PIE/TSO Release Notes

Release 3.2.1

UNICOM
SYSTEMS, INC.

PTSORN321-01

Printed: October 2002

This manual applies to PIE/TSO release 3.2.1 and to all subsequent releases of the product until otherwise indicated by new editions or updates to this publication.

UNICOM Systems, Inc.
Unicom Plaza - Suite 310
15535 San Fernando Mission Blvd.
Mission Hills, California 91345

Release Notes

Contents

Software Compatibility

PIE/TSO 3.2.1 requires the following software:

- MVS/SP 3.1 or above
- TSO/E 2.1.0 or above
- ISPF/PDF 2.3 or above
- ACF/VTAM 3.1.1. or above
- DFP 3.1 or above

PIE/TSO Network Access version 3.2.1 requires that you apply the IBM VTAM fix for APAR OY29863. This provides support for the new VTAM APPL keyword SESSLIM.

PIE/TSO supports the following security systems:

- RACF 1.9 or above
- ACF2 4.0 or above
- TOP SECRET

Migrating

PIE/TSO release 3.2.1 cannot run concurrently with release 3.1.0.

If you are migrating to PIE/TSO release 3.2.1, perform the following in addition to the regular installation procedures detailed in the *PIE/TSO Installation and Customization Guide.*

1. Use a different load library for release 3.2.1. Do not install into the 3.1.0 load library.
2. Add a second PCSINIT procedure that includes a STEPLIB DD statement pointing to the new authorized library.
3. Change a TSO logon procedure to allocate the new libraries. Add a STEPLIB DD statement that points to the new authorized library.
4. You may use your existing SESSIONS dataset. Add MTTCMP=YES or MTTCMP=NO as specified in your license agreement.
5. To implement MultiTask, see "MultiTask," in Chapter 4 of the *PIE/TSO Installation and Customization Guide*, for instructions.
6. If you have coded Network Access exits, you must convert them to AMODE=31. RMODE=ANY is optional.

⚠ WARNING

Running these exits in AMODE=24 can cause Assemble your PCSUSERS with the new macro library and the new authorized library, using the new JCL in the CNTL dataset.

7. Run your new PCSINIT procedure.
8. Run PNAIPL to restart Network Access.
9. Log on to TSO using your new logon procedure.
10. Test PIE/TSO functions, especially MultiTask. (See the *PIE/TSO Administrator Guide*, "Using MultiTask," for instructions.)
11. When you have completed testing, put PTS321 into the production libraries.

Enhancements

- MultiTask is available as an optional component of PIE/TSO. With it, users can run a task in a background session while they work in a foreground session. MultiTask is activated with the SET MTASK command. You can control access to MultiTask in your SESSIONS and PCSUSERS members.
- The OKSWAP parameter has been added to the SESSIONS member of the PARMLIB dataset to allow you to determine how PIE/TSO handles swappability. If you specify OKSWAP=YES, PIE/TSO will maintain swappability for each session individually. If you specify OKSWAP=NO, if any session prohibits swapping, swapping is also prohibited for all other sessions. OKSWAP=NO is required for TCP/IP support.
- The VSTOR parameter has been added to the SESSIONS member to determine whether PIE/TSO will check available storage before opening a new session. Specify CHECK to perform the check. Specify NOCHECK to skip the check.
- New parameters have been added to the SESSIONS member to customize the text in the MultiTSO menu.

  CMDTXT Change the text of the command prompt on the MultiTSO menu. The text can be up to 12 characters long. It must be enclosed in quotes.

  HDRTXT Change the text of the header line on the MultiTSO menu. The text can be up to 64 characters long. It must be enclosed in quotes.

  TITLE1 and TITLE2 Change the text of the title line on the MultiTSO menu. TITLE1 specifies the text in column 1 to 70. TITLE2 specifies the text in column 71 to 79. Enclose the text for both options in quotes.
- A number of new SESSIONS parameters have been added to customize the user environment more efficiently. Each of these parameters has a corresponding PIE/TSO command. Previously the user environment was customized solely with these commands executed in an initial CLIST or REXX EXEC. Now you can set these commands as defaults through the SESSIONS member. With fewer commands in the initial CLIST or EXEC, user logon processing is reduced.

  ABBREV Specify the default for using shortened names to identify a session (for example, ED to identify the EDIT session). If you specify ON, users will be able to identify sessions with short names by default. If you specify OFF, users will not be able to do so unless they execute the SET ABBREV command.

  AE Specify the default for auto-end attribute. If set to ON, MultiTSO automatically ends the session if you log off the application running in it or if the application abends. If set to OFF, MultiTSO remains at TSO ready when the application ends. This default can be changed with the SET AE command.

  DCMD Specify the default initial command for all new sessions. You can specify any TSO command and its parameters. For example, specify DCMD=MSPF to automatically start ISPF whenever you open a new session. If you do not want a default command, leave DCMD blank. This default can be changed with the SET CMDDEFLT command.

  FCMD Specify the default final command to be executed when all sessions have been ended. You can specify any TSO command and its parameters. For example, specify FCMD=LOGOFF to automatically log the user off TSO when all sessions are ended. If you do

not want a default final command, leave FCMD blank. This default can be changed with the SET CMDFINAL command.

HEADER Specify OFF to blank out the header line on the MultiTSO menu—that is, the headings for the session portion. Specify ON to turn the header line back on. Same as SET HEADER.

IMPTSO Specify the default for "implied TSO." If YES, PIE/TSO will execute any unknown command as a TSO command. It will start a new TSO session and execute the command from that session. If NO, PIE/TSO will issue an error message for unknown commands. This default can be changed with the SET IMPTSO command.

INFWIN Specify OFF to omit the Information Window from the MultiTSO menu. Specify ON to redisplay it. This default can be changed with the SET INFOWINDOW command.

JMPKEY Specify the default for the jump key. Specify any key name to turn the jump key on and set it to a particular key (PF1 to PF24 or 1 to 24 or PA3). Specify NO to turn the jump key off. This default can be changed with the JMPKEY command.

KEYS Specify ON to activate MultiTSO function keys in sessions. When session keys are on, the keys in the SKEYS table will be activated and will override the normal key function for the application running in the session. The SKEYS are the keys defined with the SET PFn and SET PA3 commands. Specify OFF to use the application function keys in every session. This default can be changed with the SET KEYS command. (While the KEYS parameter can turn SKEYS on and off only for all sessions, the SET KEYS command can turn SKEYS on or off in particular sessions.)

MENU Controls MultiTSO menu display. You can use SET MENU OFF when you want to make MultiTSO transparent to users. This is useful, for instance, if you want to use only PIE/TSO's AutoLock feature. You can allow users only one session, activate that session automatically, and turn off the MultiTSO menu. AutoLock will work, but users will never know that PIE/TSO is on. This default can be changed with the SET MENU command.

MKEYS Controls the MultiTSO menu keys. If you specify MKEYS, the MultiTSO menu will use the MKEYS table for PF key function. If you specify SKEYS, it will use the SKEYS table. This default can be changed with the SET MKEYS command.

PFSHOW Controls display of the function key help line on the MultiTSO menu. Specify ON or OFF. This default can be changed with the PFSHOW command.

SESATT Controls display of the session attribute column, on the right side of the MultiTSO menu. Specify ON or OFF. This default can be changed with the SET SESATT command.

- Two new commands have been added.

  RESET id command allows you to reset an open session's default command. If you were to close and reopen that session, it would execute whatever you specified for command. If you omit command, you would get TSO READY. This is useful at the end of CLISTs and REXX EXECs. For example, you could execute your initial CLIST in session 1 and, at the end of the CLIST, reset session 1 to execute MSPF. The user would not see the CLIST displayed on the MultiTSO menu. Instead he would see MSPF. In addition, this prevents the user from accidentally re-executing the CLIST by closing and reopening the session. (The id parameter is required; specify the session code or name.)

  SET TMP tempname and SET SECTMP tempname allow you to change the terminal monitor program controlling the sessions you create after issuing the command. For tempname, specify IKJEFT01 for normal TSO READY. Specify ADFMDF03 for the Session Manager environ-

ment. SET TMP affects only session 0. SET SECTMP affects all other sessions.

SMP/E Installation

The following SMP/E installation instructions are for existing customers only.

SMP/E Installation Summary

Two tapes are required for SMP/E installation:

- PIE/TSO SMP TAPE (Volser PTS321)
- PIE/TSO CUMTAPE PUT LEVEL nnnn (Volser PTnnnn)

The following steps are required to install the base product and maintenance:

1. Unload base installation dataset
2. Create and tailor installation dataset
3. Create the SMP/E environment
4. Receive, apply, and accept base installation
5. Unload maintenance dataset
6. Receive, apply, and accept maintenance
7. Assemble PIE/TSO members
8. Continue with the *PIE/TSO Installation Guide*

Step 1: Unload and allocate Base Install PDS

In this step you will use IEBCOPY to unload and allocate the installation dataset. You will use jobs in this dataset for the remainder of the base product installation.

1. Create JCL like the sample below.
2. Change the JOB statement. For ddname SYSUT1, change the UNIT parameter. For ddname SYSUT2, change the DSN, UNIT, and VOL=SER parameters. Do not change any other parameters.
3. Run the job. It must run with condition code 0.
4. Save your changes. You will use them again in “Step 5: Unload Maintenance Tape,” on page 12.

SAMPLE IEBCOPY JCL

```
//STEP1    EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//SYSUT1   DD   DSN=TPTS321.F16,
//              UNIT=cart,                  <= Change unit
//              VOL=SER=PTS321,
//              DCB=(RECFM=FB,BLKSIZE=4080,LRECL=80),
//              LABEL=(17,SL,EXPDT=98000),
//              DISP=SHR
//SYSUT2   DD   DSN=your.base.install.pds.name, <= Change DSN
//              DISP=(,CATLG,DELETE),
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=4080),
//              SPACE=(CYL,(1,1,47)),
//              UNIT=sysda,                  <= Change unit
//              VOL=SER=volser               <= Change volser
//SYSIN    DD   DUMMY
```

Step 2: Copy and Tailor Installation Dataset

In this step you will run member IPOUPDTE. IPOUPDTE is a two-step job that copies the base install PDS as a target PDS, then invokes IPOUPDTE to change the JCL parameters in the other members of the installation dataset. For example, the literal string JOBACCTNG is coded in the job accounting field of the job statement in all install members. When you change the value in IPOUPDTE and run the job, you change the job accounting value in all members.

If you do not have access to the IBM utility IPOUPDTE, you must still run the first step of the IPOUPDTE job. Then you must manually change all the JCL parameters for the members in the installation dataset before you execute them.

1. Edit member IPOUPDTE.
2. Change the values for SYSIN parameters. The parameters are described in the table, below.

   The SYSIN input appears after the first </*. You will see the following in the SYSIN data-stream:

   ```
   </*ACCTNG<+     (the parameter)
   </*JOBACCTNG<<  (the value to change) <======= REPLACE WITH ...
   ```

   That is, the parameter and its value appear on separate lines: the parameter is on the first line, its value is on the second line. Be sure to change the value, not the parameter.

   ☞ Do not change the value for TSCFMID. Do not remove any of the variables or the <+ or << characters in the SYSIN data stream, and do not delete member $$$COIBM.

   The IPOUPDTE job will always create a new target dataset. Check that the target dataset name does not already exist. If it does, the job will get a not cataloged condition and errors will result.
3. Run the job. It must run with condition code 0.

PARAMETER TABLE

Parameter	Default	Description
ACCTNG	JOBACCTNG	Job accounting
TSCINSTALLPDS	YOUR.TARGET.VALUE	Installation dataset. This is where changes are kept.
TSCCUSTHLQ	PIE.TSO321.SMP	TSC high level qualifier.
TSCSMPVOL	SMPVOL	Global SMP volser.
TSCSMPUNIT	3380	Unit esoteric for SMP Global.
TSCTVOL	TARG01	Target SMP volser.
TSCTUNIT	3380	Unit esoteric for SMP Target.
TSCDVOL	DIST01	Target SMP volser.
TSCDUNIT	3380	Unit esoteric for SMP DLIB.
TSCFMID	TPTS321	FORMFMID. DO NOT change this value.

Parameter	Default	Description
TSCTARGET	PTTSTGT	Target zone name. We recommend that you do not change this value.
TSCDISTLIB	PTTSDLB	Distribution zone name. We recommend that you do not change this value.
TSCSMPCSI	PIE.TSO321.SMP.CSI	SMP CSI dataset name.
TSCSMPMTS	PIE.TSO321.SMPMTS	SMP MTS dataset name.
TSCSMPPTS	PIE.TSO321.SMPPTS	SMP PTS dataset name.
TSCSMPSTS	PIE.TSO321.SMPSTS	SMP STS dataset name.
TSCSMPSCDSI	PIE.TSO321.SMPSCDS	SMP SCDS dataset name.
TSCSMPTLOG	PIE.TSO321.SMPTLOG	SMP TLOG dataset name.
TSCSMPTLOGA	PIE.TSO321.SMPTLOGA	SMP TLOGA dataset name.
TSCSMPDLOG	PIE.TSO321.SMPDLOG	SMP SMPDLOG dataset name.
TSCSMPDLOGA	PIE.TSO321.SMPDLOGA	SMP SMPDLOGA dataset name.
TSCSMPGLOG	PIE.TSO321.SMPGLOG	SMP SMPGLOG dataset name.
TSCSMPGLOGA	PIE.TSO321.SMPGLOGA	SMP CSI dataset name.
TSCS1MACLIB	SYS1.MACLIB	MACLIB dataset name.
TSCS1MODGEN	SYS1.MODGEN	MODGEN dataset name.
TSCS1AMACLIB	SYS1.AMACLIB	MACLIB dataset name.
TSCS1AMODGEN	SYS1A.MODGEN	MODGEN dataset name.
TSCS1PARMLIB	SYS1.PARMLIB	PARMLIB dataset name.
TSCDINST	PIE.TSO321.DINST	Distribution INST dataset name.
TSCDCNTL	PIE.TSO321.DCNTL	Distribution CNTL dataset name.
TSCDLOAD	PIE.TSO321.DLOAD	Distribution LOAD dataset name.
TSCDHELP	PIE.TSO321.DHELP	Distribution HELP dataset name.
TSCDMACRO	PIE.TSO321.DMACRO	Distribution MACRO dataset name.
TSCDDOC	PIE.TSO321.DDOC	Distribution DOC dataset name.
TSCDPLIB	PIE.TSO321.DPLIB	Distribution PLIB dataset name.
TSCDSAMP	PIE.TSO321.DSAMP	Distribution SAMP dataset name.
TSCDPARM	PIE.TSO321.DPARM	Distribution PARM dataset name.
TSCDINFO	PIE.TSO321.DINFO	Distribution INFO dataset name.

