directory.

**Examples**

- **HELPDESK HLP1**
  - The user with the ID of HLP1 is granted access to view the requesting user's current screen.

- **HELPDESK SYSPRG***
  - System programmers logged on with a SYSPRGn userid have the capability to view the requesting user's current screen.
KEY

KEY assigns a 3270 key to the AID value of the next command. The default is the ENTER key if a parameter is not specified.

KEY is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user's current profile.

Format

```
KEY [3270_key]
```

Parameter:

- **3270_key**
  - Name of the 3270_key assigned to the AID value of a pending command. Possible 3270_keys with AID function include the following:
  - PF01-PF24 PF function keys
  - PA01-PA03 PA function keys
  - ENTER Enter key
  - CLEAR Clear key
  - OPID 3270 operator ID card reader
  - MAGNT Magnetic slot reader/hand scanner
  - SELEC Selector Pen
  - NOAID no AID (display station)
  - NOPRT no AID (printer)
  - TSTRQ Test Request/System Request
  - ASIS Leave AID byte as is and pass through to the transaction.

Example

- KEY PF05 AE CMD ACNT
  
The PF05 key serves as the AID for the ACNT transaction.
LAUNCH (Dynamic Menus Only)

LAUNCH starts an application listed as a menu option. The application starts in a free session or within a PIE/REXX program. LAUNCH enables users to run a single session for a Dynamic Menu and the remaining applications within special sessions.

LAUNCH effectively emulates the PERFORM command with the following parameters:

```
PERFORM OPTION=(NEW,LOGNAME),T=xxxxxxxx
```

where xxxxxxxx is the execution string following the LAUNCH command.

Typically, LAUNCH is entered from Dynamic Menu’s Menu Line Definition screen. This screen is used to associate commands and transactions with options shown on a Dynamic Menus screen.

```
Execute ===> LAUNCH CMD CEMT
Type    ===> F
```

As the command is executed, a search is made for a session with the same name as the menu option application. If the requested application is currently running, it becomes the active session. If the application is not running, a new, free session is started to run the application. The new session is assigned the same name as the title of the menu option.

Format

```
LAUNCH tranid arguments
```

Parameter:

- **tranid**: ID of the transaction being executed.
- **arguments**: Transaction arguments.

Example

```
...#.. Name...     Title........................Arguments
1   MASTER      Master Terminal Commands
```

```
Execute ===> LAUNCH CMD CEMT
Type    ===> F
```

A Dynamic Menus screen displays MASTER as a menu line option that users can select. The command `LAUNCH CMD CEMT` has been previously specified as the command associated with MASTER menu option.

If a PIE session titled MASTER exists, then LAUNCH switches to the session and executes the CEMT transaction. If the MASTER session does not exist, then a new, free session is created to execute the CEMT transaction. The new session is titled Master Terminal Commands from the Title field of the Dynamic Menus screen of the selected option.
LIST displays selected PIE utility lists. A PIE utility list displays the current status of various aspects of a PIE/CICS system.

The following table shows the different types of PIE utility lists that can be displayed with the LIST command. The Profile utility list is the default if the type parameter is not specified with LIST.

<table>
<thead>
<tr>
<th>List Type</th>
<th>Selection Criteria Based On Menu Fields</th>
<th>Default Selection Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applids or Systems</td>
<td>isystem psystem type status am users maxusr actusr maxact trancount</td>
<td>All</td>
</tr>
<tr>
<td>Balancing Groups</td>
<td>groupid isystem psystem method status users maxusr objusr resp objresp cpu objcpu rate</td>
<td>All</td>
</tr>
<tr>
<td>Menus</td>
<td>group name change-date-time change-by size title admin</td>
<td>Authorized Only</td>
</tr>
<tr>
<td>NetGate Sessions</td>
<td>vterm netname userid name plu logmode sess nsb-addr status</td>
<td>All</td>
</tr>
<tr>
<td>Panels</td>
<td>group name change-date-time change-by size title admin</td>
<td>Authorized Only</td>
</tr>
<tr>
<td>Profiles</td>
<td>group name change-date-time change-by size title admin</td>
<td>Authorized Only</td>
</tr>
<tr>
<td>Scripts</td>
<td>group name change-date-time change-by size title admin</td>
<td>Authorized Only</td>
</tr>
<tr>
<td>Protected Terminals</td>
<td>terminal userid date time</td>
<td>All</td>
</tr>
<tr>
<td>Text</td>
<td>group name change-date-time change-by size title admin</td>
<td>Authorized Only</td>
</tr>
<tr>
<td>Users</td>
<td>userid term nodeid applic type tran/pnam last-input ltran status system groupid usertrace wto-trace cmd-trace unique-id</td>
<td>All</td>
</tr>
</tbody>
</table>

Command parameters restrict the displayed list to specific data that matches the selection criteria. LIST parameters are based upon the menu fields of each utility list. Entering a data element that is unique for a menu field restricts the displayed list to those items that match the selection criteria.

**Format**

```
LIST [Profiles | type] [ALL | selection criteria]
```

Parameters:

- **Type**
  - **Profiles** List of PIE/CICS user profiles. The default.
  - **Menus** List of Dynamic Menus
  - **TEXt** List of text members including news files, help members, and REXX programs.
  - **Applids or Systems** List of PIE/CICS user profiles. The default.
  - **Balancing Groups** List of Dynamic Menus
  - **Menus** List of Dynamic Menus
  - **NetGate Sessions** List of text members including news files, help members, and REXX programs.
  - **Panels** List of Dynamic Menus
  - **Profiles** List of Dynamic Menus
  - **Scripts** List of Dynamic Menus
  - **Protected Terminals** List of Dynamic Menus
  - **Text** List of Dynamic Menus
  - **Users** List of Dynamic Menus

---

110  ▼  PIE/CICS 3.4.0
LIST

Panels List of Logon Director screens
Users List of PIE users. Users is an alias of Applid.
Applid List of APPLIDs. Applid is an alias of Users.
SYSid List of SYSDIDs.
TERminals List of protected terminals
Balance List of load balancing groups
Netgate List of sessions running under NetGate
SCripts List of play-back script files (Not REXX scripts)

Selection criteria
Masks restrict displayed data from a PIE utility list to selected items that match the selection criteria. The selection criteria are based upon unique data elements of each List Utility menu field.
Selection criteria are positional. If more than one criterion is specified with LIST, they must follow the same order as the menu fields appear on the Utility List. The table on page 110 shows the selection criteria and their order for each utility list.

Examples

- **LIST users abc1**

  Session parameters for user ABC1 are shown from the PIE User Utility list.

- **LIST sysid tso**

  TSO session parameters are shown from the PIE SYSID Utility list.

- **LIST m pie piem**

  Menus beginning with the PIE group name and PIEM prefix are listed on the Menus Utility list.
Chapter 3  Application Commands

Load

LOAD reads or writes data between pre-defined files and the PIE Repository database. LOAD allows PIE objects to be transferred from one Repository to another. Also, it can be used to load a text file created in another environment (like ISPF).

The load function sends data from an input file to the PIE Repository. Messages are written to a PIE text member that give the status of the load job. The member can be browsed to read the messages produced by the load job.

The unload function sends a copy of data from the Repository to an output file. The file can be a sequential data set defined in the CICS DCT or a CICS temporary storage queue. If a transient data queue is used, it can be a PDS member or a sequential data set. Messages are written to a PIE text member containing the unload control statements.

Refer to “Repository Load Utility” on page 77 of the Operation and Administration Guide for instructions to prepare input and output files.

Format

```
LOAD [function] [queue] [group] [member]
```

Parameters

- **function**  Function performed by the LOAD command:
  - **Load**  Load a PIE object from the Load input file to the PIE Repository
  - **Unload**  Unload a copy of an object stored in the PIE Repository to the Unload Output file

- **queue**  Input transient data queue. If the queue name is four characters long, the queue is assumed to be a transient data queue. If the queue name is five or more characters, it is assumed to be a temporary storage queue. The default value is PIEI for the load (input) function and PIEO for the unload (output) function.

- **group**  Group ID of the control file. For the unload function, it must contain the unload control statements. The Load Utility writes load or unload messages to this file. The default value is PIEUTIL.

- **member**  Member name of the control file. The default value is UNLOAD.

Example

- **LOAD**

  LOAD without parameters invokes the Repository Load utility. Enter data on the appropriate screen fields to select a load or unload operation.

- **LOAD L PIEI PIEUTIL.LOAD**

  The TD queue name is PIEI and the control file name is PIEUTIL.LOAD.
LOGMode

LOGMode specifies the terminal logmode to use for remote transaction processing by a CLSDST PASS operation with either the TSWITCH or PASS commands.

Format

<table>
<thead>
<tr>
<th>LOGMode</th>
<th>SNA=\text{logmode},</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NONSNA=\text{logmode},</td>
</tr>
<tr>
<td></td>
<td>TERMINAL=\text{nodename},</td>
</tr>
<tr>
<td></td>
<td>EXCMMSG</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LOGMODE</th>
<th>RESET,</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TERMINAL=\text{nodename},</td>
</tr>
<tr>
<td></td>
<td>EXCMMSG</td>
</tr>
</tbody>
</table>

| LOGMODE | INQ,TERMINAL=\text{nodename} |

Parameters

- **SNA**: SNA logmode name.
- **NONSNA**: Non-SNA logmode name.
- **TERMINAL**: VTAM nodename of the terminal.
- **EXCMMSG**: Exclude all messages except error messages.
- **RESET**: Reset the current logmode (clear it).
- **INQ**: Current logmode status.

**PIE EXEC ACTION:**

Changes or displays the terminal logmode that is used for CLSDST PASS. This change is global and affects all sessions. For RESET, clear the logmode of the current terminal if TERMINAL is not specified. Otherwise clear the logmode for the VTAM nodename specified by TERMINAL.

For INQ, display the logmode of the current terminal if TERMINAL is not specified. Otherwise display the logmode for the VTAM nodename specified by TERMINAL. If RESET or INQ is not specified, changed the logmode. If TERMINAL is not specified, change the current terminal’s logmode. Otherwise change the logmode for the VTAM nodename specified by TERMINAL.

Both NONSNA and SNA may be specified, and a determination will be made as to the nature of the terminal in question, and the correct logmode selected.

**NOTES:**

The logmode may be specified via the PIE variable &ZLOGMOD. This command executes as transaction PLOG.
Chapter 3  Application Commands

**MENU**

MENU displays a PIE/CICS menu based upon the name of the menu entered as a command parameter.

Menus are stored in the PIE Repository database. Each menu has a unique name consisting of a group name and the menu name. A complete list of PIE/CICS menus can be displayed with the LIST MENU utility.

**Format**

```
MENU [main] [[groupid].[menuname]]
```

**Parameters**

- **main**: Designates a main menu that becomes active after exiting from a subordinate menu with the RETURN command.
- **groupid**: Specifies the group to which the menu belongs. If not specified, the standard object search strategy will apply—i.e., first search for the menu with userid as groupid, next use the user's groupid, then lastly use SYSTEM as groupid.

  Groupid must be accompanied by menuname. If only one parameter is supplied, it is regarded as the menuname parameter. If no parameter provided, the default menu from the User Directory is used.

- **menuname**: Name of the menu to be displayed. If omitted, the default menu from the User Directory is used.

**Examples**

- **MENU PIEMADM**

  The PIE Systems Administration menu is displayed. SYSTEM is the groupid assigned to the menu name.

- **MENU DEMO.PRODUCTN**

  The PRODUCTN menu that belongs to the DEMO group is displayed.
MSG

MSG reestablishes the delivery of PIE/CICS SuperMessages to the current session. By default, PIE/CICS sessions can receive SuperMessages. A session must have previously disabled the receipt of SuperMessages with the NOMSG command. See “NoMSG” on page 124.

MSG is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
MSG
```

**Parameters**

None

**Example**

- MSG AE CMD ACCNTNG

  The ACCNTNG session is reopened and SuperMessages can be received that had been previously disabled.
NOAE

NOAE disables automatic end processing in the current session. Control does not revert to a PIE/CICS menu after a transaction is completed.

Auto end should be suppressed for the following conditions:

- The transaction is still pseudo-conversing with EXEC CICS RETURN TRANSID().
- System mode is active.
- The terminal is in Basic Mapping Service paging mode.
- An ATI started task is pending

NOAE is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

| NOAE |

**Parameters**

None

**Example**

- NOAE CMD PQ01 INVNTRY
  
  Auto end is disabled for a non-PIE/CICS transaction.
**NoATI**

NoATI prevents ATI (EXEC CICS STARTed) tasks from running in the current session. NoATI is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

<table>
<thead>
<tr>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoATI</td>
</tr>
</tbody>
</table>

**Parameters**

None

**Example**

- `NoATI AE CMD WCAN`

  Transaction WCAN cannot be interrupted by an ATI started task.
NoATTach

The ATTACHX exit intercepts transactions and a different transaction or command can be substituted instead. NoATTACH disables the PIE ATTACHX exit. If NoATTACH is specified, then transactions complete without being modified.

NoATTach is useful for transactions that are normally intercepted and re-invoked with additional PIE Exec attributes. NoATTach allows the transaction to complete without undergoing routine modification by the ATTACHX exit.

NoATTach is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user's current profile.

Format

```
NoATTach
```

Parameters

None

Example

- `NOATTACH AE CMD CESF`

  This command displays the CICS sign on panel. In a typical PIE/CICS environment, the CESF transaction is intercepted by ATTACHX and the current session is ended instead.
NOCE

NOCE disables confirm end processing for transactions that complete in the current session.

NOCE is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user's current profile.

**Format**

```
NOCE
```

**Parameters**

None

**Example**

- NOCE CMD ABCD
  
  The ABCD transaction ends without a confirm end.
NOCICPRT

NOCICPRT passes the current defined CICS print key to an application running under NetGate. NOCICPRT should be executed before starting a new NetGate session with the ACCESS command.

NOCICPRT is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

NOCICPRT

**Parameters**

None

**Example**

- NOCICPRT ACCESS APPLID=CICSPROD

A NetGate session is started with the CICSPROD region as the active VTAM application running within it. If the user presses the CICS print key in this session, it is passed through to the CICSPROD region's
NOCLRRST

NOCLRRST disables the CLEAR key from restoring the primary screen size in the current session. By default, the CLEAR key restores the primary screen size.

NOCLRRST is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

NOCLRRST

Parameters

None

Example

- NOCLrrst
**NODSR**

NODSR disables NetMizer’s data stream error recovery handling in the current session. Outbound data streams are not examined for errors.

NODSR is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
NODSR
```

**Parameters**

None

**Example**

- NODSR ACCESS APL=TSO
  
  TSO is opened within a NetGate session and NetMizer error recovery handling is turned off for outbound data streams.
NoENcP

NoENcP prevents encrypted passwords from being passed as a parameter of several PIE/CICS commands. Typically, these commands transfer control of the current terminal to a remote region. In some instances, unsecured telecommunications pathways may be used to communicate with the remote region. An encrypted password provides a secure method to automate the log on process at the remote region. The region must run in a PIE environment with the capability to decrypt the password.

NoENcP is intended for sites that use remote regions that do not run as a PIE/CICS application. In that case, an unencrypted user password must be sent as a parameter with the &PSWD variable. The encrypted password variable, &ZEPSWD, cannot be used with commands that transfer control to remote regions that do not run as a PIE/CICS application.

NoENcP can be specified with the PERFORM command to pass an unencrypted password as a parameter. If Dynamic Transaction Routing is used, NoENcP can be used with the TS, TP, or TN access method commands.

NoENcP is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

<table>
<thead>
<tr>
<th>Parameters</th>
</tr>
</thead>
</table>

None

Example

- NOENCP SYS1D CKSTOR2 PERF TS CEMT

Data is sent in unencrypted form to the CKSTOR2 application when a session switch is made with the TSWITCH access method.
NoMSG

NoMSG prevents PIE/CICS SuperMessages from being delivered to the current session. NoMSG is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

| NoMSG |

**Parameters**

None

**Example**

- NOMSG CMD CEMT I TERM

  The session running the CEMT transaction will not be interrupted by a SuperMessage.
NON3270

NON3270 enables non-3270 or transparent data streaming in the current session. None of NetMizer’s data compressions techniques are used to optimize a session’s input and output data streams.

PIE/CICS maintains the screen image of every user in memory to support the following features:

- Session Manager switching
- SuperMessages
- NetMizer datastream optimization
- Print command
- Global PIE keys
- PIE scripts
- Locked terminals
- Session recovery methods

If none of these features are used in the current session, NON3270 is the recommended PIE/CICS performance mode. A significant amount of memory can be conserved and CPU usage can be reduced by running PIE/CICS in non-3270 mode.

NON3270 is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

| NON3270 |

Parameters

None

Example

- NON3270 AE CMD IND$

  NetMizer optimization should be temporarily disabled to execute the $IND (file transfer) transaction. Transparent data streaming is used to send the file without any NetMizer optimization.
NOoPT (NetMizer Only)

NOoPT disables NetMizer data stream optimization in the current session. None of NetMizer’s compression techniques are used to reduce the length of input and output data streams.

NOoPT is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

| NOoPT |

Parameters

None

Example

- PEXE NOOPT CEDF

   The CEDF transaction is executed from a CICS blank screen without NetMizer optimization.
NoPaSsP

NoPaSsP prevents a PIE profile from being passed to another application as a parameter of the PERForm command to set conditions for CICS remote transaction processing. If Dynamic Transaction Routing is used, NoPaSsP can be used with the TS, TP, or TN access method commands. Only the userid and password can be sent as command parameters.

NoPaSsP is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

| NoPaSsP |

Parameters

None

Example

- NOPASSP TS SYSID CICSPRD1 CEMT I TAS
  
The user’s profile is not sent as log on information when a session transfer is made to the remote CICSPRD1 region.
NoRBS (NetMizer Only)

NoRBS disables 3270 read buffer suppression by NetMizer in the current session. A PIE/CICS terminal sends a copy of the entire buffer in response to applications issuing a READ BUFFER request. The NetMizer default is to suppress the READ BUFFER request and send a copy of the terminal buffer stored in memory instead.

NoRBS is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

NoRBS

Parameters

None

Example

- NORBS CMD WANX
  
  3270 read buffer suppression is disabled for the WANX command. The PIE/CICS terminal sends a copy of the entire buffer in response to a READ BUFFER request.
NOTSqsub (MultiCICS Only)

NOTSQSUB prevents the substitution of temporary storage queue names in the current session. The PIE/CICS method of creating a unique TSQ name for each session based upon user ID is not used. Standard TSQ naming convention based upon the user’s terminal ID is used instead.

NOTSqsub is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

NOTSqsub

Parameters

None

Example

• NOTSQSUB AE CMD CECI

  Temporary storage queues are created with names based on the user’s terminal ID when CECI is executed.
**NOUCt**

NOUCt prevents uppercase translation of terminal input data.

NOUCt is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
NOUCt
```

**Parameters**

None

**Example**

- `NOUCt SEND`

  The SuperMessage utility screen appears to prepare a message to be sent to PIE/CICS users. NOUCT prevents the text of the message from being translated to uppercase.
**OPT (NetMizer Only)**

OPT enables full NetMizer data stream optimization in the current session. OPT enables the full complement of NetMizer optimization techniques, which includes the following:

- Repetitive character optimization
- Imaging Optimization
- Inbound suppression optimization
- Read buffer suppression
- Semantic imaging

OPT is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
OPT
```

**Parameters**

None

**Example**

- `OPT ACCESS APPL=CICS2`

  Full NetMizer data stream optimization is enabled after opening the session that runs the CICS2 region.
Op1 (NetMizer Only)

OP1 enables NetMizer level-1 data stream optimization in the current session. Level-1 optimization is restricted to outbound optimization and normal imaging.

Op1 is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

Format

```plaintext
Op1
```

Parameters

None

Example

- OP1 CMD TA01

Level-1 NetMizer optimization is enabled for any data streams that result from the TA01 transaction.
OP2 (NetMizer Only)

Op2 enables NetMizer level-2 data stream optimization in the current session. Level-2 optimization is restricted to outbound optimization and semantic imaging.

Op2 is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
OP2
```

**Parameters**

None

**Example**

- `OP2 AE CMD WXYZ`

  Level-2 NetMizer optimization is enabled for any data streams that result from the `WXYZ` command.
**OP3 (NetMizer Only)**

Op3 enables NetMizer level-3 data stream optimization in the current session. Level-3 optimization includes inbound and outbound optimization techniques plus normal imaging.

Op3 is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the default application defined in the user’s current profile.

**Format**

```
Op3
```

**Parameters**

None

**Examples**

- **PEXE OP3 MENU ACCOUNT**
  
The MENU transaction is executed from the CICS blank screen with level-3 NetMizer optimization.

- **ALLOC OP3 MENU ACCOUNT**
  
A new session is allocated and the MENU transaction is executed with level-3 NetMizer optimization.
PASS

PASS executes a CLSDST PASS transaction to transfer control of the current terminal to another VTAM application. By default, the user is disconnected from the current CICS region when the terminal transfers control.

Data can be passed to the target application as command parameters that are interpreted by the Logon Director as the good morning transaction if the target region runs under PIE/CICS. The userid and password can be passed as &ZUSER and &ZPSWD variables respectively to automate the log on at the target application. Also, the user’s old password can be passed in unencrypted and encrypted forms with the &ZOPSWD and &ZEPSWD variables respectively.

Other data that can be passed as PASS parameters include the TSO PROC, initial CICS transaction, or PIE/CICS user profile.

Format

```
PASS vtamappl [logonid[/old_password/password]] [data]]
```

Parameters

- **vtamappl**: Remote VTAM application ID (APPLID).
- **logonid**: User log on ID at the remote system.
- **/old_password**: Old password if the user changed their password from the PIE Logon screen. The old password can be passed in unencrypted and encrypted form with the &ZOPSWD and &ZEPSWD variables respectively. If the old password is sent in encrypted form with the &ZEPSWD variable, the target application must be either PIE/CICS or PIE/TSO.
- **/password**: User password for the remote VTAM application. If specified, the password must follow the logonid and be separated with a forward slash (/).
- **data**: Optional parameter to specify an initial command or transaction that executes after log on is complete at the target application. Cannot be used without logonid.

Examples

- **PASS CICSPRD1**
  
  The terminal is transferred to the log on screen of CICSPRD1. The user is disconnected from the current region. After exiting from CICSPRD1, the user is returned to the VTAM log on screen.

- **PASS CICSPRD1 USR1/PSWRD**
  
  The terminal is transferred to CICSPRD1 and the user is disconnected from the current region. The user’s ID and password were passed as parameters, allowing PIE/CICS to execute the initial command defined in the default user profile of the CICSPRD1 region after log on is complete. After exiting from CICSPRD1, the user is returned to the VTAM log on screen.
PASS CICSPRD1 &ZUSER/&ZOPSWD/&ZPSWD

The terminal is transferred to CICSPRD1 and the user is disconnected from the current region. The user changed their password from the PIE Logon screen. The old unencrypted password is sent with the &ZOPSWD variable to automate the sign on at the target application despite the password change.

PASS CICSPRD1 USR1/PSWRD CEMT I TERM

The terminal is transferred to CICSPRD1 and the user is disconnected from the current region. The user’s ID and password were passed as parameters, allowing PIE/CICS to execute the CEMT transaction passed as a command parameter. After exiting from CICSPRD1, the user is returned to the VTAM log on screen.
PASSP

PASSP enables a PIE profile name to be passed as a parameter with the PERFORM command. Also, PASSP can be used with the Dynamic Transaction Routing access method commands, TS, TN, and TP. The PIE profile name is passed as the data parameter following the userid/password parameters. This affects the current session only.

PASSP is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

Format

```
PASSP
```

Parameters

None.

Example

- SYSSID=CICSPRD1 AE PASSP TN PERF OPT=(NEW,LOGN),T=CEMT
Chapter 3  Application Commands

PERForm

PERForm conditionally initiates an application or a transaction at the current or a remote region.

The following PIE state commands normally precede the PERFORM command to set the conditions propagated to the transaction invocation:

- AE
- CE
- ENCP
- PASSP
- NOATI
- NOUCT

Format