Parameter	Default	Description
TSCDCLIST	PIE.TSO321.DCLIST	Distribution CLIST dataset name.
TSCDVIEW	PIE.TSO321.DVIEW	Distribution VIEW dataset name.
TSCDTLIB	PIE.TSO321.DTLIB	Distribution TLIB dataset name.
TSCDTMLIB	PIE.TSO321.DTMLIB	Distribution TMLIB dataset name.
TSCINST	PIE.TSO321.INST	Target INST dataset name.
TSCCNTL	PIE.TSO321.CNTL	Target CNTL dataset name.
TSCLOAD	PIE.TSO321.LOAD	Target LOAD dataset name.
TSCHELP	PIE.TSO321.HELP	Target HELP dataset name.
TSCMACRO	PIE.TSO321.MACRO	Target MACRO dataset name.
TSCDOC	PIE.TSO321.DOC	Target DOC dataset name.
TSCPLIB	PIE.TSO321.PLIB	Target PLIB dataset name.
TSCSAMP	PIE.TSO321.SAMP	Target SAMP dataset name.
TSCPARM	PIE.TSO321.PARM	Target PARM dataset name.
TSCINFO	PIE.TSO321.INFO	Target INFO dataset name.
TSCCLIST	PIE.TSO321.CLIST	Target CLIST dataset name.
TSCVIEW	PIE.TSO321.VIEW	Target VIEW dataset name.
TSCTLIB	PIE.TSO321.TLIB	Target TLIB dataset name.
TSCTMLIB	PIE.TSO321.TMLIB	Target TMLIB dataset name.

Step 3: Create the SMP/E Environment

In this step you will run four jobs to create the SMP/E environment.

If you ran the IPOUPDTE utility in “Step 2: Copy and Tailor Installation Dataset,” the jobs are ready to submit without modification. If you did not run IPOUPDTE, modify the JCL as required before submitting the jobs.

1. Submit member BUILD1 of the installation dataset. This job allocates and initializes the SMP/E CSI dataset. Before continuing, check to see that it executed with condition code 0.
2. Submit member BUILD2 of the installation dataset. This job allocates the SMP/E Global datasets. Before continuing, check to see that it executed with condition code 0.
3. Submit member BUILD3 of the installation dataset. This job allocates the SMP/E Distribution datasets. Before continuing, check to see that it executed with condition code 0.
4. Submit member BUILD4 of the installation dataset. This job updates the SMP/E CSI dataset with your PIE/TSO SMP configuration. Before continuing, check to see that it executed with condition code 0.

Step 4: Receive, Apply, and Accept Base Installation

In this step, you will run three separate jobs to receive, apply, and accept the base installation.

If you ran the IPOUPDTE utility in "Step 2: Copy and Tailor Installation Dataset," the jobs are ready to submit without modification. If you did not run IPOUPDTE, modify the JCL as required before submitting the jobs.

1. Submit member SMPREC of the installation dataset. This job will do an SMP/E RECEIVE. Before continuing, check to see that it executed with condition code 0.
2. Submit member SMPAPP of the installation dataset. This job will do an SMP/E APPLY. Before continuing, check to see that it executed with condition code 0.
3. Submit member SMPACC of the installation dataset. This job will do an SMP/E ACCEPT. Before continuing, check to see that it executed with condition code 0.

Step 5: Unload Maintenance Tape

In this step you will use IEBCOPY to unload maintenance dataset. You will use the members in this dataset to receive, apply, and accept PIE/TSO maintenance.

1. Edit the IEBCOPY job you created in “Step 1: Unload and allocate Base Install PDS,” on page 6.
2. Change that JCL to correspond with the job below.

   For ddname SYSUT1, see the tape label and change the VOL=SER parameter. Also make sure the UNIT value is correct.

   For ddname SYSUT2, change the DSN parameter. Check the UNIT and VOL=SER parameters.

   Do not change any other parameters.
3. Run the job. It must run with condition code 0.

SAMPLE IEBCOPY JCL

```
//STEP1    EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//SYSUT1   DD   DSN=RIMLIB,
//              UNIT=unit,                <== Your unit value
//              VOL=SER=PTnnnn,           <== See tape label
//              DCB=(RECFM=F,BLKSIZE=80,LRECL=80),
//              LABEL=(7,NL,EXPDT=98000),
//              DISP=SHR
//SYSUT2  DD    DSN=your.rimlib.pds.name,  <== Your dsn
//              DISP=(,CATLG,DELETE),
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160),
//              SPACE=(CYL,(1,1,47)),
//              UNIT=sysda,                <== Your unit value
//              VOL=SER=volser             <== Your volser
//SYSIN  DD     DUMMY
```

Step 6: Receive, Apply, and Accept Maintenance

In this step, you will run three separate jobs to receive, apply, and accept maintenance changes.

Adjust the JCLLIB overrides for all three jobs before running them, so that they are using the TARGET.INSTALL.PDS name that you created in "Step 2: Copy and Tailor Installation Dataset." This must be done manually, not with IPOUPDTE.

We recommend that you do not change the SOURCEID label used, because we will refer to these names as maintenance levels in the PIE/TSO SMP process.

1. Submit member SMPREC of the maintenance dataset you unloaded in the last step. This job will do an SMP/E RECEIVE. Before continuing, check to see that it executed with condition code 0.
2. Submit member SMPAPP of the maintenance dataset. This job will do an SMP/E APPLY. Before continuing, check to see that it executed with condition code 0 or 4.
3. Submit member SMPACC of the maintenance dataset. This job will do an SMP/E ACCEPT. Before continuing, check to see that it executed with condition code 0 or 4.

Step 7: Assemble PIE/TSO Members

In this step you will reassemble members from your existing PIE/TSO libraries and copy them to your new libraries.

1. Edit members PSCDFLD@ and PCSVPSWD of your existing PIE/TSO CNTL dataset.
2. Change the APF parameter to point to your new PIE/TSO target load library.
3. Run these jobs. Condition code 4 is acceptable.

Step 8: Continuing With PIE/TSO Installation

Once you have completed the preceding steps, you may continue with the regular *PIE/TSO Installation Guide*, beginning with "Step 5: Modify Your Logon Procedure," on page 10.

Migrating to MVS 5.1

To migrate PIE/TSO to MVS 5.1, perform the following.

1. Edit member PCSDFLD@ of the CNTL library. Change all occurrences of 'IEWL' to 'HEWLF064'. Review the JCL for any necessary customization, then run it.

   This job will reassemble module PCSDFLD@.

2. Repeat step 1 for member PCSVPSWD of the CNTL library. This job will reassemble module PSCVPSWD.

3. Invoke PIE/TSO under MVS 5.1.

4. Invoke MSPF. Is the correct menu displayed? If yes, your migration is complete. If no, perform the following:

   a. Edit member MULTIPRM of the SAMPLIB dataset. Find 'ISRPCP' at label ISPFNAME. Change 'ISRPCP' to 'ISPSTART'.

   This allows us to link to ISPF instead of PDF.

   b. Repeat step 1 for member MULTIPRM of the CNTL library. This job will reassemble module MULTIPRM.

   c. Refresh LLA.

   d. Reinvoke MSPF. The menu should now be correct.

PIE/TSO Installation Guide

Release 3.2.1

UNICOM
SYSTEMS, INC.

PTSOIG321-01

Printed: October 2002

This manual applies to PIE/TSO release 3.2.1 and to all subsequent releases of the product until otherwise indicated by new editions or updates to this publication.

UNICOM Systems, Inc.
Unicom Plaza - Suite 310
15535 San Fernando Mission Blvd.
Mission Hills, California 91345

Contents

Preface

Using this Guide

What is this manual about?

The PIE/TSO Installation Guide describes the installation and customization procedures for PIE/TSO and its components.

Chapter 1 details the steps required to install PIE/TSO.

Chapter 2 details MultiTSO and general PIE/TSO customization procedures.

Chapter 3 details Network Access customization procedures.

Chapter 4 details the customization procedures for MultiTask, View, Performance Group Manager, and the Logon Director.

For maintenance and operation information, see the PIE/TSO Administrator Guide.

Who is this manual for?

This manual is written for the systems programmer responsible for installing PIE/TSO.

Customer Services

Troubleshooting suggestions

We have found that many problems can be solved by looking into a few common problem areas. We suggest you look into the following.

- Have you made any recent changes to your installation, including changes to MVS, TSO, and other products? Are these changes compatible with PIE/TSO?
- Is the current release of PIE/TSO installed?
- Is your PIE/TSO password current and correctly entered?
- Are there any relevant messages on your PIE/TSO or TSO logs?

Contacting Customer Services

WWW.UNICOMSI.COM/SUPPORT

UNICOM's normal business hours are from 7:00 a.m. to 4:00 p.m. Pacific Time, Monday through Friday. However, emergency customer service is available 24 hours a day, 7 days a week.

When you call Customer Services, please make sure you have the following information available:

- related error messages
- the command or JCL you are inputting
- the dump, if one is generated

CUSTOMER SERVICES PHONE

Our phone number is (818) 838-0606.

If you call after hours our answering service will take the call. You may leave a message to be returned during normal working hours. Or, you may leave an urgent message, and a technical support representative will be paged to call you back immediately. When leaving an urgent message, be sure to leave your after-hours phone number.

If you've left an urgent message but haven't received a call back within an hour, please call back. We may be having trouble reaching you.

+ International customers, please contact your distributor for technical assistance.

CUSTOMER SERVICES FAX

Our FAX number is (818) 838-0776.

Chapter 1 Installation

This chapter details the procedures for PIE/TSO installation. It describes the minimum steps required to get PIE/TSO up and running. Following these steps will give you the basic PIE/TSO product to run and to test.

Once you have completed these procedures, go on to chapters 2 through 5 for PIE/TSO customization procedures.

Installation Overview

PIE/TSO installation will take one to two hours when done by a systems programmer with proper authorization. We recommend that you have at least one hour available before you begin the installation steps.

The installation steps are:

- Unload the first file from the product tape.
- Use members from that file to unload the rest of the tape.
- Implement the PIE/TSO library.
- Update your security system.
- Modify your TSO logon procedure.
- Install your password.
- Modify the PIE/TSO start-up procedure.
- Start PIE/TSO and test MultiTSO's basic functions.

PIE/TSO installation does not require an IPL.

Step 1: Load File One

For SMP/E installation, see instructions in the *PIE/TSO Release Notes*.

You must install all PIE/TSO datasets from the product tape, even if you are only migrating to a new release. To do this, load file 1 from tape to disk and use the members in it to load the rest of the files.

1. If you are migrating to a new release of PIE/TSO, please read the *PIE/TSO Release Notes* for special instructions before you continue.
2. Use IEBCOPY to load the first file from the tape, using the sample JCL below. Be sure to specify the correct device type in the SYSUT1 DD statement. In the SYSUT2 DD statement, supply the volume serial number and check the high level qualifier for the CNTL dataset.

   PIE/TSO members use PIE as the high level qualifier.

```
//LOADJCL  JOB  1234,'load installation dataset',
//              MSGLEVEL=(1,1)
//LOAD     EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//SYSUT1   DD   VOL=SER=PT0033,
//              UNIT=3480,          <== Correct device type
//              DSN=TSCPIE.INSTALL.CNTL,DISP=OLD,
//              LABEL=(1,SL,EXPDT=98000)
//SYSUT2   DD   UNIT=SYSALLDA,DISP=(,CATLG,DELETE)
//              DSN=pie.PTS321.CNTL,<==Correct HLQ
//              VOL=SER=volser,     <==Supply the volume
//              SPACE=(6160,(120,120,15),,,ROUND),
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSIN    DD   DUMMY
//
```

Step 2: Load the Remaining Files

In this step you will build and run a job to load the rest of the files from the tape. You may build this job automatically or manually.

To build this job automatically, you will use the INSTALLS CLIST. It will prompt you for all job variables (like job statements, qualifiers, etc.). When you have completed the questions, the INSTALLS CLIST will build a customized installation job and present it for your review. You may then modify the job if you want. It will be saved as member SINSTJCL in the CNTL dataset. The following sections detail the steps required to run the INSTALLS CLIST.

If you want to create this job manually, we recommend that you use member SINSTALL of the CNTL dataset as a model. Appendix B contains instructions for using member SINSTALL.

Choosing a library

PIE/TSO must execute from an explicitly authorized APF library. That is, the library must have an entry in an active IEAAPFxx member of SYS1.PARMLIB. If it does not, you will receive an S306-C abend when you try to start PIE/TSO.

You may install PIE/TSO modules in a link list library or an LPA library, or you may install into a new library and authorize it. If the library is not in the link list or LPA, you will need a STEPLIB DD statement in your logon procedures. If you want PIE/TSO modules in the LPA, you will have to IPL with a CLPA.

INSTALLS CLIST Worksheet

The CLIST will prompt you for the following information. If you want, you can prepare your answers in advance with this worksheet.

What is the name of the dataset you loaded file 1 into (e.g., PIE.PTS321.CNTL)?

_____

What APF library will you load PIE/TSO datasets into and what is its DISP (e.g., SYS3.PIEAUTH and SHR)? _____

What is your security system (RACF, RACR, TOPS, ACF2, NONE)? _____

What JES procedure library will you use for your PIE/TSO procedures and what is its DISP (e.g., SYS3.PROCLIB and SHR)? _____

What disk volume serial number will you use for PIE/TSO product datasets (e.g., MVS230)? _____

What esoteric unit name do you want to use for disk (e.g., SYSALLDA)? _____

What macro library are your security macros in (e.g., SYS1.MACLIB)? _____

What are the first and second level qualifiers for your PIE/TSO datasets (e.g., PIE.PTS321)? _____

If you are migrating or re-installing, do you want to overwrite old PIE/TSO datasets (YES or NO)? _____

What is the SYSOUT class (e.g., X)? _____

What esoteric unit name do you want to use for temporary disk datasets (e.g., SYSALLDA)? _____

What esoteric unit name do you want to use for tape units (e.g., 3480)? _____

What esoteric unit name do you want to use for VIO units (e.g., VIO)? _____

What job statement information do you want used (e.g., accounting information and programmer name)? _____

What special JES control statements do you want to add to the job stream (e.g., /* ROUTE statements, etc.)? _____

Running the INSTALLS CLIST

You may run the CLIST multiple times, if necessary. To cancel the CLIST enter .END at any prompt.

1. From ISPF Option 6 or TSO READY enter: EXEC 'pie.PTS321.CNTL(INSTALLS)'
   Substitute the correct high level qualifier for pie.
2. Respond to the instructions on the screen. Be sure to specify the correct security system: RACF, RACR (for RACROUTE), TOPS (for TOP SECRET), ACF2, or NONE.
3. Review the job. If you want to modify it, see Appendix B for parameter descriptions.
4. Submit your SINSTJCL job.

The installation job assembles PCSDFLD@. PCSDFLD@ refers to IBM control blocks and avoids MVS level dependencies in most load modules.

You may have several MNOTE statements during PCSDFLD@ assembly. This is normal, since IBM changes the IKJEFLWA macro in different releases of TSO/E. \logic Software provides for this in the macros used in the assembly, but it is not possible to totally eliminate the impact of IBM changes. All MNOTEs can be ignored up to TSO/E Release 2.4.1.

If you use the PIE/TSO load library as the APF authorized library, the COPYAPF step will end with a condition code of 8. That is because IEBCOPY will think that a “member select” is being attempted during a compress. If this occurs the IEBCOPY will fail with a parameter validation error before doing the compress or the copy. In the above case, 8 is an acceptable return code for this job step. PCSDFLD@ must reside in an APF authorized library or implicitly authorized library, such as link list.

Step 3: Implement the PIE/TSO Library

1. Perform one of the following steps to implement your PIE/TSO load library.
   - If you loaded PIE/TSO into a link list library, perform an LLA refresh so that MVS can find the PIE/TSO load modules. Enter: `F LLA,REFRESH`
   - You may postpone this step by adding a STEPLIB DD statement to your logon procedure. See “Step 5: Modify Your Logon Procedure,” on page 8, for details.
   - If you loaded PIE/TSO into an LPA library, IPL with a CLPA so that PIE/TSO load modules will be loaded into the LPA.

   We suggest that you postpone this step by adding a STEPLIB DD statement to your logon procedure. See “Step 5: Modify Your Logon Procedure,” on page 8, for details.
   - If you loaded PIE/TSO into new (not yet APF authorized) library, edit an IEAAPFxx member in SYS1.PARMLIB. Add an entry for the PIE/TSO load library. For example: `SYS3.PIEAUTH  MVS230`. Then IPL.

Step 4: Security Considerations

ACF2, TOP SECRET, and RACF have different requirements for PIE/TSO support. See the section that applies to the security system at your site.

ACF2

Some of the following ACF2 requirements may not apply to your installation. If you have any questions, please call UNICOM Systems Customer Services for assistance.

1. If you use command limiting, add all PIE/TSO command processors (commands that are executed as TSO commands) to the proper command tables. The current PIE/TSO command processors are:

@ALLOC	JMKEY	SCREEN
@CAN	MSPF	START
@SET	MULTISPF	STARTH
ACCESS	NOTE	STOP
CREATE	PEXEC	SWCHAR
CREATEH	PIECHECK	SWKEY
LOCK	PIEEXEC	SWITCH
HIDE	PIESTART	UNHIDE

2. If you use command limiting, add the PCSTSOLV command, even though it is not a user command. The PCSINIT procedure runs IKJEFT01 in batch and executes PCSTSOLV to determine which release of TSO is installed. PIE/TSO will not initialize if this command cannot execute.
3. Some installations have very restrictive access rules that check for the command name ISPF. If any of your rules are that restrictive, add MULTISPF and any aliases for it to the rule.

TOP SECRET

You must define PIE/TSO to TOP SECRET to allow the LOCK command to execute properly. Otherwise the LOCK command will not be allowed to validate user passwords, and locked users will be disconnected even if they enter their password correctly.

1. Using a procedure described in the *TOP SECRET Customization Guide*, add a facility named PIE and program prefix PCS to the Systems Facility Matrix and include the parameter TSOC to inform TOP SECRET that the PIE facility is TSO compatible.
2. Rename an entry in the Systems Facility Matrix to PIE. This entry may be USER0, USER1, USER2…. For example: `FAC(USER0=NAME=PIE)`
3. Add PIE to the TOP SECRET initialization procedure. The following is a sample release 4.1 entry: `FAC(PIE=ID=P,MODE=FAIL,PGM=PCS,TSOC)`
4. Allow each TSO user access to the PIE facility using the TSS ADD FAC command. PIE may be added to each user ID or to the appropriate profiles for those user IDs.

RACF

Check the conditions below and perform the required procedures, if appropriate.

- If you are controlling access to the MVS.VARY.NET resource in the OPERCMDS class, you must define the user ID PIEUSER to RACF. Then place the PIEUSER user ID in the access list for the MVS.VARY.NET resource with UPDATE authority.
- If you will use the PCSLEXIT to control user access to PIE/TSO, you must define the @PIETSO resource in the FACILITY class. Then give PIE/TSO users READ access to @PIETSO.

☞ PIE/TSO uses only the SAF (RACROUTE) interface for all security system calls.

Step 5: Modify Your Logon Procedure

In order to access PIE/TSO services, each user's logon procedure must invoke PIE/TSO's address space control program. Then whenever a user opens a session, the address space control program attaches your TSO Terminal Monitor Program (TMP) to the session, providing TSO services for the session.

For now, perform only the minimum changes to a test logon procedure. Later, when you have tested your PIE/TSO system, you can customize the procedure to perform special tasks (described in “PCSEFT01 EXEC Parameters”, beginning on page 20).

1. Make a duplicate logon procedure for testing. Authorize its use with your external security system.
2. Change the EXEC PGM=IKJEFT01 statement in your logon procedure to read

   ```
   // . . . EXEC PGM=PCSEFT01 . . .
   ```

   If you run a TMP other than IKJEFT01, please refer to “PCSEFT01 EXEC Parameters”, beginning on page 20, and “Set PIE/TSO Start-Up Options”, beginning on page 30, before continuing.
3. You may have an increase in dataset allocations because of the number of sessions you can open at the same time. To support this increase, increase the DYNAMNBR in the EXEC statement by 50.
4. If the APF library you are using for PCSEFT01 is not in the link list, supply a STEPLIB DD statement in your logon procedure and concatenate the APF library containing PCSEFT01 to any existing STEPLIB libraries currently in the test procedure. All datasets in the STEPLIB concatenation must be authorized. If any libraries in the concatenation are not APF authorized, PCSEFT01 will lose its authorization, and a system 047 abend will occur.

   This statement is only necessary until you refresh LLA (if you loaded PIE/TSO into a link list library) or IPL with a CLPA (if you loaded PIE/TSO into an LPA library).
5. If you currently use a parameter to supply an initial command to TSO during logon, remove the parameter until PIE/TSO is completely installed. If you have to test PIE/TSO with your initial command, read “PCSEFT01 EXEC Parameters”, beginning on page 20, now. If you do not modify the current parameter as described in that section, your new logon procedure will cause errors.
6. Test your new logon procedure. PIE/TSO is not yet active, so you should see no difference from standard TSO.

Example

```
//SAMPLE   PROC
//SAMPLE   EXEC PGM=PCSEFT01,DYNAMNBR=99,TIME=1440,
//STEPLIB  DD  DISP=SHR,DSN=SYS3.PTS321.LINKLIB
//SYSEXEC  DD  DSN=ISP.V3R2M0.ISPEXEC,DISP=SHR
//SYSPROC  DD  . . .
.
.
.
//SYSHELP  DD  . . .            (and other dataset allocations)
.
.
.
//SMPTABL  DD  DISP=SHR,DSN=&SYSUID..SMP.ISPTLIB
//ISPGLIB  DD  DSN=ISP.V3R2M0.ISPGENU,DISP=SHR
//SYSPRINT DD  TERM=TS,SYSOUT=*
//SYSTERM  DD  TERM=TS,SYSOUT=*
//SYSIN    DD  TERM=TS
//*
```

Step 6: Apply Your PIE/TSO Password

The SESSIONS member of the PARMLIB dataset contains PIE/TSO start-up options. These options specify your password, locate files, and tailor PIE/TSO processing.

In this step you will update the password options and identify the help dataset. Later, when you have tested your PIE/TSO system, you can customize the member to meet your installation's requirements. (Customizing the SESSIONS member is described in "Set PIE/TSO Start-Up Options", beginning on page 30.)

1. Edit member SESSIONS of the PARMLIB dataset.
2. Update the following start-up options using the information in the letter sent with your PIE/TSO product tape.

   COUSER — Identify the number of concurrent users your license agreement allows. Specify 0 if there is no limit.

   EXPDT — Identify your PIE/TSO expiration date.

   xxxCMP — Identify which components are part of your license agreement. If a component is part of your license agreement, code YES. If a component is not authorized in your license, set the parameter to NO. Any variation will invalidate your passwords, and PIE/TSO will not work. Component parameters are:

   FILCMP for File Access
   LOGCMP for Logon Director
   MTTCMP for MultiTask
   NETCMP for Network Access
   PGMCMP for Performance Group Manager
   SESCMP for MultiTSO
   VIECMP for View

   CPUID — Each licensed CPU ID is paired with a password. Specify your CPU ID and password pairs. Code one pair per line. You may use as many lines as necessary. Example:

   ```
   CPUID
   000000000000=XXXXXXXXXXX
   111111111111=YYYYYYYYYYY
   ```

   CPUID must be the last option in the SESSIONS dataset.
3. If you want full-screen HELP, specify the HELP dataset name in the HELPDS option. The default is PIE.PTS321.HELP.

Step 7: Modify the PIE/TSO Start-Up Procedure

You are now ready to modify the PIE/TSO start-up procedure.

1. Edit member PCSINIT from the PIE/TSO PROCLIB.
2. Modify the following parameters:

   APF Identify the PIE/TSO load library.

   DUMP Specify a SYSOUT class for dumps generated if there is a problem with PCSINIT. The default is A.

   PARMDS Identify the dataset that will contain the PIE/TSO start-up options. The default is PIE.PTS321.PARMLIB.

   SOUT Specify a SYSOUT class for PCSINIT output. The default is *.

Step 8: Start PIE/TSO for the First Time

1. From an operator console or equivalent, enter S `PCSINIT,ACT=START`.

   PCSINIT will execute and issue message PCS041I. If you receive a different message, consult the *PIE/TSO Messages and Codes* manual for an explanation and response.

   If PCSINIT abends with a System 306-C abend, the SYSLIB dataset is not APF authorized. The PIE/TSO modules must be copied to an APF authorized library. Modify the SYSLIB DD statement in the PCSINIT procedure accordingly, before rerunning PCSINIT. If PCSINIT abends with a System 106-F abend, your APF authorized library is probably in the link list and has gone into extents. Correct the problem before proceeding.
2. Once you receive message PCS041I, PIE/TSO is active. To begin using it, log on to TSO using your new logon procedure.

   If PCSEFT01 abends with a System 047 completion code during logon, it is not an authorized program running from an authorized library.
3. After logging on, you may press PF1 to get online help or go through the tutorial in the *User Guide or Administrator Guide* to become familiar with PIE/TSO.
4. Test PIE/TSO by following the instructions in the *User Guide* or *Administrator Guide* tutorials for MultiTSO. Test only starting sessions, running applications in those sessions, and switching, since only the basic functions of the MultiTSO component are available at this point. You cannot yet run ISPF in multiple sessions. This and other functions will become available as you follow the instructions in Chapters 2 through 5.

   As you test PIE/TSO, you may need to start and stop it frequently. See “Starting and Stopping PIE/TSO” in the *PIE/TSO Administrator Guide*, for more information.

Chapter 2
Customizing PIE/TSO

Not all PIE/TSO customization procedures are necessary. Review each section in this chapter and determine which procedures you need to perform.

The chapter itself is organized into two sections. The first section contains customization procedures that are generally recommended. The second section contains less essential procedures.

When you are finished with the procedures in this chapter, go on to chapters 3 and 4 to customize PIE/TSO for Network Access, MultiTask, PGM, View, and Logon Director. Then go on to chapter 5 to put your changes into effect and put PIE/TSO into production.

Recommended Customization

Step 1: Running Multiple Copies of ISPF

Many, many TSO applications use ISPF. To get the most out of your PIE/TSO session capabilities, you need to be able to run ISPF in multiple sessions at the same time. The Multiple ISPF feature, MULTISPF, allows you to do so.

MULTISPF consists of two load modules: MULTIPRM and MULTISPF. MULTIPRM is a table of constants used by MULTISPF to allocate ISPF profiles and other ISPF datasets. MULTISPF (or MSPF, its alias) is the TSO command that

- loads MULTIPRM
- allocates ISPF datasets
- calls ISRPCP, ISPF's main entry point

MSPF can also execute PDF instead of ISPF. When ISPF or PDF exits, all DD names allocated by MSPF are freed.

MSPF only executes ISPF load modules. It does not replace them. All the features of these products are still available.

To install MSPF, perform the following steps.

1. Edit member MULTIPRM of the SAMPLIB dataset.

   Modify the profile allocation. Specify your ISPF profile dataset name in the DSNAME parameter. See “DSNAME” on page 19 for instructions.

   Use member MULTIPRM of the CNTL dataset to reassemble.

2. If you run ISPF version 2 or above, install the ISPF user exit 16 to avoid LIST and LOG errors. To do so, edit and run members PCSESA16, ISPDFLTS, and ISPXDT of the PIE/TSO CNTL library.

Making Other Changes to MULTIPRM

Usually no changes are required to MULTIPRM other than the DSNAME change, specified above. If you do want to make other changes, see IBM's *TSO/E Programming Services* manual, the Dynamic Allocation Interface Routine (DAIR) chapter, or call UNICOM Systems Customer Services. If you make changes, keep the following in mind.

MULTIPRM is set up to dynamically allocate your ISPF profile dataset. The dataset name includes the corresponding session's one character code. (It uses A and B, never 10 or 11.) So the same profile will always be allocated to the same session. If a profile is not available for an ISPF session, MSPF will allocate a new empty ISPPROF.

You can also use MULTIPRM to allocate other ISPF datasets, if you want.

The MULTIPRM parameter `SUBSYS` is an assembler constant specifying JES2 or JES3 as the job entry subsystem. JES2 is the default. If JES3 is the primary subsystem, the SYSOUT datasets must be allocated differently than under JES2. From the users' point of view, MSPF works the same way under JES2 or JES3.

`ISPFNAME` defines the module MSPF calls after the profiles are allocated. The default is ISRPCP for PDF. If you would rather start at ISPF, change this name to ISPICP.

⚠ WARNING

Do not code MSPF for this parameter. If you do, MSPF will link to itself and go into a loop.

The DSALLOC macro specifies the attributes of the ISPF profile dataset. DSALLOC parameters follow.

DSALLOC FORMAT

```
label DSALLOC USERID={YES|NO},
              PREFIX={YES|NO},
              SUFFIX={YES|NO},
              DDNAME='ddname',
              DSNAME='dsn',offset,
              OLD=dair-name,
              ATTR=dair-name,
              CREATE=dair-name,
              UNIT=unitname,
              SYSOUT=class,
              FREE=(dair-name,dair-name,dair-name)
```

Parameter	Valid	Default
label	1 to 8 characters	None
USERID	YES or NO	YES
PREFIX	YES or NO	YES
SUFFIX	YES or NO	NO
DDNAME	1 to 8 characters	None
DSNAME	1 to 43 characters for “dsn,” 2 to 44 for “offset”	None
OLD	DAIR label	0
ATTR	DAIR label	0
CREATE	DAIR label	0
UNIT	valid unit name	None
SYSOUT	valid class	None
FREE	DAIR labels	None

label You may code any legal label. The default macros use the generic name of the dataset being allocated. PROFILE is the label on the DSALLOC macro allocating the profile dataset.

USERID Specify whether the TSO user ID (&SYSUID) is to be part of the dataset name.

PREFIX Specify whether the TSO prefix (&SYSPREF) is to be part of the dataset name.