```plaintext
[SYSID {applid|sysidnt}] [TR|TP|TS|TN] PERForm
OPTION=[(opt1,...,optn)], TranId=transid_string
```

Parameters

- **SYSID**
  - **CICS SYSIDNT** or a VTAM APPLID of the target region.
  - **sysidnt** First four characters of the CICS SYSIDNT used to route remote transactions to CICS regions by transaction routing methods.
  - **applid** 1 to 8-character VTAM applid of the destination application that executes remote transactions.

- **TS TP TR TN** Remote access transaction routing method.
  - **TN** Terminal logs on to the remote region with the ACCEss command to process the transaction locally. See “TN (NetGate Only)” on page 173.
  - **TP** Terminal transfers to the remote region with the PASS command to process the transaction locally. See “TP” on page 174.
  - **TR** Remote transactions are processed with the PIE/CICS Dynamic Transaction Routing facility. The terminal remains logged on at the current application and the remote transactions are processed by standard CICS MRO/ISC methods. The default remote access transaction routing method. See “TR” on page 175.
  - **TS** Remote transactions are processed with the PIE/CICS terminal switch (TSWITCH) method. The current terminal is reaquired automatically after completing the transactions and logging off the target application.
If the TN, TP, or TS remote transaction access method commands are specified, SYSID should be specified also. The Transid string is invoked in the system specified by SYSID and are routed by the access method specified by TN, TP, or TS. If the current system is the target, then the Transid string is invoked by transaction XCTL.

If SYSID is not specified, the Transid string is invoked in the current system by transaction XCTL with PERFORM command options.

**OPTion**

Options to set the conditions for executing the application or transaction.

- **Sesmenu**
  - Sessions Menu is the invoking menu. When PERFORM is coded in a session configuration, Sesmenu should be used.

- **NEWses**
  - Transactions or applications are executed within a new PIE session. If a current session has the same name as the tranid, that session is resumed instead.

  The new session is temporary and is cancelled after completing the transaction or application.

- **Remote**
  - A check is made within the current CICS region to see if a transaction is defined as MRO remote to a remote region specified by the SYSID parameter. If the transaction supports MRO, it is invoked locally and executed in the remote region by transaction routing methods such as PASS or TSWITCH.

- **NOName**
  - The name of a new session specified by the NEWses parameter is the same as the transid.

  NOName creates an unnamed session. The session assumes the default name of the last tranid executed within the session.

- **LOGName**
  - The name of a new session specified by the NEWses parameter is the same as the current session’s logical name. This is the name shown in the Name of a Dialog Manager or Session Manager menu.

- **NOTitle**
  - The title of a session created by the NEWses parameter is suppressed. The new session does not inherit the current session’s title.

  NOTitle attributes a null title to the new session. A default session title is created from the execution string of the PERFORM command.

- **NOCheck**
  - The verification of a valid transid in the PCT is suppressed before creating a new local session.

- **Perm**
  - A permanent session is created by the NEWses parameter rather than a temporary session.
Chapter 3  Application Commands

<table>
<thead>
<tr>
<th>Test</th>
<th>A pop-up display appears with a listing of the PERFORM command string. This option is intended for debugging purposes and does not execute the command string.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Erase</td>
<td>The transaction is executed with the TRAN command rather than the default CMD. TRAN erases the screen before invoking the transaction.</td>
</tr>
<tr>
<td>Transid</td>
<td>Transaction ID or transaction string. This must be specified as the last PERFORM command parameter.</td>
</tr>
</tbody>
</table>

**Examples**

- **AE PERF OPT=(NEW,LOGN),T=CEMT**

  A new session is created, given the current logical name, and the CEMT transaction executed.
**PEXEC/EXEC**

PEXEC and EXEC function as place holders to invoke the PIE Exec command facility. PEXEC and EXEC are state commands that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

**Format**

```
PEXEC [commands]
```

**Parameters**

- **commands**: Commands that are executable by the PIE Exec command facility.

**Example**

- PEXEC AE CMD CEMT

  The CEMT transaction is executed from the CICS blank screen.
PRINTID

PRINTID assigns CICS terminal IDs to a session’s primary and alternate printers. The printer IDs are used by the CICS print screen facility, which can be invoked with the PIE Session Manager PRINT command.

This is a global change that affects all sessions. The TCTTE is updated with the CICS termids of the primary and/or alternate printers.

The primary printer ID may be identified with the &ZPRTPRI PIE variable and the alternate with the &ZPRTALT variable.

Format

| PRINTID [primary printer|alternate printer] |

Parameters

Both parameters of the PRINTID command are positional. Place a vertical bar ( | ) immediately before the alternate printer parameter as a place holder to specify an alternate printer.

- primary printer  CICS terminal ID or the netname of the primary printer. The primary printer is the default if only one terminal ID is specified with the PRINTID command.
- alternate printer  CICS terminal ID or the netname of the alternate printer.

Examples

- PRINTID LPTR|LP02
  The primary printer is assigned LPTR as its CICS terminal ID. The alternate printer is designated LP02.
- PRINTID |LP02
  The alternate printer is assigned LP02 as its CICS terminal ID.
PROC

PROC executes commands passed by the VTAM log on message. PIE Exec initiates the PROC transaction via transaction XCTL, which schedules commands extracted from the VTAM log on message into the FIFO command queue. The PIE Exec RUN command is immediately issued to process the commands held in the queue.

Commands passed as PROC parameters are executed if the VTAM logon message did not place any commands in the command queue. The END command is executed instead if there are no commands in the VTAM log on message and no parameters are supplied with PROC.

The parameters are passed to the Variable Substitution exit. On return from this exit, PIE/CICS variables are substituted with run-time values. The resultant parameters are passed to the Command Authorization exit, which can reject or accept this command. The Command Authorization exit may even modify the parameters further.

If the command is accepted, the transaction is given control via transaction XCTL. This command executes as transaction PROC.

Format

```
PROC [commands]
```

Parameters

- **commands**: Executable PIEXEC commands that are executed if there are no queued commands from the VTAM log on message.

Examples

- **PROC**
  
  Commands in the VTAM log on message are processed as Application commands.

- **PROC SM OPEN 2**
  
  Commands in the VTAM log on message are processed as Application commands. If no commands are found in the VTAM log on message, session 2 is opened instead.
PROGRAM/PGM

PROGRAM transfers control to a program through transaction XCTL and passes parameters in the commarea.

Format

```
[ PROGRAM
  XCTL ] programname [parameters]
```

Parameters

- **programname**: Name of the program that assumes control.
- **parameters**: Commarea data for the program.

Example

- **PROGRAM DFHGMM**
  
The CICS ‘good morning’ program is executed.
**PROMpt**

PROMpt blanks the user’s current screen and displays a message sent as a command parameter. Although the screen is cleared and transactions may be entered, the terminal is not in system mode. The user returns to the previous session where the PROMPT command was entered if there is no transid (for pseudo-conversation) and the data on screen is not a valid transaction.

**Format**

```
PROMpt [message]
```

**Parameters**

- **message** 1 to 79-character message displayed on the current user’s terminal.

**Example**

- `ae cmd cspk;ae promptn Screen Print Completed.;sm returna`
  The screen is cleared and the message appears on the screen after the current screen is printed. The message confirms the screen was printed.
PRoMptN

PRoMptN places the current terminal in a native CICS session without clearing the screen. Though you can clear the screen and enter other transactions, you are not in system mode. If you press ENTER when there is no transid (for pseudo-conversation), and the data on screen is not a valid transaction, you will returned to where you issued the PROMPTN command.

Format

 PRoMptN (parameters)

Parameters

if present, are displayed.

Example

- ce promptn
  
  If this is the initial PIE command, log on messages remain on the screen until the user presses ENTER because of the confirm end state command.
RBS (NetMizer Only)

RBS specifies that NetMizer perform 3270 read buffer suppression for any read buffer request issued by an application in the current session. NetMizer intercepts the READ BUFFER request and sends the application a copy of the terminal buffer maintained in memory. This prevents a terminal I/O.

RBS is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

Format

| RBS |

Parameters

None.

Example

- RBS CMD ACTN
  
  Read buffer requests issued by transaction ACTN are suppressed to reduce network traffic. The application still receives the contents of the buffer from the PIE Screen Manager.
ReSeND (NetMizer Only)

RESEND suppresses NetMizer delta optimization for the next send operation in the current session. Rather than sending only the difference (delta) between previous and current versions of data, RESEND forces all of the data to be sent. Typically, RESEND is used after a problem occurs that may be attributable to NetMizer delta datastream optimization techniques. RESEND allows the user to send data in an unoptimized form to verify the source of the problem.

RESEND is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user's current profile.

**Format**

| ReSeND |

**Parameters**

None.

**Example**

- **RESEND CMD WACT**
  
  A problem occurred earlier with the WACT transaction. RESEND is used to invoke the transaction without NetMizer delta optimization.
RESETUSR

RESETUSR resets a PIE userid based upon the current state of the user.

RESETUSR changes a user’s status to passed who are in a TSWITCHed state. A TSWITCHed user is disconnected from PIE, but still logically connected. If the userid is of the Unique type, then the user cannot log onto PIE from a terminal that is logically connected to a different terminal. RESETUSR removes the logically connected user attribute. When the terminal is logged off the current application, it returns to the CICS log on screen but disallows auto-sign on. The user may now sign on at another terminal, even if the user’s userid is unique.

The DISCONNECT=NO state is set for users who are not in a TSWITCHed state. The next time the user logs off of CICS (explicitly or via a terminal error/lost session), the ID is logged off and not disconnected from PIE. This is useful when a user cannot explicitly log off due to a bad screen causing recursive ATNI abends.

If the user is already disconnected when the reset occurs, they are initially placed at the Sessions menu instead of their last session when they log on again and re-connect. This is a preventative measure to allow the user to cancel a session that is causing ATNI abends due to a bad datastream.

The userid is reset for the users identified by the USERID,TERMINAL, or APPLID parameters. All three parameters are optional and may be specified as wild card masks. This permits more than one userid to be reset with one command. For example, RESET USERID=*,TERMINAL=LOC* resets all users on terminals whose VTAM nodename begins with LOC.

PIE/CICS provides several methods to reset a user. Users can be reset from the PIE Utility List of User with the R line command. Also, Reset can be issued from the MVS console with the Modify command:

F jobname,P#RS USERID=userid,TERMINAL=nodename,...

Format

```
RESETUSR [USERID=userid] 
[,TERMINAL=nodename] 
[,APPLID=physical_name] 
[,EXCMSG|NOMSG]
```

Parameters

- **USERID**: Userid or mask.
- **TERMINAL**: CICS termid (if preceded with colon :) or VTAM nodename or a mask.
- **APPLID**: PIE physical name or mask.
- **EXCMSG**: Display only error messages.
- **NOMSG**: Suppress all messages.

Example

- `RESETUSR USERID=SP0U1`
Userid SPOU1 is reset.
RETOFF

RETOFF disables the PIE RETURN command and forces an ISPF END in the current session. Users step backwards through the active terminal sessions in the reverse order they were opened.

A user cannot return immediately to a main menu, which is normally the case with the RETURN command. RETOFF prevents users from using the designated RETURN key or executing the RETURN command to bypass a required stacked command or menu.

RETOFF is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

Format

RETOFF

Parameters

None.

Example

- RETOFF BROWSE BIGNEWS;BROWSE GOODNEWS

While viewing BIGNEWS, the user cannot press the RETURN key to bypass GOODNEWS, which is the next stacked command.
Chapter 3  Application Commands

RETON

RETON restores the PIE RETURN function that had been previously disabled with the RETOFF command. Users can return to a main menu by pressing the designated RETURN key or entering the RETURN command to bypass intervening PIE/CICS menus.

RETON is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user's current profile.

Format

RETON

Parameters

None.

Example

- RETOFF BROWSE BIGNEWS;BROWSE GOODNEWS;RETON BROWSE BADNEWS;AE CMD WOOT

While browsing BADNEWS, a user can press the RETURN key to cancel the browse and start executing the stacked WOOT transaction. However, the RETURN key is inactive while browsing BIGNEWS and GOODNEWS because of the RETOFF command.
RETURN

RETURN exits from the current session and returns the terminal to a designated PIE/CICS main menu. If RETURN is entered from system mode, the terminal returns to the preceding session that was open before system mode was entered.

Each subsequent RETURN command returns the terminal to the immediately preceding main menu in the session stack. This process continues until the PIE final command is executed from the user's profile.

Format

RETURN parameters

Parameters

parameters If a parameter is specified, it is passed back one level only. This has the effect of scheduling a command/menu option by returning back one level and executing the command/menu option from that menu. If a Dynamic Menu is the destination of the RETURN command, then a selection option from the menu can be specified as a parameter. If the destination is the Sessions menu, any command can be specified as a RETURN parameter. The command is executed instead of displaying the Sessions menu.

Example

- RETURN CEMT

From the Menu Entry screen to specify Dynamic Menus options, CEMT has been set as an option.

Execute ====> CEMT
Type ====> F

The CEMT option from the Dynamic Menus screen is executed when a RETURN command returns the terminal to this screen.
The PIE Exec language can create routines that execute a repeating sequence of transactions or commands. These routines are logical loops that continue until CICS is stopped or reset.

PIE has a loop counter. A call by the PIE Exec processor without a terminal input request adds 1 to the loop count. Looping stops when the count reaches 250. This prevents infinite loops from occurring.

RSTLOOP resets the PIE loop counter to zero in the current session. RSTLOOP prevents PIE from stopping a looping routine. RSTLOOP can be used to create deliberate wait states based upon looping routines. By executing the RSTLOOP command within the loop resets the counter and allows looping to continue.

RSTLOOP is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user's current profile.

**Format**

```
RSTLOOP
```

**Parameters**

None.

**Example**

- RSTLOOP

  RSTLOOP resets the PIE loop counter to zero in the current session.
RTRANID

RTRANID specifies a remote transaction code that overrides the transaction code stored in the target region’s PCT when using Dynamic Transaction Routing. PIE Dynamic Transaction Routing overrides the remote transaction code of the remote transaction invoked in the current session with tranid.

Dynamic Transaction Routing must have been enabled by specifying YES to this option from the PIE Customization Options utility.

RTRANID is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

**Format**

```
RTRANID tranid
```

**Parameters**

- **tranid**: CICS transaction ID that overrides the transaction ID stored in the target region’s PCT.

**Example**

- `RTRANID CEMT AE CMD CEOA`

  CEOA is the tranid of a remote transaction stored in the target region’s PCT. CEMT is substituted instead when CEOA is invoked in the remote region.
RUN

RUN sequentially processes commands stored in a CICS temporary storage queue assigned to the current session. This queue holds the commands written by any application running in a session with a link to the PIE Exec processor.

RUN recursively invokes the PIE Exec processor to execute the commands in the same order they were stored in the queue. Commands are processed until the queue is empty. The queue is deleted after the last command is processed.

Normally, RUN executes all commands in the queue unless the session is prematurely terminated or PIE/CICS encounters an error reading the queue. In both cases, the queue is deleted. The queue is automatically deleted at the end of a session if RUN was never used to process the stored commands.

Command parameters are passed to the Variable Substitution exit. On return from this exit, PIE/CICS variables are substituted with run-time values. The resultant parameters are passed to the Command Authorization exit, which can reject or accept this command. If the command is accepted, the transaction is given control via transaction XCTL.

Format