If both USERID and PREFIX are coded YES, and the two values are different, the generated dataset name will be in the format &SYSPREF.&SYSUID.DSN. If the values are equal, the generated dataset name will be in the format &SYSPREF.DSN.

SUFFIX Specify whether &SYSUID should be used as a suffix to the generated dataset name.

DDNAME Specify the ddname for this dataset. Code the standard ddname associated with this ISPF dataset.

You may concatenate up to four datasets with DSALLOC macros just as you concatenate with DD

statements. Omit the DDNAME operand on the second or subsequent DSALLOC macros you are concatenating. The DDNAME coded on the previous DSALLOC will be used.

DSNAME Specify the dataset name for the referenced DDNAME. It should be the same name your installation uses for native ISPF datasets. If you use the variable &&SMFID, MSPF will substitute the SMF ID of the system in use.

The offset value is used to make the dataset names unique across sessions. Specify the position of the dataset name to replace. Zero is not a valid offset. You may leave the offset blank, and MSPF will put the session code at the end of the base dataset name. For a complete description of this concept, see “Step 1: Modify the PCSDDNTB Module,” on page 22.

To dummy out a dataset, code ('NULLFILE',0) for DSNAME. Also code PREFIX=NO and USERID=NO on this macro.

OLD Specify existing dataset disposition. Use a valid DAIR control block.

ATTR Specify DCB attributes for this dataset. Use a valid DAIR control block.

CREATE Refer forward to a DSALLOC label for a new dataset's space allocation. It is ignored for existing datasets.

UNIT Specify the unit name (SYSDA, 3380, VIO, 3390, etc.) for new datasets. By default, new datasets are allocated on the unitname specified in SYS1.UADS or the UADS equivalent.

SYSOUT Specify an output class. Code this parameter instead of the DSNAME parameter if you want it to go to a JES SYSOUT dataset instead of a disk dataset. The SYSOUT parameter is primarily for the LIST and LOG datasets.

FREE Specify how the dataset will be deallocated when the session is terminated normally. Normal termination means the user exits MSPF or ISPF cleanly with PF3 or =X. The three values reference DAIR control blocks coded in MULTIPRM. FREEOLD will keep a dataset. FREETEMP will delete it.

Step 2: PCSEFT01 EXEC Parameters

PCSEFT01 is PIE/TSO's address space control program. When a user opens a new session, the address space control program attaches your TSO Terminal Monitor Program (TMP) to the session, to provide TSO services.

PCSEFT01 must be executed in each user's logon procedure before the user can access PIE/TSO services. In the logon procedure, you may use the PARM parameter of the EXEC statement to pass parameters to PCSEFT01. Use these parameters to:

- substitute a different Terminal Monitor Program (TMP) for TSO's default, IKJEFT01
- open a PIE/TSO session and/or execute an initial TSO command or CLIST/REXX EXEC

If you do not want to change your TMP or execute initial commands, you may omit the PARM statement.

FORMAT

The format of the PARM statement is:

```
PARM='[tmp [sectmp]/] [pietso-command [n] [tso-command]]'
```

Separate parameters with spaces. Enclose all parameters in one pair of quotes.

CHANGING THE TMP

The first set of parameters (tmp sectmp) allow you to substitute your TMP for TSO's default IKJEFT01. These parameters are required if you execute a TMP other than IKJEFT01 in your logon procedure.

tmp Identify the TMP to be used for PIE/TSO session 0. It will override the TMP start-up option in the PARMLIB(SESSIONS) member. (For information on the TMP start-up option, see “Step 5: Set PIE/TSO Start-Up Options,” on page 30.)

sectmp Identify the TMP to be used for all other sessions. This parameter is optional, even if you code tmp. However, if you code sectmp you must also code tmp. Sectmp overrides the SECTMP start-up option.

Always place a slash mark after your TMP parameters—even if you don't code any parameters after them.

EXAMPLES

To change the TMP to MYTMP for all sessions, enter: `PARM='MYTMP MYTMP/'`

To change the TMP for session 0, enter: `PARM='ADFMDF03/'`

To change the secondary TMP only, enter `PARM='IKJEFT01 MYTMP/'`. Notice that both TMPs are specified.

EXECUTING AN INITIAL COMMAND, CLIST, OR REXX EXEC

The second set of parameters (`pietso-command tso-command`) sets up PIE/TSO sessions and executes an initial TSO command at logon.

pietso-command Specify a PIE/TSO command and its parameters.

It makes the most sense to start a PIE/TSO session with this parameter—that way, you can run an initial TSO command in that session. So in the following examples, we use the START command. See the *PIE/TSO Command Reference* for information on other PIE/TSO commands.

n Specify a particular session code to open, if you want. For example, specify `START 1` to open session 1.

tso-command Specify a TSO command to execute in the session opened. You can specify an initial CLIST or REXX EXEC, if you want. You can pattern your initial CLIST after our sample CLIST, PIEINIT1, which executes many PIE/TSO commands. For more information on initial CLISTs and REXX EXECs, see “Building Multiple Sessions Automatically,” on page 35 of the *PIE/TSO Administrator Guide*.

EXAMPLES

With `PARM='START'`, PIE/TSO opens a session and sends the user to TSO READY mode when he or she logs on. This skips the initial PIE/TSO menu.

With `PARM='START 1 ISPF'`, PIE/TSO opens session 1 and starts ISPF in it automatically.

With `PARM='START %INIT'`, PIE/TSO opens a session and executes a CLIST named INIT.

Optional Customization

Step 1: Modify the PCSDDNTB Module

At times, you will want to run the same application in multiple sessions. As you would expect, each session will need its own, unique datasets—for work files and other uses. Because the application uses the same ddnames in every session, MVS would ordinarily prevent new allocations in new sessions. Let's say, for example, you try to run a certain CLIST in two sessions at the same time, and the CLIST uses the ddname INPUT. In session 1, the CLIST allocates INPUT to dataset A.B.C. In session 2, it needs to allocate INPUT to dataset X.Y.Z. But since INPUT is already allocated to A.B.C, MVS won't allow the CLIST to use that ddname unless you free it from the first allocation. So the CLIST will fail in session 2.

PIE/TSO solves this problem by changing the ddname for specified datasets when you switch from session to session. When you switch out of a session, PIE/TSO changes the session's ddnames—so the original ddnames are free to allocate in a new session. When you switch back to a session, PIE/TSO corrects the ddnames. This approach allows you to run multiple copies of ISPF, SAS, Panvalet, or any other command processor with fixed ddnames.

For example, our CLIST from above uses the ddname INPUT. When you switch from session 1, PIE/TSO will substitute the session code (1) for any letter in the ddname. (You specify which letter.) Let's say it changes INPUT to IN1UT. Now INPUT is free. So when the CLIST goes to allocate INPUT in session 2, there is no conflict. When you switch out of session 2, PIE/TSO changes that INPUT to IN2UT. When you switch back into session 1, PIE/TSO will change IN1UT back to INPUT, and the ddname is back to its original allocation.

You specify which ddnames must be changed in our PCSDDNTB table.

Do not use the PCSDDNTB table to allow multiple commands write access to the same dataset with DISP=SHR. In addition, if a CLIST or command processor always allocates the same output dataset and ddname, do not code that ddname in PCSDDNTB. PCSDDNTB is designed for commands or CLISTs that need to allocate one ddname to different output datasets.

☞ If you add a ddname to PCSDDNTB, and the ddname is allocated in the logon procedure, the dataset allocated will belong to the first session that is opened. If any other session needs a dataset with that ddname, that dataset must be allocated separately. PIE/TSO will not allocate the dataset for you. If you like, you can pre-allocate files for all the ddnames PIE/TSO can use. For example, pre-allocate datasets for IN1UT, IN2UT, IN3UT, etc.

To see how your allocations are stored, execute the `LISTA ST SYS` command. This TSO command lists all ddnames (and datasets) currently allocated in the address space. If ISPF and MULTISPF are both active in the address space, the LISTA command will show two sets of ISPF datasets allocated with unique ddnames.

The standard MVS enqueue mechanism for datasets is not affected by PCSDDNTB. Any security system such as RACF, TOP SECRET, or ACF2 is still in control of dataset allocations. No user commands need to be modified to use PCSDDNTB.

To change the PCSDDNTB table, perform the following procedure.

1. Edit member PCSDDNTB of the SAMPLIB dataset. See the format description below.
2. Specify ddnames with @SDDNAME macros.

   If the command executes under ISPF and you need to pre-allocate the datasets, also add the

ddname and dataset name to the MULTIPRM table. (See “Running Multiple Copies of ISPF,” beginning on page 16.)

If you will allocate the datasets in a CLIST or EXEC, you may add the ddnames to the PCSDDNTB module only.

In either case, you must add the ddnames to PCSDDNTB.

3. Assemble your new table with member PCSDDNTB of the CNTL dataset.

   You may create a separate ddname table for specific users and/or terminals. To do so, create a separate load module. Use PCSDDNT2 of the CNTL dataset to assemble it. Be sure to specify the alternate table name in the MEMBER parameter. Specify the member name in the DDNTABL parameter of the PCSUSERS table. See “Step 6: Overriding Defaults for Users and Terminals,” on page 42, for more information.

FORMAT

```
@SDDNAME ddname,OFFSET=n
```

Parameter	Valid	Default
ddname	1 to 8 characters	None
OFFSET	1 to 7	None

ddname Specify the ddname to be unique in each session. You may code either the complete name or a generic name. To code a specific name, include it in quotes and if the name is less than eight characters, space fill to eight characters. Code ddnames in order from the most specific to most generic. Code names with wildcards last. Use caution in coding generic ddnames. There may be more matches than you expect.

To code a generic name, include wildcards and/or specify only the first part of the name. Do not use quotes. A question mark (?) indicates any character in that position. A pound sign (#) indicates any numeric in that position.

So, to match the exact name SYSUT, specify: `'SYSUT    '`

To match any name beginning with SYSUT, specify: `SYSUT`

To match only SYSUT names ending in a numeric, specify: `SYSUT#`

To match ddnames with SYS as the third, fourth, and fifth letters, specify: `??SYS`

OFFSET Specify which position in the ddname PIE/TSO should change. When a user switches out of a session using this ddname, PIE/TSO will substitute the session code for the character in this position. The session code identifies which session this dataset belongs to. PIE/TSO restores the proper ddname when the user switches back to this session.

OFFSET=0 corresponds to the first character in the ddname. OFFSET=7 refers to the eighth character. Do not code an OFFSET value greater than the length of the ddname.

Make certain that the offset replacement will always change the name of the dataset. For instance, if the dataset name is KEJ723, you don't want to assign offsets to the positions containing 7, 2, or 3. If you set the offset to 3, the “7” position, and you ran the application in session 7, there would be no name change. Note that sessions 10 and 11 will be identified with the letters A and B.

EXAMPLE 1

In this example, we are adding the ddname INPUT to the table. The OFFSET=2 parameter tells PIE/TSO to overlay the character “P” in INPUT with the code of the session running the application. This will occur when you switch out of the session (e.g., IN3UT). When you switch back into the session, PIE/TSO will restore the ddname to INPUT.

```
@SDDNAME INPUT,OFFSET=2
```

EXAMPLE 2

This example is part of the default table, from member PCSDDNTB of the SAMPLIB dataset.

```
*        ALLOW MULTIPLE ISPFS
*
         @SDDNAME ISPPROF,OFFSET=2
         @SDDNAME ISPLOG,OFFSET=2
         @SDDNAME ISPLIST,OFFSET=2
         @SDDNAME ISPCTL,OFFSET=2   ISPCTL0/ISPCTL1/ISPTCL2/ETC.
         @SDDNAME ISPLST,OFFSET=2   ISPLST0/ISPLST1/ISPLST2/ETC.
         @SDDNAME ISPWRK,OFFSET=2   ISPWRK0/ISPWRK1/ISPWRK2/ETC.
*        @SDDNAME 'SYSPRINT',OFFSET=2   MULTIPLE SYSPRINTS
*        @SDDNAME 'SYSIN   ',OFFSET=2   MULTIPLE SYSINS
         @SDDNAME ISP#####,OFFSET=2
*
*        ALLOW MULTIPLE INFO/MVS
*
         @SDDNAME VSAM,OFFSET=2     VSAM/VSAM1/VSAM2/VSAM4 .   . .
*
*        ALLOW MULTIPLE QUEUE
*
         @SDDNAME HASPCKPT,OFFSET=2
         @SDDNAME HASPACE#,OFFSET=2
*
*        ALLOW MULTIPLE PANVALET
*
         @SDDNAME PAN,OFFSET=2
```

Step 2: Move Load Modules to SYS1.LPALIB

We recommend that you move some of the PIE/TSO load modules to an LPALST library to improve performance. You may move the modules to the LPALST library or a load library in the MLPA list. By doing so, you will save memory in users' address spaces.

All PIE/TSO load modules are reentrant and refreshable. So they can all reside in the LPA. However, some modules provide more benefit in the LPA than others. Those that provide the most benefit are listed below—in order of benefit with approximate size.

PIE/TSO modules

- PCSEFT01: 27K
- PCSSTAT: 27K
- PCSSRMIO: 1K (TSOMON=YES before R 5.0)
- PCSSRMI: 1K (TSOMON=YES R 5.0 and higher)
- MULTISPF (MSPF): 6K

Network Access modules

- PNATL01S: 26K
- PNAVL01: 38K
- PNATL00 (alias ACCESS): 2K
- PNASTAT1: 3K
- PNASTAT2: 15K
- PNATVINI: 1K

1. Use members LPACOPY1 (for PIE/TSO) and LPACOPY2 (for Network Access) of the CNTL dataset to move and copy these modules.

Step 3: Implement Security Features

LOCK

The LOCK command prevents unauthorized use of a PIE/TSO terminal. When the terminal is locked, prevents anybody from entering commands on a locked terminal until a password has been entered. PIE/TSO unlocks the terminal after the user enters their password correctly. When the terminal is unlocked, PIE/TSO restores all sessions as they were before the lock.

1. Use any of the following methods to invoke LOCK.
   - Invoke the LOCK command manually from the PIE/TSO menu.
   - Use AutoLock. AutoLock invokes LOCK whenever a user is inactive (does not press ENTER or function key for a period of time). Specify the period and activate AutoLock with the LOCK start-up option. To activate AutoLock:

     a. Edit your PARMLIB(SESSIONS) dataset.

     b. Code the `LOCK`, `LINPUT`, `LKIGN`, and `MAXTRY` options. See "Step 5: Set PIE/TSO Start-Up Options," on page 30, for information on these parameters.
   - Use IEFUTL to lock inactive users after a short time. Locking inactive users instead of canceling them minimizes user disruption.

You may use our sample IEFUTL to modify your own exit. To view it, edit member PCSUTL of the SAMPLIB dataset. If users are executing under PIE/TSO, PCSUTL locks their terminal instead of canceling them. If users become inactive after normal working hours, PCSUTL cancels them immediately.