```
RUN
```

Parameters

None.

Example

- RUN

  Commands are processed sequentially from the temporary storage queue assigned to the current session.
SCRIpt

SCRIpt controls PIE/CICS playback scripts that automate user tasks. A playback script records a sequence of keystrokes that achieves some task. It has no logic functions to make conditional branches within the script to adapt to a particular situation. PIE/REXX should be used instead to create scripts with the capability to make conditional branches. A playback script stops processing when you switch out of the session in which the script is active.

PIE variables can be used with a playback script. Current values are substituted for variables when the script runs. For example, if you write a script to log on to a VTAM application, you can supply the variable &ZUSER when you record the script. When user J SMITH executes the script, J SMITH is substituted for the variable.

Both playback and REXX scripts are PIE objects stored in the PIE Repository. REXX scripts are text objects, listed with the Text Member List utility. Playback scripts are “script” objects, which can be listed with the Script List utility.

Format

| AUTHOR piexec_command_string |
| EXECUTE [SYSTEM]groupdid.member_name |
| BROWSE [SYSTEM]groupdid.member_name |

SCRIpt STOP

| SAVE [SYSTEM]groupdid.member_name title |
| DELETE [SYSTEM]groupdid.member_name |
| MATCHSCR FROM=screen_location TO=screen_location |
| MATCHOUT FROM=screen_location TO=screen_location |

Parameters

AUTHOR
Record a script. If piexec_command_string is given, this command is executed. Normally a PIEXEC command string will be specified. If authoring is started without it, then it must be executed under the same conditions that is was authored. For example, if authoring was started in the middle of a pseudo-conversational transaction, it must be executed there also.

EXECUTE
Play the script. Groupid.member_name is the name of the script file. If omitted, the user is prompted for the name.

BROWSE
Browse the script without executing it. Groupid.member_name is the name of the script file. If omitted, the user is prompted for the name.

STOP
Stop script recording

SAVE
Save the recorded script to the PIE repository. A script should be saved immediately after recording stops. Groupid/member name is the name of the script file to be created. Title is the title to be given to the script file. If omitted, the user is prompted for a name.

DELETE
Delete a script from the PIE repository. Groupid/member name is the name of the script file to be deleted. If omitted, the user is prompted for the name.
MATCHSCR  Match screen. This records a screen match of data on the screen. This command is issued during script recording to indicate the next response (pressing ENTER, for example) is not issued until a match is made for the specified data and screen location. The matching data is indicated by the FROM and TO keywords. If omitted, the defaults are FROM=CURSOR and TO=ENDFIELD. See screen_location below for other values.

MATCHSCR is optional and only need be used with certain types of applications.

MATCHOUT  Match output. This records a match of data on the screen regardless of the location of the field. This command is issued during script recording to indicate the next reply (pressing ENTER, for example) is not issued until a match is found for the specified data. The recorded data to match on is indicated by the FROM and TO keywords. If omitted, the defaults are FROM=CURSOR and TO=ENDFIELD. See screen_location below for other values.

MATCHOUT is optional and only need be used with certain types of applications.

screen_location  Terminal screen position to match data used in a playback script.

CURSOR  Current cursor position. The current cursor position can be specified by the following:

(row,col)  Row and column of the 3270 screen. 01,01 would be the very first position on the screen at the top left corner.

position  Cursor position in the 3270 buffer. 0 would be the first position on the screen.

BEGFIELD  Beginning of the field where the cursor is located.

ENDFIELD  End of the field where the cursor is located.

BEGINLINE  Beginning of the line where the cursor is located.

ENDLINE  End of the line where the cursor is located.

NOTES:

- If SCRIPT is invoked without parameters, a pop-up window lists the parameters and associated PF keys. Pressing a PF key is equivalent to entering SCRIPT and the parameter together.

- The STOP, SAVE, MATCHSCR, and MATCHOUT parameters are normally issued through a SCRIPT pop-up menu during a script recording session. The BROWSE and DELETE parameters are normally executed as line commands from the LIST SCRIPTS utility.

**Examples**

- **SCRIPT AUTHOR ACCESS APPLID=TSO**

  Script recording starts by accessing TSO through NetGate.

- **SCRIPT EXECUTE SYSTEM.TSLOGON**
The SYSTEM.TSOLOGON script is executed to log on to TSO through NetGate.
SEND

SEND starts the MultiCICS SuperMessage facility to broadcast non-destructive messages to selected PIE/CICS users. Neither user data nor transactions are affected by a message. Messages can be sent from a PIE/CICS menu or an application command line.

A SuperMessage can be received by several methods. By default, users do not see their messages until they press an AID key. The user’s current screen is retained when the message appears. After viewing the message, pressing the AID key again restores the original screen.

SEND command parameters specify the following:

- Message recipient
- Message delivery method
- Message format

If SEND is specified with all required parameters, a message is sent and a confirmation message appears on the screen of the issuing terminal.

If the command contains errors or is missing a required parameter, the SuperMessage utility appears with a message describing the error. The fields of the SuperMessage screen are completed with the parameters that were correctly specified with the invalid SEND command.

The SuperMessage utility appears if SEND is issued without parameters. The following figure shows an example of the SuperMessage screen with fields to complete that are equivalent to SEND parameters.

![SuperMessage Screen](image-url)
**Format**

```
SEND|PSEN
[Userid=pie_userid,]
[TERminal={nodename\ncics_termid},]
[System={applid\nsysid},]
[GRoupid=pie_groupid]
[APpl=pie_logical_name]
[TPype=pie_app_type]
[TRans=pie_phys_name]
[STatus={CONN\nDISC},]
[IMM\nBRO\nBOX\nREX\nONL]
[RECque={PVT\nPUB}]
[FROMUser=user_id,]
[FROMTerm=termid,]
[EXCmsg,]
[NOmsg]
{Message=message_text}
DSTMem={groupid.membername\nmembrename}
```

**Parameters**

- **Userid** 1 to 8-character user ID.
- **Terminal** VTAM node name or CICS termid. Begin a CICS termid with a :: For example, specify the CICS termid 1234 as :1234.
- **System** Specify the last system the user accessed. For NetGate, PASS, and...
### Chapter 3  Application Commands

TSWITCH users, this will be a VTAM APPLID. For transaction routing users, this will be a SYSID.

<table>
<thead>
<tr>
<th>Groupid</th>
<th>User Group name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>PIE logical name of the user’s current application. The logical name is the name of the session or menu line the application is executed from. (See the Sessions menu or Dynamic Menu for the name.)</td>
</tr>
<tr>
<td>Type</td>
<td>PIE application type code. For example, the code for a transaction would be CMD or TRAN. A PIE menu would be MENU.</td>
</tr>
<tr>
<td>Trans/PName</td>
<td>PIE physical name of the user’s current transaction. This is the CICS tranid or PIE menu name.</td>
</tr>
<tr>
<td>CONN/DISC</td>
<td>Current PIE user status. Specify CONN for users connected to PIE/CICS. Specify DISC for users who are signed on to PIE/CICS, but who are currently disconnected.</td>
</tr>
<tr>
<td>Delivery</td>
<td>Delivery method of the message. You may specify:</td>
</tr>
<tr>
<td>IMM</td>
<td>The message is displayed the next time the user presses an AID key. Users can press any AID key to leave the message display. This is the default.</td>
</tr>
<tr>
<td>BRO</td>
<td>Users are placed in PIE browse mode the next time they press an AID key. Browse is useful for long messages because it allows users to scroll forward or backward through the message. Users can exit browse with PF3 or the END command.</td>
</tr>
<tr>
<td>BOX</td>
<td>Deliver the message in a PIE message box that overlays the current screen. With this option, you must use the Message Text box (not a text file). The message is restricted to only 3 lines of text. The screen is restored when the user presses ENTER.</td>
</tr>
<tr>
<td>REX</td>
<td>Execute a PIE REXX program when the message is delivered. The program runs as a transaction for each user receiving the message. It can retrieve the message as the argument and perform any number of functions. For example, it can send a warning message and then log off each user. If you want to broadcast a lengthy, multi-part announcement, you can provide PF keys that allow users to navigate to different parts of the message.</td>
</tr>
<tr>
<td>ONL</td>
<td>The message is not displayed automatically. It is</td>
</tr>
</tbody>
</table>

The REXX program name is SUPERMSG. For the group ID, PIE/CICS performs its standard hierarchical search—user ID, group ID, SYSTEM—for each recipient. With this set up, you can have SuperMessage deliver the message any way you like, and you can have it deliver messages differently for different users.
placed in the receive queue. Users must explicitly browse their message queue to read the message. (See “Recv Que,” below, for instructions on browsing a message queue.)

<table>
<thead>
<tr>
<th>Recv Que</th>
<th>Type of receive queue. You may specify:</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVT</td>
<td>Create a unique private queue for each user on the distribution list. Users can browse their queue with the PIE BROWSE PIE.MSG Application command.</td>
</tr>
<tr>
<td>PUB</td>
<td>Create a public queue for each CICS region. All users can browse this queue with the PIE BROWSE PIE.SYSMSG Application command. Using a single queue is more efficient, but less secure than a private queue. This is the default.</td>
</tr>
</tbody>
</table>

**Examples**

- **SEND**

  The SuperMessage utility appears if SEND is issued without parameters.

  - **SEND U=*,M=CICS IS COMING DOWN AT 5 P.M.**

    The message is sent to all users on the current CICS region.

  - **F CICSPROG,PSEND TRANS=AP01,DEL=BRO,MSGMEM=OPERATOR.APMSG**

    Send a message from the console to all users on CICSPROD who are currently using transaction ‘AP01’. The message is in PIE text member OPERATOR.APMSG because this is a long message (over 1 page). You want users to be placed in browse mode as they open the message. Browse mode enables users to scroll forward and backward as they read the message.
SES

SES executes a session command within a PIE/CICS application session. SES permits users to manage their PIE/CICS sessions from an application without being required to return to the Sessions menu.

If SES is entered without a command, a PIE pop-up window appears that lists common session commands that can be selected with PF keys.

Commands and parameters are passed to PIE/CICS Session Manager. The command is executed if it is valid. If the command is invalid, a PIE window appears with a list of common environment commands. Each command is associated with a PF or PA key. Pressing the appropriate key executes the command.

SES is an alias of the following PIE/CICS commands:

- EC
- ENV
- SM

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

SES [session_command] [parameters]

Parameters

<table>
<thead>
<tr>
<th>session_command</th>
<th>PIE/CICS environment command executed from the Sessions menu.</th>
</tr>
</thead>
<tbody>
<tr>
<td>parameters</td>
<td>PIE/CICS environment command parameters.</td>
</tr>
</tbody>
</table>

Examples

- SES
  A PIE pop-up window appers with a list of common session commands that can be selected with PF keys.
- SES CLOSE PAYROLL
  The session named PAYROLL is closed.
- SES BUL
  The bulletin board is displayed.
**SIGNOFF**

SIGNOFF logs off a user from all active PIE/CICS sessions. All active and pending PIE/CICS transactions are cancelled that are scheduled for the user. The terminal is retained by CICS and can accept a subsequent log on request from the same or a different user.

If a PIE/CICS session cannot be cancelled, a popup alert window appears in that session after attempting to cancel all sessions with the SIGNOFF command. The affected session must be terminated normally before cancelling PIE/CICS sessions with SIGNOFF.

**Format**

```
SIGNOFF
```

**Parameters**

None.

**Example**

- **SIGNOFF**

  All active PIE/CICS sessions are cancelled. Depending upon parameters set in the current profile, the user returns to a native CICS session, a VTAM log on screen, or to the preceding application that was active before PIE/CICS was started.
Chapter 3  Application Commands

SM

SM executes a session command within a PIE/CICS application session. SM permits users to manage their PIE/CICS sessions from an application without being required to return to the Sessions menu.

SM parameters are passed to PIE/CICS Session Manager. If they are a valid command string, SM is processed and the transaction is given control with transaction XCTL.

The parameters are passed to the Variable Substitution exit. On return from this exit, PIE/CICS variables are substituted with run-time values. The resultant parameters are passed to the Command Authorization exit, which can reject or accept this command. The Command Authorization exit may even modify the parameters further.

If the parameters are invalid, the PIE pop-up window is displayed with an error message. The pop-up window lists a subset of common PIE commands that can be selected with a PF key. Also, if SM is entered without a command, the PIE pop-up window appears listing the same common session commands.

SM is an alias of the following PIE/CICS commands:
- EC
- ENV
- SES

The commands are functionally equivalent. Page references to these commands are listed in the table at the beginning of this chapter.

Format

```
SM [session_command] [parameters]
```

Parameters

- **session_command**: PIE/CICS environment command executed from the Sessions menu.
- **parameters**: PIE/CICS environment command parameters.

Examples

- **SM**

  A PIE pop-up window appears with a list of common session commands that can be selected with PF keys.

- **SM PROF KEY**

  The Global PIE Keys screen of the user’s current profile is displayed.

- **SM SWITCH 2**

  Session 2 listed on the Sessions menu is switched to within an application session.
**START**

START initiates a transaction with the CICS Automatic Task Initiation (ATI) facility. The EXEC CICS START command. START parameters are passed as data.

The time interval is set to zero. The PIE Exec command ends immediately and relinquishes session control to the transaction specified with the START command.

**Format**

```
START transid [parameters]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>transid</td>
<td>Name of the CICS transaction. A tranid must be specified with the START command.</td>
</tr>
<tr>
<td>parameters</td>
<td>Additional parameters belong to the specified transaction. The task can retrieve these parameters with the EXEC CICS RETRIEVE command. The parameters are passed to the Variable Substitution exit. On return from this exit, PIE/CICS variables are substituted with run-time values. The resultant parameters are passed to the Command Authorization exit, which can reject or accept this command. It may make further modifications to the parameters.</td>
</tr>
</tbody>
</table>

**Example**

- `ST xbz`
  
  Transaction XBZ is started at the current terminal.
STORage

STORage browses selected PIE control blocks or an address range in memory. Also, PIE tracing can be specified with the STORage command. The Browse facility displays the selected control block or address range in memory. A message appears above the first row of the display that shows the starting address and length of memory available for browsing based on parameters included with the STORage command.

STORage [CB=PIE_Control_block] [ADDRESS=00000000|xxxxxxxx] [LENGTH=256|yyyyyyyy] [TRACE=OFF|ON]

Parameters

CB  1 to 8 - alphanumeric character name of a PIE control block to browse in memory.
The CB parameter is mutually exclusive with other STORage parameters

ADDRESS  1 to 8 - hex number offset within memory as the starting address to browse. The default starting address is 00000000.
The offset address is padded with leading zeros if the starting address is specified with less than 8 digits.
The ADDRESS parameter is used in conjunction with the LENGTH parameter. It is mutually exclusive with the CB or TRACE parameters.

LENGTH  1 to 8-hex length of memory to browse in bytes. The default length is 256 bytes.
The LENGTH parameter is used in conjunction with the ADDRESS parameter. It is mutually exclusive with either the CB or TRACE parameters.
TRACE Request for global PIE tracing. The default is OFF. The TRACE parameter is mutually exclusive with other STORage parameters

Example

- STOR ADDRESS=01000FFF LENGTH=512

  The Browse facility displays the contents of memory starting from address 01000FFF and continuing for the next 512 (hex) bytes.
SYSID

SYSID identifies a target application by its CICS SYSIDNT or a VTAM APPLID to route remote transactions, or switch to this VTAM application with a CLSDST PASS operation.

Normally, SYSID is used in conjunction with the TN, TP, TR, and TS access commands. These commands set remote transaction access methods. The TN, TP, and TS commands route transactions to the application identified by the APPLID parameter of SYSID. The SYSIDNT parameter is used instead by the TR command to identify a remote application.

If either the APPLID or SYSIDNT parameters are specified as SELF, the current CICS region’s SYSIDNT is assigned as the target for remote transactions.

SYSID is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

Note Dynamic Transaction Routing must be enabled to support remote transaction routing provided by the TN, TP, TR, and TS access commands. Dynamic Transaction Routing is specified as an option from the Customization Options utility.

Format

SYSID {applid|sysidnt}

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>applid</td>
<td>1 to 8-character VTAM applid of the destination application that executes remote transactions identified by the TN, TP, and TS remote access method commands.</td>
</tr>
<tr>
<td>sysidnt</td>
<td>1 to 4-character CICS SYSIDNT used by the TR command to route remote transactions to CICS regions by the Transaction Route method.</td>
</tr>
</tbody>
</table>

Examples

- **TS SYSID CICSPRD1 AE CMD CEMT I TAS**
  
The CEMT transaction is routed to the CICSPRD1 region and executes by the transaction switch method.

- **SYSID AOR2 AE CMD CEDA**
  
  CEDA is executed in AOR2. The default TR access method was used to route the transaction.
**SYStem**

SYStem places the current terminal into an emulated native CICS transaction environment called system mode. After the command is entered, PIE/CICS erases the current screen and restores the standard CICS environment. Users can enter transactions from the familiar CICS blank screen.

PIE/CICS remains in system mode until an END or RETURN command is executed. Control returns to the preceding session that was open when the SYStem command was entered.

SYSTEM is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

**FORMAT**

```
SYStem
```

**Parameters**

None.

**Example**

- `SYSTEM PEXE CEMT I TERM`

The current session is placed into system mode to execute the CEMT transaction from the CICS blank screen. Entering either the END or RETURN commands restores the previous application that was open when the SYStem command was entered.
TERMDIR

TERMDIR opens the Terminal Directory. After entering the command, the Terminal Directory appears on the current session screen, as shown in the following figure.

<table>
<thead>
<tr>
<th>Command</th>
<th>Scroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>--- Terminal Userid.. Password Logon... Logof PIE Dial Opt SLM TRM RST Updated.</td>
<td>---</td>
</tr>
<tr>
<td>___ (MODEL)</td>
<td>PIELMAIN USE YES NO NO NO FUL NO 10/13/93</td>
</tr>
<tr>
<td>___ :MC24</td>
<td>SYS0 YES YES NO NO YES SRT NO 10/13/93</td>
</tr>
<tr>
<td>___ :M4B4</td>
<td>PIELMAIN YES YES NO NO YES SRT NO 10/13/93</td>
</tr>
<tr>
<td>___ PIEN2014</td>
<td>SYS1 YES YES FUL NO FUL NO 05/05/95</td>
</tr>
<tr>
<td>___ PIEN2015</td>
<td>USR2 YES YES FUL NO FUL NO 05/05/95</td>
</tr>
<tr>
<td>___ PIEN*</td>
<td>USR1 YES YES FUL NO FUL NO 11/05/96</td>
</tr>
<tr>
<td>___ :8E38</td>
<td>PIELMAIN YES FUL NO NON NO 04/18/95</td>
</tr>
</tbody>
</table>

PF: 1 HELP 3 END 4 RETURN 7 UP 8 DOWN

FORMAT

TERMDIR

Parameters

None.

Example

- TERMDIR

The Terminal Directory appears after issuing the command.
TN (NetGate Only)

TN enables PIE/CICS to intercept remote transactions that are normally handled by CICS MRO. PIE/CICS uses the NetGate ACCesS command to log on to the remote region and execute the transaction locally. The session remains at the remote region after the transactions is executed.

TN is mutually exclusive with other PIE/CICS remote access methods; TP, TR, and TS. TN should be preceded by a SYSID command that identifies the remote application’s VTAM APPLID.

TN can be used with other PIE commands to pass sign on data to the target region and automate the sign on process. The following conditions must be met before a remote transfer to a target region can occur with automated user sign on:

- The target region runs under PIE/CICS.
- The target region must run PSGM as the ‘good morning’ transaction.
- PROC must be the initial command executed for the user at the target region.

TN is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

**Note** Dynamic Transaction Routing must be enabled to support remote transaction routing provided by the TN, TP, TR, and TS access commands. Dynamic Transaction Routing is specified as an option from the Customization Options utility.

**FORMAT**

```
TN
```

**Parameters**

None.

**Example**

```
SYSID=CICSPRD1 AE TN ACNT JJONES WEEKLY
```

The ACNT transaction is routed to CICSPR1 by logging on to the region with the NetGate ACCesS command.
TP

TP intercepts remote transactions that are normally routed by CICS MRO. PIE/CICS uses the PASS command to log off the current terminal and log on to the remote region and execute the transaction. The terminal remains at the remote region after the transaction is completed.

TP is mutually exclusive with other PIE/CICS remote access methods; TN, TR, and TS. TP should be preceded by a SYSID command that identifies the remote application's VTAM APPLID. TP uses the full eight character name as the application's APPLID.

TN can be used with other PIE commands to pass sign on data to the target region and automate the sign on process. The following conditions must be met before a remote transfer to a target region can occur with automated user sign on:

- The target region runs under PIE/CICS.
- The target region must run PSGM as the 'good morning' transaction.
- PROC must be the initial command executed for the user at the target region.

TP is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user's current profile.

Note

Dynamic Transaction Routing must be enabled to support remote transaction routing provided by the TN, TP, TR, and TS access commands. Dynamic Transaction Routing is specified as an option from the Customization Options utility.

Format

TP

Parameters

None

Example

- SYSID=CICSPRD1 AE TP ACNT JJONES WEEKLY

The terminal is logged off of the current application and passed to the CICSPRD1 region to execute the ACNT transaction. The terminal remains logged on at ACNT after the transaction is completed.
TR processes remote CICS transactions with PIE/CICS’s dynamic transaction routing facility. The terminal remains logged on at the current application and remote transactions are routed by standard CICS MRO/ISC methods.

TR is the default remote access method. TR is mutually exclusive with other PIE/CICS remote access methods; TN, TP, and TS. TR should be preceded by a SYSID command that identifies the CICS region’s SYSIDNT.

TR is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

**Note**

Dynamic Transaction Routing must be enabled to support remote transaction routing provided by the TN, TP, TR, and TS access commands. Dynamic Transaction Routing is specified as an option from the Customization Options utility.

**FORMAT**

| TR |

**Parameters**

None

**Example**

- `SYSID CICSPRD1 AE CMD ACNT JJONES WEEKLY`

The ACNT transaction is invoked from the CICS blank screen and sent by PIE CICS’s dynamic transaction routing method to be executed at the remote CICSPRD1 region.
TRACE

TRACE provides the capability to trace PIE commands, external security system WTO console messages, and GTF data:

Format

```
TRACE {OFF|ON} {TYPE=CMD|WTO|USER} {[,.USERID=user_id][,.TERMINAL=nodename]} [.EXCMSG] [.NOMSG]
```

Parameters

- **ON or OFF**: Specify whether to turn TRACE on or off for this user.
- **TYPE**: Type of data that is traced.
  - **CMD**: Trace PIE commands issued by a user or from a designated terminal. The trace output is placed in a temporary storage queue (@YZTnnnn). This is equivalent to SET CMD TRACE ON. You can view the trace output stored in the queue with the BROWSE PIE.TRACE command.
  - **WTO**: Trace WTO console messages issued by an external security system providing PIE/CICS menu line security. This is the same as SET WTORACE ON.
  - **USER**: Specify before running a GTF trace. Run this type of trace only under the direction of UNICOM Systems, Inc. Software Customer Service.
- **USERID**: User ID of the user you want to trace. You may use wildcards.
- **TERMINAL**: VTAM node name or CICS termid. You can use wildcards. Always begin CICS termids with a:. For example, specify the CICS termid 1234 as:1234.
- **EXCMSG**: Display only error messages.
- **NOMSG**: Never display a message.
Example

- TRACE ON TYPE=CMD USERID=USR1

Commands issued by USR1 are traced and logged to a temporary storage queue. The following figure shows an example of a command trace log stored in a temporary storage queue.

```
****************************** TOP OF DATA ***************
13:37:25 USR1  00 SM:RETURNP
13:37:42 USR1  05 DM:BROWSE    browse pie.trace
13:37:42 USR1  05 DM:BROWSE    >PBRO PIE.TRACE
****************************** BOTTOM OF DATA ***************
```
Chapter 3  Application Commands

TRANsction

TRANsaction initiates a transaction through transaction XCTL. TRANsaction clears the screen, displays the text of the transaction, and then transfers control to the new transaction.

Parameters are passed in a TIOA to the Variable Substitution exit. On return from this exit, PIE/CICS variables are substituted with run-time values. The resultant parameters are passed to the Command Authorization exit, which can reject or accept this command. It may even modify the parameters further.

Format

| TRANsaction transid [parameters] |

Parameters

- **transid** 1 to 4-alphanumeric character transaction identifier.
- **parameters** Additional parameters between the transid and the end of the command string are treated as transaction parameters.

Example

- **TRAN ACTN**
  
  The ACTN transaction executes in the current session.
TS

TS processes remote CICS transactions with the PIE/CICS terminal switch (TSWITCH) method. PIE/CICS executes a CLSDST PASS transaction to transfer control of the current terminal to another VTAM application and execute the transaction. The current terminal is reacquired automatically after completing the remote transaction and logging off the target application.

TS is mutually exclusive with other PIE/CICS remote transaction switching methods: TN, TR, and TP. TS should be preceded by the SYSID command to identify the target system by its VTAM APPLID.

TN can be used with other PIE commands to pass sign on data to the target region and automate the sign on process. The following conditions must be met before a remote transfer to a target region can occur with automated user sign on:

- The target region runs under PIE/CICS.
- The target region must run PSGM as the ‘good morning’ transaction.
- PROC must be the initial command executed for the user at the target region.

TS is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

Note Dynamic Transaction Routing must be enabled to support remote transaction routing provided by the TN, TP, TR, and TS access commands. Dynamic Transaction Routing is specified as an option from the Customization Options utility.