PCSUTL uses a macro supplied by UNICOM Systems to determine whether users are executing under PIE/TSO. The referenced macro is in the PIE/TSO MACLIB.

CONTINUOUS JOB STEP WAIT TIME

Another specialized IEFUTL exit disconnects PIE/TSO users instead of canceling them when the continuous job step wait time limit is exceeded. To implement this exit, perform the following steps.

1. If you have an existing IEFUTL exit, integrate the code from our exit into your existing exit. The source is in member PCSUTL2 of the PIE/TSO SAMPLIB dataset.
2. Assemble and link PCSUTL2 or the modified version of your old exit. You may use the sample JCL in member PCSUTL2 of the PIE/TSO CNTL dataset.
3. Edit your TSOKEY00 member and change the RECONLIM parameter to allow your users to be disconnected for an extended period of time. The value you specify is in minutes. Our sample TSOKEY00 allows users to be disconnected for eight hours.

   ```
   USERMAX=500,
   RECONLIM=480,
   BUFRSIZE=2048,
   HIBFREXT=48000,
   LOBFREXT=24000,
   CHNLEN=4,
   CONFTXT=NO,
   MODESW=YES,
   SCRSIZE=1920
   ```

4. Add or update an IEALPAxx member in SYS1.PARMLIB so that it refers to the new IEFUTL in the correct link list library. For ESA version 4, your entries would look like the following:

   ```
   INCLUDE LIBRARY(IPO4.LINKLIB)
   MODULES(IEFUTL)
   ```

5. IPL with an MLPA that includes your updated or new IEALPAxx member or IPL with a CLPA. (We suggest that you postpone this IPL until you are finished with PIE/TSO customization.)

Step 4: Issuing PIE/TSO Commands from ISPF Panels

If you like, your users can execute PIE/TSO commands from ISPF menus. For instance, they can open a new session or switch to the PIE/TSO menu. You may include any PIE/TSO command in an ISPF menu panel. (You can also use PIE/TSO PF keys to perform these functions or you can execute them from the command line.)

We've provided two sample panels, ISP@PRIM and ISR@PRIM, in the distribution panel library. These panels allow selection numbers P8, P9, and PA to start new sessions. Use these panels only as samples. Do not allocate our sample library in your logon procedure.

The example below allows the user to switch to the PIE/TSO menu with the S option

```
%------------ SPF-MVS PRIMARY OPTION MENU ---------
%SELECT OPTION ===>_OPT     +
%
%  0 +ISPF PARMS   - Specify terminal and user parameters
%  1 +BROWSE       - Display source data or output listings
%  2 +EDIT         - Create or change source data
%  3 +UTILITIES    - Perform utility functions
%  4 +FOREGROUND   - Invoke language processors in foreground
%  5 +BATCH        - Submit job for language processing
%  6 +COMMAND      - Enter TSO command, CLIST, or REXX EXEC
%  7 +DIALOG TEST  - Perform dialog testing
%  8 +LM UTILITIES - Perform library utility functions
%  9 +IBM PRODUCTS - Additional development products
%  S +SWITCH       - Switch to PIE/TSO menu
%  X +EXIT         - Terminate ISPF using log and list defaults
%
+Enter%END+command to terminate ISPF.
)INIT
  .HELP = ISR00003
  &ZPRIM = YES        /* ALWAYS A PRIMARY OPTION MENU    */
  &ZHTOP = ISR00003   /* TUTORIAL TABLE OF CONTENTS      */
  &ZHINDEX = ISR91000 /* TUTORIAL INDEX - 1ST PAGE       */
  &ZSCLMPRJ = &Z      /* TUTORIAL INDEX - 1ST PAGE   @L1A*/
  VPUT (ZHTOP,ZHINDEX,ZSCLMPRJ) PROFILE /*          @L1C*/
)PROC
&ZQ = &Z
IF (&ZCMD ^='')
  &ZQ = TRUNC(&ZCMD,'.')
  IF (&ZQ = '')
    .MSG = ISRU000
&ZSEL = TRANS( &ZQ
   0,'PANEL(ISPOPTA)'
   1,'PGM(ISRBRO) PARM(ISRBRO01)'
   2,'PGM(ISREDIT) PARM(P,ISREDM01)'
   3,'PANEL(ISRUTIL)'
   4,'PANEL(ISRFPA)'
   5,'PGM(ISRJB1) PARM(ISR) NOCHECK'
   6,'PGM(ISRPTC)'
   7,'PGM(ISPYXDR) PARM(ISR) NOCHECK'
   8,'PANEL(ISRLPRIM)'
```

```
   9,'PANEL(ISRDIIS)'
   S,'CMD(SWITCH)'
  ' ',' '
   X,'EXIT'
   *,'?' )
  &ZTRAIL = .TRAIL
)END
```

Step 5: Set PIE/TSO Start-Up Options

The SESSIONS member of the PARMLIB dataset contains PIE/TSO start-up options. These options specify your password, identify files, and tailor PIE/TSO processing.

The PCSINIT procedure reads the SESSIONS member. Be sure to identify the correct dataset in PCSINIT's PARMDS parameter.

You can override many start-up options for particular users or terminals in the PCSUSERS table. See "Step 6: Overriding Defaults for Users and Terminals," on page 42, for details.

SYNTAX

Each statement in the SESSIONS member must be an 80 character statement, beginning in column one. Sequence numbers are not allowed.

Place an asterisk (*) in column one to code a comment statement. You may place a comment within a statement if you separate the parameter and value from the comments with one or more spaces.

You may not code continuations.

FORMAT

```
COUSER=n
EXPDT=julian-date
FILCMP={YES|NO}
LOGCMP={YES|NO}
MTTCMP={YES|NO}
NETCMP={YES|NO}
PGMCMP={YES|NO}
SESCMP={YES|NO}
VIECMP={YES|NO}
[ABBREV=ON|OFF]
[ACCESS={CANCEL|NOCANCEL}]
[AE=ON|OFF]
[APF={SWITCH|NOSWITCH}]
[ASYS={YES|NO}]
[BGPGN=n]
[CMDTXT='text']
[DCMD=command]
[FCMD=command]
[FENCE=nnn]
[FGPGN=n]
[HEADER=ON|OFF]
[HDRTXT='text']
[HELPDD=ddname]
[HELPDS=dataset.name]
[HELPLN=nnnn]
[IMPTSO=ON|OFF]
[INFODD=ddname]
```

```
[INFODS=dataset.name]
[INFWIN=ON|OFF]
[INTCMD={YES|NO}]
[JMPKEY=key|OFF]
[KEYS=ON|OFF]
[LINPUT={PROCESS|DISCARD}]
[LKIGN=xxxx]
[LOCK=nnnn]
[LOGDIR={YES|NO}]
[LOGOFF={YES|NO}]
[MAXSES=nn]
[MAXTRY=nnnn]
[MENU=ON|OFF]
[MKEYS=MKEYS|SKEYS]
[MTASK={YES|NO}]
[OKSWAP={YES|NO}]
[PASENC={YES|NO}]
[PFSHOW=ON|OFF]
[PGM={YES|NO}]
[PIE={YES|NO}]
[PREFIX=x]
[SCREEN={SAVE|NOSAVE}]
[SECTMP=tmp-name]
[SESATT=ON|OFF]
[SMFRTY=nnn]
[SWAP={YES|NO}]
[SWCHAR=x]
[SWEXIT={YES|NO}]
[SWKEY=key]
[SYSKEY={SWITCH|NOSWITCH}]
[TITLE1='text']
[TITLE2='text']
[TMP=tmp-name]
[TSOMON={YES|NO}]
[UCLASS={01|015}]
[UPCASE={YES|NO}]
[VIEW={YES|NO}]
[VIEWDD=ddname]
[VIEWDS=dataset.name]
[VSTOR=CHECK|NOCHECK]
CPUIDS
cpuid=password
```

Parameter	Valid	Default
COUSER	0 to 99	0
EXPDT	Julian date	None
FILCMP	YES or NO	NO
LOGCMP	YES or NO	NO
MTTCMP	YES or NO	NO
NETCMP	YES or NO	NO
PGMCMP	YES or NO	NO
SESCMP	YES or NO	NO
VIECMP	YES or NO	NO
CPUIDS	CPU ID, 11 character password	None
ABBREV	ON or OFF	OFF
ACCESS	CANCEL or NOCANCEL	CANCEL
AE	ON or OFF	OFF
APF	SWITCH or NOSWITCH	SWITCH
ASYS	YES or NO	YES
BGPGN	1 to 999	None
CMDTXT	1 to 12 characters in quotes	Command ===>
DCMD	any TSO command	None
FCMD	any TSO command	None
FENCE	0 to 2048	256
FGPGN	1 to 999	None
HEADER	ON or OFF	ON
HDRTXT	1 to 64 characters in quotes	See PIE/TSO menu
HELPDD	1 to 8 characters	PIEHELP
HELPDS	1 to 44 characters	PIE.PTS321.HELP
HELPLN	200 to 9999	400
IMPTSO	ON or OFF	OFF
INFODD	1 to 44 characters	PIEINFO

Parameter	Valid	Default
INFODS	1 to 44 characters	PIE.PTS321.INFO
INTCMD	YES or NO	YES
INFWIN	ON or OFF	ON
JMPKEY	PF1 to PF24 or 1 to 24 or OFF	OFF
KEYS	ON or OFF	OFF
LINPUT	PROCESS or DISCARD	PROCESS
LKIGN	4 characters	No mask
LOCK	0 to 9999	0
LOGDIR	YES or NO	YES
LOGOFF	YES or NO	NO
MAXSES	1 to 12	12
MAXTRY	0 to 9999	5
MENU	ON or OFF	ON
MKEYS	MKEYS or SKEYS	MKEYS
MTASK	YES or NO	NO
OKSWAP	YES or NO	YES
PASENC	YES or NO	NO
PFSHOW	ON or OFF	ON
PGM	YES or NO	YES
PIE	YES or NO	YES
PREFIX	1 character	@
SCREEN	SAVE or NOSAVE	SAVE
SECTMP	TMP name	IKJEFT01
SESATT	ON or OFF	ON
SWAP	YES or NO	YES
SMFRTY	128 or 255 or NONE	NONE
SWCHAR	1 non-alphanumeric character	\
SWEXIT	YES or NO	NO

Parameter	Valid	Default
SWKEY	ENTER or PFn	ENTER
SWSCAN	1 to 256	32
SYSKEY	SWITCH or NOSWITCH	SWITCH
TITLE1	1 to 70 characters in quotes	See PIE/TSO menu
TITLE2	1 to 9 characters in quotes	See PIE/TSO menu
TMP	TMP name	IKJEFT01
TSOMON	YES or NO	NO
UCLASS	01 or 015	01
UPCASE	YES or NO	NO
VIEW	YES or NO	YES
VIEWDD	1 to 8 characters	PIEVIEW
VIEWDS	1 to 44 characters	PIE.PTS321.VIEW
VSTOR	CHECK or NOCHECK	NOCHECK

THE OPTIONS

The values in the distributed SESSIONS member represent the recommended values. However the values for EXPDT, CPUID, password, COUSER, and all xxxCMP parameters vary from installation to installation. The SESSIONS member is listed in the example on page 40.

For COUSER, EXPDT, all xxxCMP values, and CPUIDS, provide the values specified in the letter sent with your product tape.

COUSER Specify the license limit on concurrent PIE/TSO users. Zero specifies that there is no limit.

EXPDT Specify the license expiration date.

FILCMP Specify whether File Access is part of your license agreement.

LOGCMP Specify whether the Logon Director is part of your license agreement.

MTTCMP Specify whether MultiTask is part of your license agreement.

NETCMP Specify whether Network Access is part of your license agreement.

PGMCMP Specify whether the Performance Group Manager is part of your license agreement.

SESCMP Specify whether PIE/TSO is part of your license agreement.

VIECMP Specify whether View is part of your license agreement.

CPUIDS Specify the CPUs on which you are licensed to run PIE/TSO. Pair each CPU ID with its password. Begin the pairs on the next line after the CPUIDS parameter, which must be the last parameter in the SESSIONS member. Place only one pair on a line. You may code as many lines as you need.

⚠ WARNING

PIE/TSO becomes unusable if you change either the CPU IDs or passwords without authorization. Contact UNICOM Systems if you ever need to change a CPU ID, either temporarily or permanently.

You may set the values for the following parameters as you want.

ABBREV Specify the default for using shortened names to identify a session (for example, ED to identify the EDIT session). If you specify ON, users will be able to identify sessions with short names by default. If you specify OFF, users will not be able to do so unless they execute the SET ABBREV command.

ACCESS Specify whether users may cancel Network Access sessions from the PIE/TSO menu or whether they must log off the VTAM application to end the session.

AE Specify the default for auto-end attribute. If set to ON, PIE/TSO automatically ends the session if you log off the application running in it or if the application abends. If set to OFF, PIE/TSO remains at TSO ready when the application ends. This default can be changed with the SET AE command.

APF Specify whether users can switch out of a session running an APF authorized application.

ASYS Specify whether users can switch out of a session that is processing a task. If YES, users can turn asynchronous switching on in their environment with the SET ASYS command. When asynchronous switching is on, SNA terminal users can switch from a running session to the PIE/TSO menu with the ATTN key, and non-SNA terminal users can switch with RESET and PA1.

ASYS must be set to YES to use MultiTask. If ASYS is set to NO, users cannot use asynchronous switching.

BGPGN Specify a special performance group for the MultiTask background session. If you don't code BGPGN, the background session will use the performance group the user logged on with or the performance group assigned to the application with the PGM component.

CMDTXT Change the text of the command prompt on the PIE/TSO menu. The text can be up to 12 characters long. It must be enclosed in quotes.

DCMD Specify the default initial command for all new sessions. You can specify any TSO command and its parameters. For example, specify DCMD=MSPF to automatically start ISPF whenever you open a new session. If you do not want a default command, leave DCMD blank. This default can be changed with the SET CMDDEFLT command.

FCMD Specify the default final command to be executed when all sessions have been ended. You can specify any TSO command and its parameters. For example, specify FCMD=LOGOFF to automatically log the user off TSO when all sessions are ended. If you do not want a default final command, leave FCMD blank. This default can be changed with the SET CMDFINAL command.

FENCE Specify the amount of storage, in K, to be reserved in the region when the user executes the PIE/TSO FENCE command. Some programs, such as APL2, do a GETMAIN for the entire region. Because the program takes all available memory, users can't start any more sessions. However if users fence memory before starting the program that does the GETMAIN for the entire region, they can release the memory after the program is initialized, and they will have memory to start new sessions.

FGPGN Specify a special performance group for the MultiTask foreground session. If you don't code FGPGN, the foreground session will use the performance group the user logged on with or the performance group assigned to the application with the PGM component.

HEADER Specify OFF to blank out the header line on the PIE/TSO menu—that is, the headings for the session portion. Specify ON to turn the header line back on. Same as SET HEADER.

HDRTXT Change the text of the header line on the PIE/TSO menu. The text can be up to 64 characters long. It must be enclosed in quotes.