Format

Parameters

None.

Example

- SYSID CICSPRD1 AE TS CMD CEMT I TAS

Control of the current terminal is switched to CICSPRD1 to execute the CEMT transaction. The current terminal is reacquired after completing the transaction and logging off CICSPRD1.
TSQSub (MultiCICS Only)

TSQSub permits temporary storage queue (TSQ) names to be substituted in the current session. Unique TSQ names are substituted for each PIE/CICS session based upon user ID. By default, PIE/CICS changes the terminal ID and/or the user ID to a binary number that is unique for each user and session.

- A three-character user ID and a one-character session ID are substituted for the four-character terminal ID.
- A two-character user ID and one-character session ID are substituted for the three-character operator ID.

TSQSub is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

**Format**

```
TSQSub
```

**Parameters**

None.

**Example**

- TSQSub
TSWITCH executes a CLSDST PASS transaction to transfer control of the current terminal to another VTAM application. The current terminal is reacquired automatically after the user logs off the target application.

Data can be passed to the target application as command parameters that are interpreted by the PIE/CICS Logon Director as the good morning transaction. The userid and password can be passed as &ZUSER and &ZPSWD variables respectively to automate the log on at the target application. Also, an old password can be passed with the &ZOPSWD and &ZEPSWD variables if the user changed their password at the PIE Logon screen. Other data that can be passed as TSWITCH parameters include the TSO PROC, initial CICS transaction, or PIE/CICS user profile.

Some customization is required to support TSWITCH application transfers. For example, the initial command in the target region must be PROC. A customization procedure is described in “Controlling Target Application Programs” on page 116 of the PIE/CICS Customization Guide.

**Format**

```
TSWITCH vtamappl [logonid[/password]] [data]]
```

**Parameters**

- **vtamappl** Remote VTAM application ID (APPLID).
- **logonid** User log on ID at the remote system.
- **old_password** Old password if the user changed their password from the PIE Logon screen. The old password can be passed in unencrypted and encrypted form with the &ZOPSWD and &ZEPSWD variables respectively. If the old password is sent in encrypted form with the &ZEPSWD variable, the target application must be either PIE/CICS or PIE/TSO.
- **password** User password for the remote VTAM application. If specified, the password must follow the logonid and be separated with a forward slash (/).
- **data** Optional parameter to specify an initial command or transaction that executes after log on is complete at the target application. Cannot be used without logonid.
Examples

- **TSWITCH CICSPRD USER1**
  
The terminal switches to the CICSPRD region. If the region is running the PIE/CICS Logon Director, USER1 is placed in the USERID field. The terminal returns to the originating session after logging off CICSPRD.

- **TSWITCH CICSPRD &ZUSER/&ZOPSWD/&ZPSWD**
  
The terminal switches to the CICSPRD region. The user has changed their password at the PIE Logon screen. The user's old and new passwords are sent with the &ZOPSWD and &ZPSWD variables respectively to automate the sign on at the target application.

- **TSWITCH CICSPRD &ZUSER/&ZPSWD CEMT I TAS**
  
The terminal switches to the CICSPRD region and passes the userid, password, and initial transaction that executes after log on is complete. The terminal returns to the originating session after logging off CICSPRD.
**UCT**

UCT converts terminal input data to uppercase in the current session.

UCT is a state command that sets the conditions of the immediate PIE/CICS command environment. Normally, a state command accompanies other commands that transfer control to another program, transaction, or PIE/CICS function. If entered by themselves, state commands execute the Default Application defined in the user’s current profile.

**Format**


**Parameters**

None.

**Example**

- UCT AE CMD ACCT

  Terminal input data resulting from the ACCT command is converted to uppercase.
Chapter 3  Application Commands

USERDATA, USERDATA2, USERDATA3, USERDATA4, and USERDATA5

The USERDATAx commands insert data into a user data field that is common to all PIE/CICS sessions. This data can be set to the &ZUDATAx variables or extracted by the PIE/CICS User Data Extraction Utility. Also, the data may be retrieved (in a TOR or AOR region) with the PC@UMRUD macro.

If the USERDATAx commands are specified with data, the PC@UPVUD user data validation exit is invoked to validate the data. The parameters are passed to the Variable Substitution exit. On return from this exit, PIE/CICS variables are substituted with run-time values. The resultant parameters are passed to the Command Authorization exit, which can reject or accept this command. The Command Authorization exit may make further modifications to the parameters.

Format

```
{ UserDATa } UserDaTa2
{ UserDaTa3 } UserDaTa4
{ UserDaTa5 } data
```

Parameters

data

Data placed into a user data field common to all PIE/CICS sessions. If omitted, the Change User Data panel is displayed.

USERDATAx commands are associated with specific &ZUDATAx variables. The following table shows the &ZUDATAx variable associated with each USERDATAx command and the byte length of the data placed into the field.

<table>
<thead>
<tr>
<th>Command</th>
<th>Length of Data</th>
<th>Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERDATA</td>
<td>8</td>
<td>&amp;ZUDATA</td>
</tr>
<tr>
<td>USERDATA2</td>
<td>16</td>
<td>&amp;ZUDATA2</td>
</tr>
<tr>
<td>USERDATA3</td>
<td>16</td>
<td>&amp;ZUDATA3</td>
</tr>
<tr>
<td>USERDATA3</td>
<td>16</td>
<td>&amp;ZUDATA4</td>
</tr>
<tr>
<td>USERDATA4</td>
<td>8</td>
<td>&amp;ZUDATA5</td>
</tr>
</tbody>
</table>

Example

- **USERDATA4 338844;SCRIPT EXECUTE SYSTEM.DOPAY1**

A playback script called SYSTEM.DOPAY1 invokes a transaction that enters data from the &ZUDATA4 variable into a screen field. The value of 338844 is placed into USERDATA4, which is associated with the &ZUDATA4 script variable.
USERDIR

USERDIR opens the PIE/CICS User Directory. After entering the command, the User Directory appears in the current session, as shown in the following figure.

<table>
<thead>
<tr>
<th>Command</th>
<th>Scroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERDIR</td>
<td>USERDIR</td>
</tr>
</tbody>
</table>

**Format**

```
USERDIR
```

**Parameters**

None.

**Example**

- USERDIR

The User Directory appears after issuing the command.
Appendix A Customer Support

This appendix describes how to get help from Customer Service when you experience a problem with a UNICOM Systems, Inc. Software product. This appendix includes separate sections that describe several diagnostic suggestions to rule out user errors and the information you should have ready before reporting the problem.

Contacting Customer Service

UNICOM Systems, Inc. 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, Inc. 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, Inc. 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 5:00 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, Inc. 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 P#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 that any recent changes to your site’s operating system, CICS, or other products are fully compatible with PIE/CICS.
- Verify that 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 that all PIE/CICS system tables were reassembled after applying maintenance to CICS or upgrading to another release.
- Verify that all users have current PIE/CICS passwords and have 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, Inc. 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, Inc. Software Customer Service:

CICS release and PUT Level ______________________________________________
CICS configuration MRO/ISC etc. ____________________________________________
Real or virtual terminal ___________________________________________________

Before calling UNICOM Systems, Inc. Software Customer Service, get answers to the following questions.

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  Command Variables

This appendix lists the variables that can be used with PIE/CICS commands. A brief description is included with each variable. The common ampersand (&) prefix is not included with the variable name.

INFO PIE/CICS status information line. This variable string includes the date, time, termid, CPU ID, system ID, and the userid.

INFO is equivalent to specifying the following variables: &ZDAYN &ZMONTHN &ZDAY &ZDATE &ZTIME term=&ZLTERM &ZTERM sys=&ZSYSID cpu=&ZSMF &ZUSER.

Z Null String.

ZAPPL(applid)
  Applid status from the Network Monitor.

ZAPPLD(applid)
  Applid description from the Network Monitor.

ZAPPLUSE(applid)
  Applid current/maximum users from the Network Monitor.

ZAPPLX(applid)
  Applid exception status from the Network Monitor. If the APPLID is available, the variable will be null.

ZATYPE Application type.

ZCDATE Date in the following formats based upon the value set with the DATFORM parameter in the CICS SIT.

MM/DD/YYYY
DD/MM/YYYY
YYYY/MM/DD

ZCJDATE Julian date in DDD.YYYY depending upon the value set with the DATFORM parameter in the CICS SIT.

ZCICREL CICS release number. Examples: 3.2.1, 2.1.2, 4.1.1.

ZCRY PIE/CICS copyright year expressed as a 4-digit number.

ZCUSDERS Number of users connected to PIE/CICS.

&ZCYEAR 4-digit year

ZDATE Date in specified CICS format as either dd/mm/yy or yy/mm/dd.
ZDAY Day of month (dd).
ZDAYN Name of day (MON,TUE,...).
ZDUSERS Number of users disconnected from PIE/CICS.
ZEPSWD Encrypted old password that was changed during the current signon. Encrypted passwords are supported by UNICOM Systems, Inc. Software’s PIE/CICS, PIE/TSO, and NonStop CICS products. This variable can be used only if the target VTAM application is PIE/CICS or PIE/TSO. CPUs that share or distribute encrypted user passwords must be synchronized to Greenwich mean time.
ZESC Escape sequence (switch string).
ZFPROF Fully qualified profile name of the current user. The profile name is composed of the group name and the profile name separated by a dot. For example: SYSTEM.ACCOUNT
ZGROUP Groupid of user.
ZHDESK Help desk ID.
ZINFO PIE/CICS status information line. This variable string includes the date, time, termid, CPU ID, system ID, and the userid. INFO is equivalent to specifying the following variables: &ZDAYN &ZMONTHN &ZDAY &ZDATE &ZTIME term=&ZLTERM &ZTERM sys=&ZSYSID cpu=&ZSMF &ZUSER.
ZINFOC Year expressed as a 4-digit number to incorporate the century within the date.
ZJDATE Julian date expressed as yy:ddd.
ZJOB CICS job name.
ZLANG Custom Menus language character
ZLNAME Logical name.
ZLOGMOD Logmode used for PASS and TSWITCH.
ZLOGPFX Log on hexadecimal prefix data 'X'00A0' (secured).
ZLPTERM CICS pseudo terminal identifier.
ZLSYSID CICS System identifier (local SYSID).
ZLTERM CICS Terminal identifier.
ZLUSERS Number of users currently logged onto PIE/CICS.
ZMONTH Month (mm).
ZMONTHN Month name (JAN,FEB,...).
ZOPID CICS Operator identifier.
ZOPSWD Old password that was changed during the current signon.
ZPIEREL PIE/CICS release number in the format: VV.RR.MM.
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZPNAME</td>
<td>Physical name.</td>
</tr>
<tr>
<td>ZPROF</td>
<td>Current profile name.</td>
</tr>
<tr>
<td>ZPRTALT</td>
<td>Alternate printer identifier.</td>
</tr>
<tr>
<td>ZPRTPRI</td>
<td>Primary printer identifier.</td>
</tr>
<tr>
<td>ZPSWD</td>
<td>Password. This is a secured variable and is not displayed. The password can be specified as a parameter with the PASS, TSWITCH, and ACCESS commands.</td>
</tr>
<tr>
<td>ZS</td>
<td>First string delimiter.</td>
</tr>
<tr>
<td>ZSESID</td>
<td>Current PIE session identifier.</td>
</tr>
<tr>
<td>ZSMF</td>
<td>SMF identifier of a CPU.</td>
</tr>
<tr>
<td>ZSYS(sysid)</td>
<td>SYSID status from the Network Monitor.</td>
</tr>
<tr>
<td>ZSYSD(sysid)</td>
<td>SYSID description from the Network Monitor.</td>
</tr>
<tr>
<td>ZSYSID</td>
<td>CICS APPLID.</td>
</tr>
<tr>
<td>ZSYUSE(sysid)</td>
<td>SYSID current/maximum users from the Network Monitor.</td>
</tr>
<tr>
<td>ZSYSX(sysid)</td>
<td>SYSID exception status from the Network Monitor. If the SYSID is available, the variable will be null.</td>
</tr>
<tr>
<td>ZTERM</td>
<td>VTAM node name.</td>
</tr>
<tr>
<td>ZTIME</td>
<td>Time of day in hh:mm:ss format.</td>
</tr>
<tr>
<td>ZTITLE</td>
<td>Title name.</td>
</tr>
<tr>
<td>ZTRANS</td>
<td>Last transaction identifier.</td>
</tr>
<tr>
<td>ZTSQPRF</td>
<td>PIE temporary storage queue prefix.</td>
</tr>
<tr>
<td>ZUDATA</td>
<td>User data (8 bytes). Can be abbreviated to Z1.</td>
</tr>
<tr>
<td>ZUDATA2</td>
<td>User data (16 bytes). Can be abbreviated to Z2.</td>
</tr>
<tr>
<td>ZUDATA3</td>
<td>User data (16 bytes). Can be abbreviated to Z3.</td>
</tr>
<tr>
<td>ZUDATA5</td>
<td>User data (8 bytes). Can be abbreviated to Z5.</td>
</tr>
<tr>
<td>ZUSER</td>
<td>User identifier.</td>
</tr>
<tr>
<td>ZX0</td>
<td>Hexadecimal value X'00'.</td>
</tr>
<tr>
<td>ZX1</td>
<td>Hexadecimal value X'01'.</td>
</tr>
<tr>
<td>ZX2</td>
<td>Hexadecimal value X'02'.</td>
</tr>
<tr>
<td>ZX3</td>
<td>Hexadecimal value X'03'.</td>
</tr>
<tr>
<td>ZX4</td>
<td>Hexadecimal value X'04'.</td>
</tr>
<tr>
<td>ZX5</td>
<td>Hexadecimal value X'05'.</td>
</tr>
<tr>
<td>ZX6</td>
<td>Hexadecimal value X'06'.</td>
</tr>
</tbody>
</table>
ZX7    Hexadecimal value X’07’.
ZX8    Hexadecimal value X’08’.
ZX9    Hexadecimal value X’09’.
ZXA    Hexadecimal value X’0A’.
ZXB    Hexadecimal value X’0B’.
ZXC    Hexadecimal value X’0C’.
ZXD    Hexadecimal value X’0D’.
ZXE    Hexadecimal value X’0E’.
ZXF    Hexadecimal value X’0F’.
ZYEAR  Year in the format of yy.
Appendix C  PIE/CICS Transactions

This appendix includes a table listing CICS transaction identifiers (tranids) used by various versions of PIE/CICS.