HELPDD Specify the ddname you will use if you pre-allocate the PIE/TSO HELP dataset.

HELPDS Specify the name of the PIE/TSO HELP dataset to pre-allocate it automatically.

HELPLN Specify the size, in 80 character lines, of the dynamically acquired buffer for PIE/TSO HELP. If you specify less than 200 lines, PCSINIT will issue the warning message PCS235W MINIMUM HELPLN IS 200, CHANGED TO 200, and it will acquire a 200 line buffer.

☞ If the HELPLN buffer is too small to contain a particular HELP member, an END OF BUFFER message will appear after the last line displayed on the screen. If you want HELP to display this member, increase the HELPLN value, and reinitialize PIE/TSO by starting PCSINIT with an ACT=NEWPARM parameter. The new value of HELPLN will not take effect until users log on again.

IMPTSO Specify the default for “implied TSO.” If YES, PIE/TSO will execute any unknown command as a TSO command. It will start a new TSO session and execute the command from that session. If NO, PIE/TSO will issue an error message for unknown commands. This default can be changed with the SET IMPTSO command.

INFODD Specify the ddname you will use to pre-allocate the PIEINFO dataset.

INFODS Identify the PIEINFO dataset to pre-allocate it automatically. If you have changed this dataset name from the default, change the name on this parameter.

INFWIN Specify OFF to omit the Information Window from the PIE/TSO menu display. Specify ON to redisplay it. This default can be changed with the SET INFOWINDOW command.

INTCMD Specify how to treat an initial command supplied from the COMMAND field of your logon screen. YES executes it in only the first session. NO executes it in every session.

JMPKEY Specify the default for the jump key. Specify any key name to turn the jump key on and set it to a particular key (PF1 to PF24 or 1 to 24 or PA3). Specify NO to turn the jump key off. This default can be changed with the JMPKEY command.

KEYS Specify ON to activate PIE/TSO function keys in sessions. When session keys are on, the keys in the SKEYS table will be activated and will override the normal key function for the application running in the session. The SKEYS are the keys defined with the SET PFn and SET PA3 commands. Specify OFF to use the application function keys in every session. This default can be changed with the SET KEYS command. (While the KEYS parameter can turn SKEYS on and off only for all sessions, the SET KEYS command can turn SKEYS on or off in particular sessions.)

LINPUT Specify how to handle data typed on a screen after the terminal has been locked. PROCESS processes the input when the terminal is unlocked. DISCARD discards it. The user command SET LOCKINPUT performs the same function. However, if you set LINPUT=DISCARD, users cannot SET LOCKINPUT PROCESS. If you set LINPUT=PROCESS, they can SET LOCKINPUT as they want.

LKIGN Specify terminal IDs that should not be AutoLocked. Supply a four character, generic mask for the terminal ID. You may use wildcard characters in your mask. An asterisk (*) indicates any character (or no character) in that position and forward. A question mark (?) indicates any character in that position only. A pound sign (#) indicates any number in that position only. Example: A mask of V?#* would match VA6 and V92X but not V9AX or V9.

LOCK Specify the time, in seconds, to wait before AutoLocking a terminal that is not active. Zero, the default, disables AutoLock. Users may lock their terminals manually at any time with the LOCK command.

LOGDIR Specify whether to activate the Logon Director. YES causes the Logon Director to examine the logon command string for the proper control characters. NO disables this function.

LOGOFF Specify whether the LOGOFF command will log the user off TSO. If YES, when the user enters LOGOFF from the last active session, PIE/TSO will log the user off TSO. If NO, PIE/TSO will return the user to the PIE/TSO menu.

MAXSES Specify the maximum number of sessions a user can have open at the same time.

MAXTRY Specify the maximum number of times a user can enter invalid passwords in response to the AutoLock message. After this number has been exceeded, PIE/TSO will disconnect the user ID on the terminal.

MENU Controls PIE/TSO menu display. You can use SET MENU OFF when you want to make PIE/TSO transparent to users. This is useful, for instance, if you want to use only PIE/TSO's AutoLock feature. You can allow users only one session, activate that session automatically, and turn off the PIE/TSO menu. AutoLock will work, but users

will never know that PIE/TSO is on. This default can be changed with the SET MENU command.

MKEYS Controls the PIE/TSO menu keys. If you specify MKEYS, the PIE/TSO menu will use the MKEYS table for PF key function. If you specify SKEYS, it will use the SKEYS table. This default can be changed with the SET MKEYS command.

MTASK Authorize use of MultiTask. MTASK does not turn MultiTask on or off in a user's environment. The user must do that with the SET MTASK command.

OKSWAP Determine how PIE/TSO handles swappability. If you specify YES, PIE/TSO will maintain swappability for each session individually. If you specify NO, if an application in one session prohibits swapping, swapping is prohibited for all sessions. OKSWAP=NO is required for TCP/IP support.

PASENC Specify whether all users who log on with the Logon Director must use the encrypted form of the password. (PIE/CICS can produce encrypted passwords. See the *PIE/CICS Resource Definition and Management* manual for more information.)

PFSHOW Controls display of the function key help line on the PIE/TSO menu. Specify ON or OFF. This default can be changed with the PFSHOW command.

PGM Authorize use of the Performance Group Manager (PGM).

PIE Authorize use of PIE/TSO. If you want to limit access, specify NO and allow access to specific users or terminals with the PCSUSERS table. See “Step 6: Overriding Defaults for Users and Terminals,” on page 42, for more information.

PREFIX Specify a prefix for PIE/TSO commands to make them unique. So if you code PREFIX=@ and you already have a TSO CREATE command, you can rename the PIE/TSO CREATE module to @CREATE, and execute the command as @CREATE. When you execute CREATE, the non-PIE/TSO command will execute.

The PIEEXEC command provides an alternative to this approach for some PIE/TSO commands. Please see the *PIE/TSO Command Reference* for a description of PIEEXEC.

SCREEN Specify how PIE/TSO is to treat data on the screen when users switch out of a session before hitting ENTER or a PF key to enter the data. SAVE processes the current screen, less the field containing the switch sequence, when the session is reentered. NOSAVE passes the current reshow key (default PA2) to the application when the user reenters the session. Users may change this default with the SCREEN command.

SECTMP Specify which Terminal Monitor Program (TMP) program to use for sessions 1 through 11. You can override the SECTMP parameter in an individual's logon procedure, in the EXEC statement PARM parameter.

SESATT Controls display of the session attribute column, on the right side of the PIE/TSO menu. Specify ON or OFF. This default can be changed with the SET SESATT command.

SMFRTY Start-up SMF recording. To record data, specify an SMF record type (128-255) that is not being used by any other application. Example: `SMFRTY=227`

To turn off recording, specify `SMFRTY=NONE`.

SWAP Authorize PGM SWAP/NOSWAP processing.

SWCHAR Specify the switch character. The switch character is used in conjunction with the switch key (SWKEY below) to switch from session to session. The only valid switch characters are

non-alphanumeric characters found on a standard keyboard. APL characters and other special graphics characters are not supported. Users can override the default with the SWCHAR command.

SWEXIT Specify whether to call the PCSWEXIT switch exit at each switch request.

SWKEY Specify the switch key. The switch key is used in conjunction with the switch character (SWCHAR above) to switch from session to session. Specify any PF key or ENTER. Users can override the default with the SWKEY command.

SWSCAN Specify the number of modified fields to search in a data stream for the switch sequence. IMS users especially may need to increase this value for use with Network Access.

SYSKEY Specify whether a user can switch out of a session running an application in system key or in supervisor state.

TITLE1 and TITLE2 Change the text of the title line on the PIE/TSO menu. TITLE1 specifies the text in column 1 to 70. TITLE2 specifies the text in column 71 to 79. Enclose the text for both options in quotes.

TMP Identify the Terminal Monitor Program (TMP) for session 0. You can override the TMP parameter in an individual's logon procedure, in the EXEC statement PARM parameter. Specify ADFMDF03 if you run the IBM TSO Session Manager.

TSOMON Specify whether to run the interface to Legent's TSOMON. If you use TSOMON, code TSOMON=YES. Also code PIE=YES in TSOMON's initial parameters.

UCLASS Specify the authorized classes for PIE/TSO commands. All PIE commands have been assigned to either class 0, 1, or 5. The common commands are either class 0 or class 1. The system programmer commands are class 5. (See the *PIE/TSO Administrator Guide* for these commands.) Classes 0 and 1 are required. Class 5 is optional. Do not use commas or spaces between classes.

UPCASE Specify whether the PIE/TSO menu should be displayed in all capital letters.

VIEW Authorize use of View.

VIEWDD Specify the ddname for View datasets. The View dataset stores View screen images.

VIEWDS Identify the global View dataset. If you have changed this dataset name from the default, change the name on this parameter. View must have a dataset allocated before it will execute. Specify NULLFILE to view screens, but not save them.

VSTOR Determine whether PIE/TSO will check available storage before opening a new session. Specify CHECK to perform the check. Specify NOCHECK to skip the check.

Example

```
ASYS=YES                ALLOW ASYNCHRONOUS SWITCHING
LINPUT=PROCESS          PROCESS INPUT EVEN IF LOCKTIME EXPIRED
MTASK=NO                DO NOT ALLOW MULTITASKING
HELPDS=PIE.PTS321.HELP  DSNAME FOR PIE TUTORIAL
HELPDD=PIEHELP          DDNAME FOR PIE TUTORIAL
HELPLN=400              MAX NUMBER OF LINES FOR TUTORIAL
MAXSES=12               MAX NUMBER OF SESSIONS (UP TO 12)
TMP=IKJEFT01            TMP NAME FOR SESSION 0
SECTMP=IKJEFT01         TMP NAME FOR OTHER SESSIONS
APF=SWITCH              SWITCH OR NOSWITCH TO DISABLE
SWCHAR=/                SWITCH CHARACTER
SWKEY=ENTER             SWITCH PFKEY
ACCESS=CANCEL           ALLOW ACCESS SESSIONS TO BE CANCELED
INFODS=PIE.PTS321.INFO  DSNAME FOR PIE INFO DATASET
INFODD=PIEINFO          DDNAME FOR PIE INFO DATASET
MAXTRY=5                NUMBER OF ATTEMPTS TO ENTER
*                       PASSWORD WHEN LOCKED
PREFIX=@                PREFIX TO MAKE PIE TSO COMMANDS UNIQUE
SWSCAN=32               NUMBER OF FIELDS TO SEARCH FOR SWITCH
*                       SEQUENCE
SYSKEY=SWITCH           DISABLE SWITCH WHEN USER IS IN
*                       PRIVILEGED STATE
SCREEN=SAVE             SCREEN=SAVE/NOSAVE
UCLASS=01               USER AUTHORIZATION CLASSES FOR PIE
*                       COMMANDS
UPCASE=NO               DISPLAY PIE/TSO MENU IN MIXED CASE
LOCK=0000               AUTOLOCK OFF
FENCE=256               HOW MUCH OF ADDRESS SPACE TO SAVE FOR
*                       LATER USE
TSOMON=NO               DISABLE INTERFACE TO TSOMON
SWEXIT=NO               DISABLE SWITCH EXIT
VIEW=YES                ENABLE THE VIEW FEATURE
INTCMD=YES              INITIAL COMMAND IN ALL SESSIONS
COUSER=0                NUMBER OF CONCURRENT USERS UNLIMITED
FILCMP=YES              FILE ACCESS COMPONENT
LOGCMP=YES              LOGON DIRECTOR COMPONENT
MTTCMP=NO               MULTITASK COMPONENT
NETCMP=YES              NETWORK ACCESS COMPONENT
SESCMP=YES              TSO SESSIONS COMPONENT
VIECMP=YES              VIEW COMPONENT
PGMCMP=YES              PERFORMANCE GROUP MANAGER
PIE=YES                 ALL USERS AUTHORIZED TO USE PIE
LOGDIR=YES              IF LOGON DIRECTOR IS TO BE AVAILABLE
LOGOFF=YES              LAST SESSION LOGOFF - LOGOFF USE
PGM=YES                 PGM AVAILABLE
SWAP=YES                SWAP STATUS TO BE PROCESSED
PASENC=NO               NO ENCRYPTED PASSWORDS FOR LOGON
*                       DIRECTOR
VIEWDS=PIE.PTS321.VIEW  DEFAULT PDS USED TO STORE SCREEN IMAGES
```

```
VIEWDD=PIEVIEW          DDNAME USED TO ALLOCATE THE PRIMARY
*                       OR ALTERNATE VIEW DATASETS
EXPDT=YY.DDD            EXPIRATION DATE FOR PIE/PIE/TSO
*
CPUIDS                  CPUIDS MUST BE LAST PARAMETER
000000000000=XXXXXXXXXXX  INSERT YOUR CPUIDS AND PASSWORDS HERE.
111111111111=YYYYYYYYYYY  ENTER AS MANY cpuid=password
222222222222=ZZZZZZZZZZZ  PAIRS AS REQUIRED.
```

Step 6: Overriding Defaults for Users and Terminals

You may have users who need special environments or special privileges. For instance, you may want to authorize system programmers to execute class 5 commands. You may override certain installation defaults for both users and terminals by creating entries in the PCSUSERS table.

The default PCSUSERS table does not override any PARMLIB(SESSIONS) defaults.

1. To change your PCSUSERS table, edit member PCSUSERS of the SAMPLIB dataset.

   The PCSUSERS table consists of one or more @SUSER macros. Use only one table for all @SUSER macros.

2. Add or change @SUSER macros in the table. See the format and parameters on the following pages.

   To change from a default, code the corresponding parameter in the macro for that user or terminal. If you do not code a parameter, the PARMLIB(SESSIONS) value for that parameter stands.

3. Use member PCSUSERS of the CNTL dataset to assemble and link your new table.

FORMAT

```
@SUSER  { userid          }
        { TERM=nodename   }

        [,ACCESS={ CANCEL   }]
                 { NOCANCEL }

        [,APF={ SWITCH   }]
              { NOSWITCH }

        [,ASYS={ YES }]
               { NO  }

        [,BGPGN=n]
        [,FGPGN=n]
        [,CLASS=nnn]
        [,DDNTABL=ddname]
        [,ENQTABL=module]
        [,FENCE=nnn]
        [,LKIGN=mask]
        [,LOCK=nnnn]
        [,MENU=module]

        [,MTASK={ YES }]
                { NO  }

        [,PGM={ YES }]
              { NO  }

        [,PGMTABL=module]
        [,PIE=YES|NO]

        [,SCREEN={ SAVE   }]
                 { NOSAVE }

        [,SESSIDS=x]

        [,SWAP={ YES }]
               { NO  }

        [,SYSKEY={ SAVE   }]
                 { NOSAVE }
```

Parameter	Valid	Default
userid	1 to 7 characters	None
TERM	nodename	None
ACCESS	CANCEL or NOCANCEL	SESSIONS ACCESS
APF	SWITCH or NOSWITCH	SESSIONS APF
ASYS	YES or NO	SESSIONS ASYS
BGPGN	1 to 999	SESSIONS BGPGN
FGPGN	1 to 999	SESSIONS FGPGN
CLASS	01 or 015	SESSIONS UCLASS
DDNTABL	1 to 8 characters	PCSDDNTB
ENQTABL	1 to 8 characters	PCSENQTB
FENCE	0 to 2048	SESSIONS FENCE
LKIGN	4 characters	SESSIONS LKIGN
LOCK	0 to 9999	SESSIONS LOCK
MENU	module	PNAAPPLS
MTASK	YES or NO	SESSIONS MTASK
PGM	YES or NO	SESSIONS PGM
PGMTABL	module	PCSPGMTB
PIE	YES or NO	SESSIONS PIE
SCREEN	SAVE or NOSAVE	SESSIONS SCREEN
SESSIDS	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B	All sessions
SWAP	YES or NO	SESSIONS SWAP
SYSKEY	SWITCH or NOSWITCH	SESSIONS SYSKEY

Code either userid or TERM.

userid Specify the user or users affected. The userid parameter must immediately follow the @SUSER parameter. You may specify a particular user ID or a generic one.

To match all users beginning with a particular string, specify only the string. For instance, to match all entries beginning with SP, specify SP as your userid. You may also use a question mark to match any character in that position. To match all users, use an asterisk (*) as the userid.

☞ PIE/TSO will use only the first @SUSER macro it finds that matches this user ID. So you must code all the parameters you want in effect for a user on a single macro. In addition, you should place the most specific macros first, then the generic macros, with the asterisk macro last. Any macro after the asterisk macro will be ignored.

TERM Specify the terminal or terminals affected. Be sure to specify physical nodenames. The same generics that are available to the userid parameter are available to TERM, with the same matching qualifications.

You may mix @SUSER TERM macros and @SUSER userid macros in your PCSUSERS load module. If there is a match for both userid and terminal, the user will receive whichever macro is listed first in the table.

ACCESS Specify whether users may cancel Network Access sessions from the PIE/TSO menu or whether they must log off the VTAM application to end the session.

APF Specify whether users can switch out of a session running an APF authorized application.

ASYS Specify whether users can switch out of a session that is processing a task. If YES, SNA terminal users can switch to the PIE/TSO menu with the ATTN key, and non-SNA terminal users can switch with RESET and PA1. ASYS must be set to YES to use MultiTask. When ASYS=YES, users may turn asynchronous switching on in their environment with the SET ASYS command.

BGPGN Specify a special performance group for the MultiTask background session. If you don't code BGPGN, the background session will use the performance group the user logged on with or the performance group assigned to the application with the PGM component.

CLASS Specify the authorized classes for PIE/TSO commands. All PIE commands have been assigned to either class 0, 1, or 5. The common commands are either class 0 or class 1. The system programmer commands are class 5. (See the *PIE/TSO Administrator Guide* for these commands.) Classes 0 and 1 are required. Class 5 is optional. Do not use commas or spaces between classes.

DDNTABL Identify the module containing a ddname table to substitute for the PCSDDNTB table.

ENQTABL Identify the module containing an ENQ table to substitute for the PCSENQTB table.

FENCE Specify the amount of storage, in K, to be reserved in the region when the user executes the PIE/TSO FENCE command. Some programs, such as APL2, do a GETMAIN for the entire region. Because the program takes all available memory, users can't start any more sessions. However if users fence memory before starting the program that does the GETMAIN for the entire region, they can release the memory after the program is initialized, and they will have memory to start new sessions.

FGPGN Specify a special performance group for the MultiTask foreground session. If you don't code FGPGN, the foreground session will use the performance group the user logged on with or the performance group assigned to the application with the PGM component.

LKIGN Specify terminal IDs that should not be AutoLocked. Supply a four character, generic mask for the terminal ID. You may use wildcard characters in your mask. An asterisk (*) indicates any character (or no character) in that position and forward. A question mark (?) indicates any character in that position only. A pound sign (#) indicates any number in that position only. Example: A mask of V?#* would match VA6 and V92X but not V9AX or V9.

LOCK Specify the time, in seconds, to wait before AutoLocking a terminal that is not active. Zero, the default, disables AutoLock. Users may lock their terminals manually at any time with the LOCK command.

MENU Identify the ACCESS menu module to be used with Network Access.

MTASK Authorize use of MultiTask. MTASK does not turn MultiTask on or off in a user's environment. The user must do that with the SET MTASK command.

PGM Authorize use of the Performance Group Manager (PGM).

PGMTABL Identify the module containing a PGM table to substitute for PCSPGMTB.

PIE Authorize use of PIE/TSO.

SCREEN Specify how PIE/TSO is to treat data on the screen when users switch out of a session before hitting ENTER or a PF key to enter the input. SAVE processes the current screen, less the field containing the switch sequence, when the session is reentered. NOSAVE passes the current reshow key (default PA2) to the application when the user reenters the session. Users may change this default with the SCREEN command.

SESSIDS Specify which sessions the user can open. You may authorize all 12 sessions or as few as you want. Code a string of IDs without commas.

SWAP Authorize PGM SWAP/NOSWAP processing.

SYSKEY Specify whether a user can switch out of a session running an application in system key or in supervisor state.

EXAMPLE

All TSO user IDs beginning with SP will be allowed to execute all command classes and cancel Network Access sessions, and their terminals will be automatically locked if they are inactive for 30 minutes.

In addition, any user from a terminal whose nodename begins with L5A will be able to cancel Network Access sessions, will be limited to class 0 and 1 commands, and their terminals will be automatically locked if they are inactive 5 minutes.

Any user ID that begins with AP will have 5 sessions numbered 1-5.

Any user ID that ends with DP39 may execute class 0, 1, and 5 commands.

All other users will take default values from the SESSIONS member in PARMLIB.

```
TITLE 'PIE/TSO PIE/TSO USERS TABLE'
PRINT NOGEN

@SUSER SP,ACCESS=CANCEL,LOCK=1800,CLASS=01234567
@SUSER TERM=L5A?????,ACCESS=CANCEL,LOCK=300,CLASS=01
@SUSER AP,SESSIDS=12345
@SUSER ???DP39,CLASS=015
@SUSER *      DEFAULT ENTRY
END
```

Step 7: Update the PCSSWEQT Table

Some ENQs should not be interrupted to switch to another session. Some of these ENQs are strictly MVS, and others belong to program products. PIE/TSO has a table of ENQs for which switching must be denied.

If you experience ENQ lockouts with PIE/TSO, add the major name of the ENQ that caused the problem to the table. The table is maintained in the CSECT PCSSWEQT in load module PCSTABLE.

1. To update the table, edit member PCSSWEQT of the SAMPLIB dataset.
2. Add the new entry. The entry must be 8 characters long: `DC  CL8'name  '`
3. Be sure the table ends with: `DC F'-1'`
4. Use the member PCSSWEQT of the CNTL dataset to assemble and link edit the new module. Changes will go into effect when users log on again.
5. Please notify UNICOM Systems of any ENQs you have added to PCSSWEQT so that we can add the ENQ to our default table.

Step 8: Modify the PCSENQTB Module

Very few installations have to change the PCSENQTB.

The ENQ/DEQ mechanism is used by programs or commands to control access to a serially reusable resource. ENQs are used to protect a program from itself (SCOPE=STEP), from other programs on the same system (SCOPE=SYSTEM), and from programs on other systems (SCOPE=SYSTEMS).

Many TSO commands issue an exclusive STEP ENQ on a resource. This prevents any other program in the same address space from using that resource until the first program is done with it. Sometimes an exclusive ENQ is done to protect data integrity. Other times an exclusive ENQ is done because the programmer did not want anyone to execute the same command recursively.

When PIE/TSO became available, the concept of ENQs in a TSO address space changed. Some STEP ENQs really should be SCOPE=SESSION ENQs. For example, ISPF does an exclusive STEP ENQ on itself and the user ID. Under standard TSO, that ENQ prevents a user from executing ISPF recursively. The ENQ is valid.

With PIE/TSO and MSPF installed, the ISPF ENQ should really only apply to one session, not the whole address space. By coding the appropriate @SENQ macro, a STEP ENQ can be treated as though it were a SESSION ENQ. @SENQ extends the purpose of an ENQ.

Two common commands that can take advantage of this approach are ISPF and TEST. The PCSENQTB shipped with PIE/TSO, as CSECT PCSENQTB in PCSTABLE, has the appropriate @SENQ macros coded to support these two commands. They allow users to execute MSPF in several sessions, as well as to execute several copies of the TEST command at one time.

This approach does not allow any user to bypass the standard ENQs done for dataset integrity. Only one user can edit a particular member of a dataset at a time.

You may add any user ENQ to the PCSENQTB table, if appropriate. However do not exceed ten @SENQ macros.

The source for the default PCSENQTB table is in SAMPLIB. You may use the JCL in member PCSENQTB of the CNTL dataset to reassemble and link the table.

You may substitute a special table for particular users or terminals. To do so, create a new table and assemble it with PCSENQT2 of the CNTL datast. Be sure to specify the alternate table name in the MEMBER parameter. Then identify the alternate table in the PCSUSERS table. See “Step 6: Overriding Defaults for Users and Terminals,” on page 42, for more information.

⚠ WARNING

@SENQ does not eliminate the need to execute the IBM ENQ macro.

FORMAT

The format of @SENQ is basically the same as the IBM ENQ macro.

```
label @SENQ (qname,rname,E,length,STEP),DSECT=NO
```

label	Specify any legal label.
qname	Identify the qname coded on the corresponding ENQ macro.
rname	Specify a user name. To substitute any user name at execution time, code @USERID.
E	Specifies exclusive use. No other command will be allowed to successfully ENQ on the qname specified in this session ID for this user ID.
length	Specify the length of the rname on the corresponding ENQ macro. It is normally 7 or 8.
STEP	Specify the scope of the ENQ. If you specify STEP the ENQ will be treated as a SCOPE=SESSION ENQ. You should usually code STEP.
DSECT	Specify whether to generate a DSECT of the @SENQ entry. DSECT=NO is the default.

EXAMPLE

This example is from the default table.

```
@SENQ (IKJTMPNM,@USERID,E,8,STEP),DSECT=NO    TEST command
@SENQ (SPFUSER,@USERID,E,7,STEP),DSECT=NO     SPFUSER ENQ
```

Chapter 3
Implementing Network Access

This chapter describes how to implement Network Access. You must have completed the procedures in Chapters 1 and 2 before attempting to implement Network Access.

To implement Network Access you will:

- customize the Network Access application table
- customize the Network Access user ID tables
- add a new logmode table to VTAM (optional)
- add virtual terminals to VTAM
- update IMS terminal tables (optional)
- authorize the ACCESS command
- move load modules to SYS1.LPALIB (optional)
- edit the Network Access parameter dataset (optional)
- modify the PNAIPL procedure
- activate and test Network Access

If done by a VTAM administrator with the proper authorization, Network Access implementation should take approximately one to two hours. The actual time depends on the complexity of your network and number of applications you want to make available for access.

We strongly suggest that you define only a minimum number of applications and user IDs at the start. When you have tested a skeleton system, then you may phase in more applications and user IDs as appropriate.

VTAM is extremely flexible. Network Access is also extremely flexible. This flexibility puts you in control. However the necessary counterpart to flexibility and control has always been user knowledge. It is essential that you take the time to learn how to implement Network Access properly. Please read and understand the remainder of this overview before you begin Network Access implementation.

You define which applications Network Access can access, which users can access those applications, and which virtual terminals and logmodes to use in the process. You will define these items by modifying the application table (PNAAPPLS) and at least one user ID table (PNAUSERS).

You will then create VTAM definitions for the logmodes and virtual terminals Network Access will use.

The following sections explain what these tables and definitions do and how they work together.

Before users can access a VTAM application through Network Access, they must pass through a two layer filtering system.

First, the VTAM application must be authorized. That is, Network Access will only establish a

session with the chosen VTAM application if it is authorized to do so. You grant this authorization through the application table, also called the PNAAPPLS table. You can authorize all applications or you can restrict authorization to particular applications.

Second, the user must be authorized to access that VTAM application. You grant this authorization through user ID tables, also called PNAUSERS tables. You can authorize all users and all terminals or you can restrict access to particular users and terminals.

You can have as many user ID tables as you need. There is only one global application table. (However you can create different application tables for specific users and terminals.)

Now, how do the tables relate to each other? Each entry in the application table authorizes Network Access to access a VTAM application and pairs that application with a user ID table. Every user ID or terminal in that table will be able to access that application.

For example the application table entry

```
@APPL CICSTEST,USERS=PNASYSPR...
```

authorizes Network Access to access the CICSTEST system and pairs CICSTEST with the PNASYSPR user ID table. Any user ID or terminal in the PNASYSPR table will be able to access CICSTEST through Network Access.

You may have several entries for each application in the application table. That way you can authorize different groups of user IDs and terminals to use the same application. For example, you can authorize CICSTEST this way:

```
@APPL CICSTEST,USERS=PNASYSPR...
@APPL CICSTEST,USERS=PNAAPROG1...
@APPL CICSTEST,USERS=PNAAPROG2...
```

Your systems programmers and two groups of applications programmers will have access to CICSTEST. As you can see, in this example we have created separate user ID tables for different groups of people. You can use any type of grouping that is effective for you.

MAKING ACB AND LOGMODE ASSIGNMENTS

When Network Access establishes a session with an application it selects a virtual terminal, assigns a logmode (also called a bind image), and opens an ACB.

You must ensure that Network Access assigns a correct virtual terminal and logmode when it opens its ACB. How you make these assignments depends on the VTAM application being accessed.

Most applications will accept the logmode Network Access sends. Some, however, insist on using their own logmode (such as older versions of CICS and IMS). These applications assign logmodes to individual terminal IDs in their own internal tables.