The table also includes the default security class assigned to each transaction. Security classes are designated as follows:

1. No security. PSND runs as a non-terminal attached task initiated by group 2 users.
2. End-user security.
3. Administrator security.

Transactions marked with * were added to PIE/CICS Version 2.3.0 and/or 2.4.0 with PTFs.

The PIE Exec facility uses some transactions to execute commands. The commands are listed at the right of the table.

<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>same</td>
<td>2</td>
<td>Text edit</td>
</tr>
<tr>
<td>P #ER</td>
<td>same</td>
<td>2</td>
<td>Alert window</td>
</tr>
<tr>
<td>none</td>
<td>none</td>
<td>2</td>
<td>Subtask options (NetGate/1)</td>
</tr>
<tr>
<td>P #GD</td>
<td>same</td>
<td>3</td>
<td>Installation Verification</td>
</tr>
<tr>
<td>P #IV</td>
<td>same</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>P #LU</td>
<td>PDUP</td>
<td>3</td>
<td>Repository Load utility</td>
</tr>
<tr>
<td>P #MN</td>
<td>same</td>
<td>2</td>
<td>Invoke a menu</td>
</tr>
<tr>
<td>P #ND</td>
<td>same</td>
<td>2</td>
<td>User Directory detail editor</td>
</tr>
<tr>
<td>P #NT</td>
<td>same</td>
<td>2</td>
<td>NetGate</td>
</tr>
<tr>
<td>P #OL</td>
<td>none</td>
<td>2</td>
<td>Limit transactions message</td>
</tr>
<tr>
<td>P #PC</td>
<td>PROC</td>
<td>2</td>
<td>PROC command</td>
</tr>
<tr>
<td>P #PI</td>
<td>same</td>
<td>2</td>
<td>Printer change command</td>
</tr>
<tr>
<td>P #PO</td>
<td>same</td>
<td>2</td>
<td>Pop-up pseudo command</td>
</tr>
<tr>
<td>P #PR</td>
<td>same</td>
<td>2</td>
<td>Screen print</td>
</tr>
<tr>
<td>P #RM</td>
<td>same</td>
<td>2</td>
<td>Run any remote tranid</td>
</tr>
<tr>
<td>P #RS</td>
<td>same</td>
<td>2</td>
<td>Resetuser command</td>
</tr>
<tr>
<td>P #RT</td>
<td>same</td>
<td>2</td>
<td>Restart command</td>
</tr>
<tr>
<td>none</td>
<td>none</td>
<td>2</td>
<td>Subtask region control (NetGate/1)</td>
</tr>
<tr>
<td>none</td>
<td>none</td>
<td>2</td>
<td>NetGate/1 shutdown</td>
</tr>
<tr>
<td>Tranid</td>
<td>2.4/3.3 Tranid</td>
<td>Security Class</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>---------------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>P#ST</td>
<td>same</td>
<td>2</td>
<td>Session Menu</td>
</tr>
<tr>
<td>P#SX</td>
<td>same</td>
<td>3</td>
<td>PIE control block display</td>
</tr>
<tr>
<td>P#S1</td>
<td>same</td>
<td>2</td>
<td>Sample Applications</td>
</tr>
<tr>
<td>P#TD</td>
<td>PTEL</td>
<td>3</td>
<td>Terminal directory edit</td>
</tr>
<tr>
<td>P#TM</td>
<td>same*</td>
<td>2</td>
<td>Terminal Security Monitor control</td>
</tr>
<tr>
<td>P#TR</td>
<td>new</td>
<td>2</td>
<td>Trace command</td>
</tr>
<tr>
<td>P#UD</td>
<td>PDIR</td>
<td>3</td>
<td>User Directory edit</td>
</tr>
<tr>
<td>P#XE</td>
<td>same*</td>
<td>2</td>
<td>REXX main controller</td>
</tr>
<tr>
<td>P#XP</td>
<td>same*</td>
<td>2</td>
<td>REXX script processor</td>
</tr>
<tr>
<td>P#XS</td>
<td>same*</td>
<td>2</td>
<td>REXX script application suspend</td>
</tr>
<tr>
<td>P#XX</td>
<td>REXX*</td>
<td>2</td>
<td>REXX top level controller</td>
</tr>
<tr>
<td>PAFU</td>
<td>same</td>
<td>3</td>
<td>Profile edit</td>
</tr>
<tr>
<td>PAMU</td>
<td>same</td>
<td>3</td>
<td>Menu edit</td>
</tr>
<tr>
<td>PAPP</td>
<td>same</td>
<td>2</td>
<td>Logon panel painter</td>
</tr>
<tr>
<td>PAPU</td>
<td>same</td>
<td>3</td>
<td>Logon panel edit</td>
</tr>
<tr>
<td>PATU</td>
<td>same</td>
<td>3</td>
<td>Text edit</td>
</tr>
<tr>
<td>PBAL</td>
<td>same</td>
<td>2</td>
<td>Load balance sample transaction</td>
</tr>
<tr>
<td>PBRO</td>
<td>same</td>
<td>1</td>
<td>Browse text</td>
</tr>
<tr>
<td>PBUL</td>
<td>same</td>
<td>2</td>
<td>Invoke bulletin menu</td>
</tr>
<tr>
<td>PCDP</td>
<td>same</td>
<td>1</td>
<td>Pass command</td>
</tr>
<tr>
<td>PEDT</td>
<td>same</td>
<td>1</td>
<td>Generalized line editor</td>
</tr>
<tr>
<td>PEXE</td>
<td>same</td>
<td>2</td>
<td>PIEEXEC</td>
</tr>
<tr>
<td>PEXX</td>
<td>same</td>
<td>2</td>
<td>PIEEXEC pseudo-converse tranid</td>
</tr>
<tr>
<td>PGED</td>
<td>same</td>
<td>2</td>
<td>Profile detail editor</td>
</tr>
<tr>
<td>PIE</td>
<td>same</td>
<td>2</td>
<td>Sessions commands</td>
</tr>
<tr>
<td>PINO</td>
<td>same</td>
<td>3</td>
<td>Installation options (password)</td>
</tr>
<tr>
<td>PLOG</td>
<td>same</td>
<td>2</td>
<td>Logmode change</td>
</tr>
<tr>
<td>PLOK</td>
<td>same</td>
<td>2</td>
<td>Lock command</td>
</tr>
<tr>
<td>PLSO</td>
<td>same</td>
<td>3</td>
<td>Object list display</td>
</tr>
<tr>
<td>PMND</td>
<td>same</td>
<td>2</td>
<td>Menu detail edit</td>
</tr>
<tr>
<td>PMZR</td>
<td>same</td>
<td>2</td>
<td>Funny display</td>
</tr>
<tr>
<td>PNAM</td>
<td>same</td>
<td>3</td>
<td>Network Monitor Commands</td>
</tr>
<tr>
<td>PNMZ</td>
<td>same</td>
<td>3</td>
<td>Netmizer stats</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>POPU</td>
<td>same 2</td>
<td>Pop-up menus</td>
<td></td>
</tr>
<tr>
<td>PPER</td>
<td>same 2</td>
<td>Perform command</td>
<td>PERFORM</td>
</tr>
<tr>
<td>PSCR</td>
<td>same 2</td>
<td>Script command</td>
<td>SCRIPT</td>
</tr>
<tr>
<td>PSEN</td>
<td>same 3</td>
<td>SuperMessage send panel</td>
<td>SEND</td>
</tr>
<tr>
<td>PSGM</td>
<td>same 1</td>
<td>Logon Director</td>
<td></td>
</tr>
<tr>
<td>PSND</td>
<td>same 1</td>
<td>SuperMessage background sender</td>
<td></td>
</tr>
<tr>
<td>PSNY</td>
<td>same 2</td>
<td>SuperMessage deliverer</td>
<td></td>
</tr>
<tr>
<td>PSTG</td>
<td>same 3</td>
<td>PIE storage stats</td>
<td></td>
</tr>
<tr>
<td>PSWC</td>
<td>same 2</td>
<td>Switch transaction</td>
<td></td>
</tr>
<tr>
<td>PSWH</td>
<td>same 2</td>
<td>Tswitch command</td>
<td>TSWITCH</td>
</tr>
<tr>
<td>PSXM</td>
<td>same 1</td>
<td>Force Logon Director next tranid</td>
<td></td>
</tr>
<tr>
<td>PSYS</td>
<td>same 2</td>
<td>Customization Options screen</td>
<td></td>
</tr>
<tr>
<td>PTSQ</td>
<td>same 3</td>
<td>TSQ utility</td>
<td></td>
</tr>
<tr>
<td>PUSR</td>
<td>same 2</td>
<td>User data update</td>
<td>USERDATA</td>
</tr>
<tr>
<td>PUSX</td>
<td>same 2</td>
<td>Logon Director extension</td>
<td></td>
</tr>
<tr>
<td>PVCU</td>
<td>same 2</td>
<td>CUT &amp; PASTE</td>
<td>SM CUT/SM PASTE</td>
</tr>
<tr>
<td>PVVC</td>
<td>same 2</td>
<td>View capture</td>
<td>SM VIEW</td>
</tr>
<tr>
<td>PVVS</td>
<td>same 2</td>
<td>View session menu display</td>
<td></td>
</tr>
</tbody>
</table>
Index

Symbols
&PSWD variable 123
&ZPSWD variable 135
&ZUSER variable 135

Numerics
3270 terminal
assigning keys with the KEY command 108
specifying screen position for a cut or paste operation 82

A
ACCessS command 89
ALias command 14
ALLOCate command 15
ALLOCATH command 16
ALLOCATT command 18
ATI command 92
ATTach command 93
ATTACHX exit
   NoATTach command 118
   use with ATTach command 93
   audience vii
   AutoEnd command 91

B
book conventions x
BROWSE command 94
BROWSEN command 95
bulletin
   setting with BUlletin command 19
   BUlletin command 19

C
CAncel command 22
CANCELC command 23
CE command 96
   CESF command 24
   CLOSE command 26
   CLOSEC command 27
   CLSDST PASS 135
   use with LOGMode command 113
   Command Access Authorization exit 8
   Command Authorization exit 99
   COMMAND/COM/CMDS command 99
   commands
      ACCesS 89
      ALias 14
      ALLocate 15
      ALLOCATH 16
      ALLOCATT 18
      ATTach 93
      AutoEnd 91
      BROWSE 94
      BROWSEN 95
      BUlletin 19
      CAncel 22
      CANCELC 23
      CE 96
      CESF 24
      CLOSE 26
      CLOSEC 27
      CMD 99
      CRReate 28
      CSSF 29
      CUT 31
      CUTEdit 33
      default execution 8
      DISC 35
      DISPlay 37
      DRop 38
      DSR 100
      EC 101
      EDIT 102
      ENCP 103
      ENd 39
      ENV 104
      ESCape 40
      Free 43
      GETC 105
      GOTO 106
      HELP 46
      HELPDESK 107
      Hide 47
      KEY 108
      LAUNCH 109
      LIST 110
      LOCK 48
      LOGMode 113
LOGoff 49
MENU 114
MSG 115
Name 51
NOAE 116
NoATI 117
NoATTach 118
NOCE 119
NOCLRST 121
NODSR 122
NoENCp 123
NOMSG 124
NON3270 125
NOPASSp 127
NoRBS 128
Notes 53
NOTSqsub 129
NOUct 130
OP1 132
OP2 133
OP3 134
OpEn 54
OPT 131
PASS 135
PASSp 137
PASTE 55
PERFORM 138
PEEXEC 141
PGM 144
PIE 56
PRINT 57
PRINTID 142
PROC 143
PROfile 58
PROGRAM 144
PROMpt 145
PROMptN 146
Query 59
queue processing 9–10
QUIT 61
RBS 147
RESTART 62
RETOFF 151
RETON 152
RETURN 64, 153
RETURNA 64
RSTLOOP 154
RTRANID 155
scheduling with the PIE Exec Link
Interface 9
SCRIpt 157
SEND 160
SES 164
SESMENU 66
SESsion 65
SET 67
SIGNOFF 165
SKIPnext 70
SM 166
STart 71, 167
STORage 168
structure
delimiters 3
escape string 4
session identifier 2
variables 1–2
wildcard characters 4
SWITCHR 75
TErm 76
TITLE 77
token 7
TRACE 176
tracing 69
TRANsaction 178
TSQSub 180
TSWITCH 181
UCT 183
Unhide 79
USERDATA 184
USERDIR 185
VIEW 82
VIEW CAPTURE 80
Confirm End command 96
Confirm End
disabling with NOCE command 119
contacting Customer Service 188
CREate command 28
CSSF command 29
CUT command 31, 82
CUTEDIT command 82
CUTEdit command 33

D
delimiters 3–??
delimiters 3
DISC command 35
DISPlay command 37
distribution lists
creating private message queue for
members on the recipient list 163
DROp command 38
DSR command 100
Dynamic Transaction Routing 103
overriding a remote transaction code
with the RTRANID command 155
sending profiles with the PERFORM
command 137

E
EC command 101
EDIT command 102
ENCP command 103
END command 39
ENV command 104
ESCAPE command 40
escape string 4
EXEC command 141
exits
Command Access Authorization 8
Command Authorization 99
Variable Substitution 99
variable substitution 7

F
FIFO Command Queue 9
Free command 43

G
GETC command 105
Global PIE Keys
menus 58
global PIE keys
assigning 67
disabling with the SKIPnext command
70
temporary assignment with the SET
command 67
GOTO command 106

H
HELP command 46
HELPDESK command 107
HIDE command 47

I
INFO variable 191, 192
initial command

from VTAM logon message 10
scheduling by PROC 10

K
KEY command 108

L
LAUNCH command 109
length attributes of variables 2
LIST command 110
LOCK command 48
LOGMode command 113
LOGoff command 49

M
MENU command 114
as a default command 8
messages
blanking screen with the PROMpt
command 145
SEND command 160
setting short messages 67
MSG command 115
MultiCICS
prevent substitution of temporary
storage queue names with the
NOTSqsub command 129
substituting temporary storage queue
names with the TSQSub command
180

N
NAme command 51
NetGate
passing the CICS print key with the
CICPRT command 97
starting a session with the ACCesS
command 89
NetMizer
disable error recovery handling with the
NODSR command 122
disable read buffer suppression with the
NoRBS command 128
enable error recovery handling with the
DSR command 100
enable full data stream optimization with
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPT command</td>
<td>131</td>
</tr>
<tr>
<td>enable level-1 data stream optimization with the OP1 command</td>
<td>132</td>
</tr>
<tr>
<td>enable level-2 data stream optimization with the OP2 command</td>
<td>133</td>
</tr>
<tr>
<td>enable level-3 data stream optimization with the OP3 command</td>
<td>134</td>
</tr>
<tr>
<td>enable read buffer suppression with the RBS command</td>
<td>147</td>
</tr>
<tr>
<td>set transparent data streaming with the NON3270 command</td>
<td>125</td>
</tr>
<tr>
<td>Network Monitor</td>
<td></td>
</tr>
<tr>
<td>use with ACCesS command</td>
<td>90</td>
</tr>
<tr>
<td>use with the ACCesS command</td>
<td>89</td>
</tr>
<tr>
<td>NOAE command</td>
<td>116</td>
</tr>
<tr>
<td>NoATI command</td>
<td>117</td>
</tr>
<tr>
<td>NoATTach command</td>
<td>118</td>
</tr>
<tr>
<td>NOCE command</td>
<td>119</td>
</tr>
<tr>
<td>NOCLRrst command</td>
<td>121</td>
</tr>
<tr>
<td>NODSR command</td>
<td>122</td>
</tr>
<tr>
<td>NOENCp command</td>
<td>123</td>
</tr>
<tr>
<td>NOMATIC command</td>
<td>124</td>
</tr>
<tr>
<td>NON3270 command</td>
<td>125</td>
</tr>
<tr>
<td>NOPASSP command</td>
<td>127</td>
</tr>
<tr>
<td>NORBS command</td>
<td>128</td>
</tr>
<tr>
<td>NOTes command</td>
<td>53</td>
</tr>
<tr>
<td>NOTSQSUB command</td>
<td>129</td>
</tr>
<tr>
<td>NOUCT command</td>
<td>130</td>
</tr>
<tr>
<td>OPI command</td>
<td>132</td>
</tr>
<tr>
<td>OPen command</td>
<td>54</td>
</tr>
<tr>
<td>OPT command</td>
<td>131, 134</td>
</tr>
<tr>
<td>OPT2 command</td>
<td>133</td>
</tr>
<tr>
<td>PIE Exec</td>
<td></td>
</tr>
<tr>
<td>authorization by user exit</td>
<td>8</td>
</tr>
<tr>
<td>command scheduling</td>
<td>9</td>
</tr>
<tr>
<td>command string format</td>
<td>5</td>
</tr>
<tr>
<td>defaults</td>
<td>8</td>
</tr>
<tr>
<td>FIFO Command Queue</td>
<td>9</td>
</tr>
<tr>
<td>Link Interface</td>
<td>9</td>
</tr>
<tr>
<td>modifications of commands by user exit</td>
<td>7</td>
</tr>
<tr>
<td>uppercase</td>
<td>7</td>
</tr>
<tr>
<td>using variables</td>
<td>7</td>
</tr>
<tr>
<td>PIE Exec facility</td>
<td>5</td>
</tr>
<tr>
<td>PIECVP TS queue</td>
<td>188</td>
</tr>
<tr>
<td>PNAM file</td>
<td></td>
</tr>
<tr>
<td>retrieving data for the ACCesS command</td>
<td>89</td>
</tr>
<tr>
<td>PRINT command</td>
<td>57</td>
</tr>
<tr>
<td>printer</td>
<td></td>
</tr>
<tr>
<td>setting CICS IDs with the PRINITID command</td>
<td>142</td>
</tr>
<tr>
<td>setting CICS print key with the CICPRT command</td>
<td>97</td>
</tr>
<tr>
<td>PRINTID command</td>
<td>142</td>
</tr>
<tr>
<td>PROC command</td>
<td>143</td>
</tr>
<tr>
<td>processing</td>
<td>10</td>
</tr>
<tr>
<td>PROfile command</td>
<td>58</td>
</tr>
<tr>
<td>Profile utility</td>
<td></td>
</tr>
<tr>
<td>starting with PROfile command</td>
<td>58</td>
</tr>
<tr>
<td>profiles</td>
<td></td>
</tr>
<tr>
<td>invoking with the PROfile command</td>
<td>58</td>
</tr>
<tr>
<td>prevent passing with the NoPaSSP command</td>
<td>127</td>
</tr>
<tr>
<td>PROGRAM/PGM command</td>
<td>144</td>
</tr>
<tr>
<td>PROMpt command</td>
<td>145</td>
</tr>
<tr>
<td>PRoMptn command</td>
<td>146</td>
</tr>
<tr>
<td>Q</td>
<td></td>
</tr>
<tr>
<td>Query command</td>
<td>59</td>
</tr>
<tr>
<td>QUIT command</td>
<td>61</td>
</tr>
<tr>
<td>R</td>
<td></td>
</tr>
<tr>
<td>RBS command</td>
<td>147</td>
</tr>
<tr>
<td>recommended reading</td>
<td>viii</td>
</tr>
<tr>
<td>Repository</td>
<td></td>
</tr>
<tr>
<td>Load utility</td>
<td></td>
</tr>
<tr>
<td>invoking with the Load command</td>
<td>112</td>
</tr>
<tr>
<td>RESETUSR command</td>
<td>149</td>
</tr>
<tr>
<td>RESTART command</td>
<td>62</td>
</tr>
</tbody>
</table>
RETOFF command 151
RETON command 152
RETURN command 64, 153
RETURNA command 64
RSTLOOP command 154
RTRANID command 155
RUN command 156
as default command 8
processing the FIFO command queue 9

S
screens
  cut and edit data with the CUTEdit command 33
  cut data with the CUT command 31
  disable CLEAR key from restoring the primary screen size with the NOCLRRST command 121
  pasting previously cut data 55
  restore the primary size of current screen with the CLRRst command 98
  setting temporary scroll length 68
SCRIipt command 157
security
  tracing WTO console messages with the TRACE command 176
SEND command 160
SES command 164
SESMENU command 66
  session 0 40
SESSION command 65
  session identifier 2
sessions
  allocate with CReate command 28
  assigning a toggle key 78
  assing a title with the TItle command 77
  cancel 22
  close 26
  close and deallocate with the CANCEL command 22
  close with the DRop command 38
  close with the Free command 43
  close with the TErm command 76
  conditionally cancel 23
  conditionally close 27
  conditionally close and deallocate with the CANCELC command 23
  create with ALLocate command 15
  creating a hidden session with ALLOCATH command 16
  creating temporary with ALLOCATT command 18
  disconnect 35
  exclude with the HIDE command 47
  hidden 16
  naming 51
  obtain status with the DIsplay command 37
  obtaining status with the Query command 59
  opening 54
  renaming temporarily with ALIas command 14
  restarting 62
  special session 0 40
  starting 71
  temporary 18
  unhide 79
  using EC command 101
Sessions Configuration menu 15, 16, 58
Sessions menu
  assigning a title with the TItle command 77
  switching to with the SESMenu command 66
  switching to with the SESsion command 65
SET command 67
SIGNOFF command 165
SKIPNEXT command 70
SM command 166
START command 71
STart command 167
state commands 5
storage
  viewing with the STORage command 168
STORage command 168
SuperMessage
  MSG command 115
  prevent delivery with the NoMSG command 124
  use with the SEND command 160
syntax conventions x
SYSID command 170
SYSTEM command as default command 8

T
UNICOM Systems, Inc. Software Concepts, phone and fax numbers 187
temporary storage queues 94, 95, 180
TERm command 76
TERMDIR command
commands
TERMDIR 172
Terminal Characteristics menu 58
Terminal Directory
invoking with the TERMDIR command 172
terminals
locking 48
setting logmode for remote transactions 113
Text Utility
viewing members with the BROWSE command 94
Text utility
browsing members with the BROWSE command 95
using EDIT command 102
TITLE command 77
TN command 173
TOGGLE command 78
token 7
TRACe command 176
transaction
identifiers 195–197
security classes 195
TRANsaction command 178
transactions
P#IV 188
TS command 179
TSQSub command 180
TSWITCH command 135, 179, 181
receiving SuperMessages 162

V
variable substitution 7
Variable Substitution exit 178
variables
&PSWD 123
&ZPSWD 135
&ZUSER 135
concatenation 2
list of 1
modifying length 2
ZHDESK 107
VIEW CAPTURE command 80
VIEW CUT/PASTE/CUTEDIT commands 82
VTAM
PROC command 143

W
wildcard characters 4

X
XCTL 6, 144

Z
Z variable 191
ZAPPL variable 191
ZAPPLD variable 191
ZAPPLUSE variable 191
ZAPPLX variable 191
ZATYPE variable 191
ZAPPLUSE variable 191
ZAPPLX variable 191
ZCICREL variable 191
ZCRY variable 191
ZCUSERS variable 191
ZDATE variable 191
ZDAY variable 192
ZDAYN variable 192
ZDUSERS variable 192
ZEPSWD variable 192
ZESC variable 192
ZFPROF variable 192
ZGROUP variable 191
ZHDESK variable 107, 192
ZINFO variable 192
ZDATE variable 192
ZJDATE variable 192
ZJOB variable 192
ZLNAME variable 192
ZLOGMOD variable 192

U
UCT command 183
UNHIDE command 79
uppercase
prevent translation with the NOUCt command 130
UCT command 183
User Directory
invoking with the USERDIR command 185
user exits
variable substitution 7
USERDATA command 184
USERDIR command 185
ZLOGPFX variable 192
ZLPTERM variable 192
ZLSYSID variable 192
ZTERM variable 192
ZLUSERS variable 192
ZMONTH variable 192
ZMONTHN variable 192
ZOPID variable 192
ZPIEREL variable 192
ZPNAME variable 193
ZPROF variable 193
ZPRTALT variable 193
ZPRTPRI variable 193
ZPSWD variable 193
ZS variable 193
ZSESID variable 193
ZSMF variable 193
ZSYS variable 193
ZSYSID variable 193
ZSYSUSE variable 193
ZSYSX variable 193
ZTERM variable 193
ZTIME variable 193
ZTITLE variable 193
ZTRANS variable 193
ZTSQPRF variable 193
ZUDATA variable 193
ZUDATA2 variable 193
ZUDATA3 variable 193
ZUDATA4 variable 193
ZUDATA5 variable 193
ZUSER variable 193
ZX0-ZXF variable 193
ZYEAR variable 194