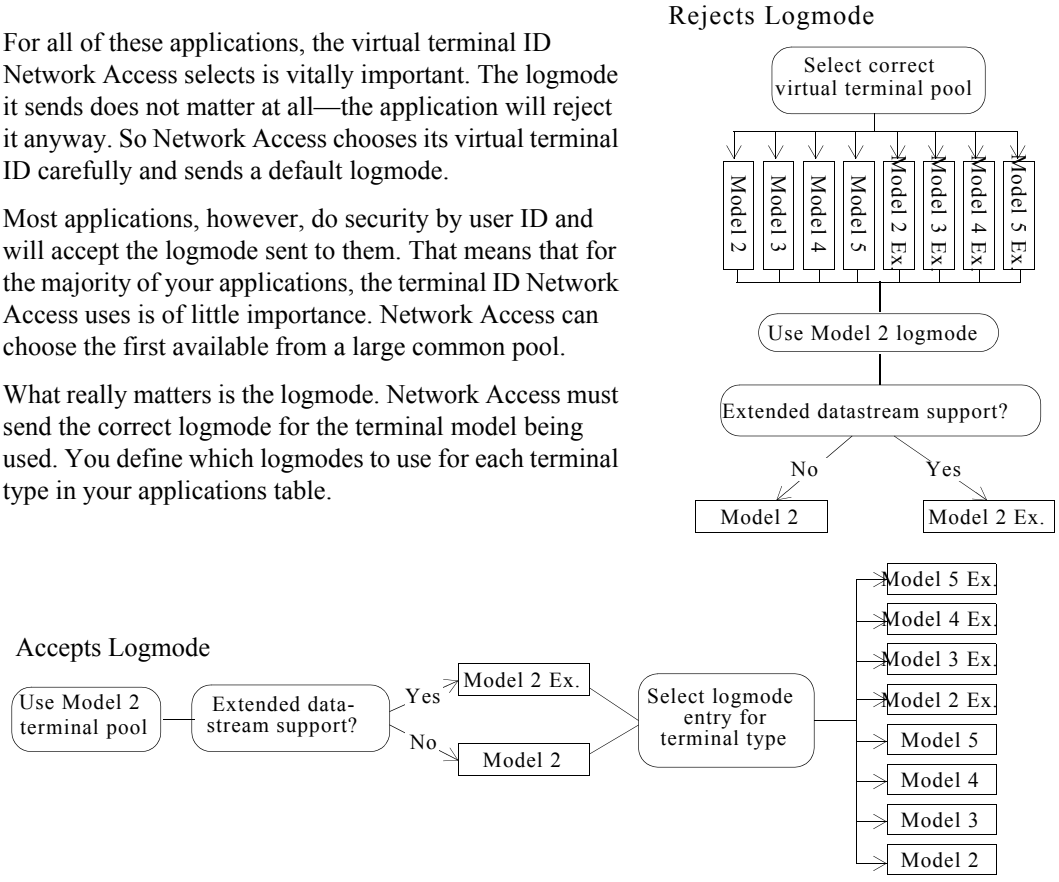
Network Access supports eight different terminal models (2 through 5, with or without extended data stream support). You could be using any of these models when you use Network Access. When Network Access chooses a virtual terminal ID to access one of these applications, it must choose an ID that corresponds to an appropriate logmode in that application's internal table. In practical terms, this means Network Access needs a separate terminal pool for every terminal model you use in your installation.

In addition, some of those same applications (such as IMS) handle security by terminal ID. That means that every user must have his or her own private virtual terminal IDs. (If users share a common pool of terminal IDs, they will also share common authority.) To allow you to give particular users their own virtual terminals, virtual terminal pools are defined in the user ID tables.

For all of these applications, the virtual terminal ID Network Access selects is vitally important. The logmode it sends does not matter at all—the application will reject it anyway. So Network Access chooses its virtual terminal ID carefully and sends a default logmode.

Most applications, however, do security by user ID and will accept the logmode sent to them. That means that for the majority of your applications, the terminal ID Network Access uses is of little importance. Network Access can choose the first available from a large common pool.

What really matters is the logmode. Network Access must send the correct logmode for the terminal model being used. You define which logmodes to use for each terminal type in your applications table.



Sample Definitions

Now, let's turn these concepts into some specific examples. Let's begin with the most common type of application first—an application that accepts the logmode sent to it and which does security by user ID, not terminal ID. TSO and any reasonably current version of CICS fall into this category. Let's define a CICS version 1.7 system, called CICSTEST, to Network Access. First you must indicate to Network Access whether this application will accept the logmode Network Access sends. You do this in the application table with the @APPL macro and the LOGMOD parameter:

```
@APPL CICSTEST,LOGMOD=ACCEPT...
```

Now you must identify the logmodes to send for each terminal model you use in your installation. There is a separate @APPL parameter for each terminal model.

- MOD2ENT, MOD3ENT, MOD4ENT, and MOD5ENT correspond to models 2 through 5 without extended data stream support
- MOD2EXT, MOD3EXT, MOD4EXT, and MOD5EXT correspond to models 2 through 5 with extended data stream support

Let's assume that you use every terminal model at your installation:

```
@APPL CICSTEST,LOGMOD=ACCEPT,                                  X
      MOD2ENT=M32782S,                                         X
      MOD3ENT=M32783S,                                         X
      MOD4ENT=M32784S,                                         X
      MOD5ENT=M32785S,                                         X
      MOD2EXT=M32782X,                                         X
      MOD3EXT=M32783X,                                         X
      MOD4EXT=M32784X,                                         X
      MOD5EXT=M32785X
```

Now you must set up the virtual terminal pools from which Network Access will choose virtual terminal IDs. You set up terminal pools in the user ID table. Since this type of application doesn't require any specific user ID information, you can use a global entry in your default user ID table. (If you want to limit access to this application to certain users or physical terminals, you can create a special user ID table to include only those users and physical terminals. In this example, we are giving access to everyone.)

Like the @APPL macro, the user ID table macro, @AUSER, has a separate parameter for each terminal model.

- VT2POOL, VT3POOL, VT4POOL, and VT5POOL correspond to models 2 through 5 without extended data stream support
- VX2POOL, VX3POOL, VX4POOL, and VX5POOL correspond to models 2 through 5 with extended data stream support

The two model 2 parameters, VT2POOL and VX2POOL, identify the default pools—one for terminals without extended data stream support, one for terminals with it. These are the only pools Network Access will use for CICSTEST, since it is a LOGMOD=ACCEPT application. So you only have to code those two pool parameters:

```
@AUSER *,VT2POOL=APAT,VX2POOL=APAT
```

We used the same virtual terminal pool for both parameters. We could have used different pools if we wanted. But using the same pool name means that we are using a single common pool for all users. We can use this same pool for any LOGMOD=ACCEPT application. Now you must ensure

that this terminal pool is defined to VTAM:

```
APAPB001 APPL  ACBNAME=APAT0001,MODETAB=MTABSL2,EAS=1,          X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAPB002 APPL  ACBNAME=APAT0002,MODETAB=MTABSL2,EAS=1,          X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAPB003 APPL  ACBNAME=APAT0003,MODETAB=MTABSL2,EAS=1,          X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
...
```

These VTAM definitions point to a logmode table. (If MODETAB is omitted, VTAM uses the default logmode table.) That logmode table must include the logmode definitions named in your @APPL macro, above.

```
M32782S  MODEENT LOGMODE=M32782S,
               FMPROF=X'03',
               TSPROF=X'03',
               PRIPROT=X'B1',
               SECPROT=X'90',
               COMPROT=X'3080',
               RUSIZES=X'8989',
               PSERVIC=X'020000000000185018507F00'
*                        0 1 2 3 4 5 6 7 8 9 1011
...
```

With this, your definitions for CICSTEST are complete.

Now let's define a non-standard type application, like an older version of CICS—we'll call it CICSOLD. Remember that this type of application does not accept the logmode Network Access sends. It uses its own logmode, chosen according to terminal ID, from its own internal table. So Network Access must send it a terminal ID that will correspond to the terminal model that is actually in use in CICSOLD's internal table.

Let's start with the application table. You must identify CICSOLD as an application that will reject the logmode Network Access sends:

```
@APPL CICSOLD,LOGMOD=NOACCEPT. . .
```

Although CICSOLD will reject the logmode Network Access sends, protocol demands that Network Access send a logmode anyway. So you must code two default logmodes for it to send—once again, one for terminals with extended data stream support and one for terminals without it. You must use the model 2 parameters, MOD2ENT and MOD2EXT.

```
@APPL CICSOLD,LOGMOD=NOACCEPT,MOD2ENT=M32782S,MOD2EXT=M32782X
```

Since these logmodes must actually be defined in VTAM, it is wise to identify real logmodes that are actually in use elsewhere. Here you are using the logmodes you used for CICSTEST earlier.

Now you must define your virtual terminal pools in your user ID tables. You must define a separate terminal pool for every terminal model you use at your installation. This obviously requires a specialized user ID table. The default table will not meet your needs. Let's call your new user ID table, PNAOLDC. You must reference that table in your @APPL macro:

```
@APPL CICSOLD,USERS=PNAOLDC,                                    X
      LOGMOD=NOACCEPT,MOD2ENT=M32782S,MOD2EXT=M32782X
```

Now you must define the pools in your PNAOLDC user ID table:

```
@AUSER  *,MAXSESS=12,                                          X
         VT2POOL=APC2,                                         X
         VT3POOL=APC3,                                         X
         VT4POOL=APC4,                                         X
         VT5POOL=APC5,                                         X
         VX2POOL=AXC2,                                         X
         VX3POOL=AXC3,                                         X
         VX4POOL=AXC4,                                         X
         VX5POOL=AXC5
```

Now set up your VTAM definitions for each pool:

```
APAXBC41 APPL  ACBNAME=APC20001,MODETAB=MTABSL2,EAS=1,          X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAXB432 APPL  ACBNAME=APC20002,MODETAB=MTABSL2,EAS=1,          X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
...
```

These definitions must point to a logmode table that includes the MOD2ENT and MOD2EXT logmodes you specified for CICSOLD in your applications table earlier. In this case this is already done, because you use the same logmodes and logmode table you used for CICSTEST.

Now let's create a definition for an older IMS system, called IMSP. All the requirements for IMSP are the same as for CICSOLD, until you get to the user ID table.

You must still define a separate terminal pool for each terminal model that might be used. In addition, because IMS does security by terminal ID, not user ID, you must define a separate set of terminal pools for each user. This requires another specialized user ID table. In our example, we will authorize two users—JSMITH and JDOE.

```
         @AUSER  JSMITH,MAXSESS=1,                             X
                 VT2POOL=JIS2,                                 X
                 VT3POOL=JIS3,                                 X
                 ...
                 VX5POOL=JISD

         @AUSER  JDOE,MAXSESS=1,                               X
                 VT2POOL=JAD2,                                 X
                 VT3POOL=JAD3,                                 X
                 ...
                 VX5POOL=JADD
```

Once again you will set up your VTAM definitions for each pool. And you must ensure that the logmode table these definitions point to actually contain the logmodes you specified in your IMSP @APPL macro.

APPLICATION TABLE

```
@APPL   CICSTEST,TEXT='TEST CICS, VERSION 1.7',                X
        LOGMOD=ACCEPT,                                         X
        MOD2ENT=M32782S,                                       X
        MOD3ENT=M32783S,                                       X
        MOD4ENT=M32784S,                                       X
        MOD5ENT=M32785S,                                       X
        MOD2EXT=M32782X,                                       X
        MOD3EXT=M32783X,                                       X
        MOD4EXT=M32784X,                                       X
        MOD5EXT=M32785X

@APPL   CICSOLD,TEXT='ANCIENT CICS',                           X
        USERS=PNAOLDC,                                         X
        LOGMOD=NOACCEPT,                                       X
        MOD2ENT=M32782S                                        X
        MOD2EXT=M32782X

@APPL   IMSP,TEXT='PRODUCTION IMS',                            X
        USERS=PNAIMSP,                                         X
        LOGMOD=NOACCEPT,                                       X
        MOD2ENT=M32782S                                        X
        MOD2EXT=M32782X

END
```

USER ID TABLES

```
PNAUSER  TITLE 'DEFAULT TABLE FOR APPLICATIONS WITH LOGMOD=ACCEPT'
         @AUSER  *,VT2POOL=APAT,VX2POOL=APAT,MAXSESS=12
         END

PNAOLDC  TITLE 'USERS TABLE FOR SESSIONS WITH CICS 161'
         @AUSER  *,MAXSESS=12,                                  X
                 VT2POOL=APC2,                                  X
                 VT3POOL=APC3,                                  X
                 VT4POOL=APC4,                                  X
                 VT5POOL=APC5,                                  X
                 VX2POOL=AXC2,                                  X
                 VX3POOL=AXC3,                                  X
                 VX4POOL=AXC4,                                  X
                 VX5POOL=AXC5
         END

PNAIMSP  TITLE 'USERS TABLE FOR SESSIONS WITH IMS'
         @AUSER  JSMITH,MAXSESS=1,                              X
                 VT2POOL=JIS2,                                  X
                 VT3POOL=JIS3,                                  X
                 VT4POOL=JIS4,                                  X
                 VT5POOL=JIS5,                                  X
                 VX2POOL=JISA,                                  X
                 VX3POOL=JISB,                                  X
                 VX4POOL=JISC,                                  X
                 VX5POOL=JISD

         @AUSER  JDOE,MAXSESS=1,                                X
                 VT2POOL=JAD2,                                  X
                 VT3POOL=JAD3,                                  X
                 VT4POOL=JAD4,                                  X
                 VT5POOL=JAD5,                                  X
                 VX2POOL=JADA,                                  X
                 VX3POOL=JADB,                                  X
                 VX4POOL=JADC,                                  X
                 VX5POOL=JADD
         END
```

VTAM DEFINITIONS

```
APAPIETB VBUILD TYPE=APPL
*
* APPLICATION MINOR NODES FOR POOL OF VIRTUAL TERMINALS
* ESTABLISH SESSIONS WITH APPLICATIONS THAT ACCEPT SUGGESTED
* BIND FROM TERMINALS.  THESE APPLICATIONS INCLUDE: TSO,
* OMEGAMON VTAM, VM (VSCS), ROSCOE, CICS WITH AUTOINSTALL, ETC.
* LOGMODE DEFINITIONS ARE CONTAINED IN MTABSL2.
*
APAPB001 APPL  ACBNAME=APAT0001,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAPB002 APPL  ACBNAME=APAT0002,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAPB003 APPL  ACBNAME=APAT0003,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
*
*  APPLICATION MINOR NODES FOR POOL OF VIRTUAL TERMINALS USED TO
* ESTABLISH SESSIONS WITH CICS AS MODEL 4 EXTENDED TERMINALS.
* THESE NETWORK NAMES ARE DEFINED IN CICS TCT AS MODEL 4 EXTENDED
* TERMINALS.
*
APAXBC41 APPL  ACBNAME=AXC40001,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAXB432 APPL  ACBNAME=AXC40002,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
*
*  APPLICATION MINOR NODES FOR POOLS OF VIRTUAL TERMINALS USED TO
* ESTABLISH SESSIONS WITH IMS.EACH POOL IS ASSIGNED TO A SINGLE
* USER. EACH POOL IS DEFINED IN IMS AS THE APPROPIATE MODEL TYPE.
*
...
APAPBI01 APPL  ACBNAME=JISA0001,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAPBI02 APPL  ACBNAME=JISB0001,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAPBI01 APPL  ACBNAME=JISC0001,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
APAPBI02 APPL  ACBNAME=JISD0001,MODETAB=MTABSL2,EAS=1,           X
               PARSESS=NO,SESSLIM=YES,AUTH=NVPACE
...
```

MODE TABLE

```
MTABSL2  TITLE 'PIE/NETWORK ACCESS BIND IMAGES'
*
MTABSL2  MODETAB
*
*/* M3278?S  MODEL ? - WITHOUT EXTENDED DATA STREAM SUPPORT
*
M32782S  MODEENT LOGMODE=M32782S,
               FMPROF=X'03',
               TSPROF=X'03',
               PRIPROT=X'B1',
               SECPROT=X'90',
               COMPROT=X'3080',
               RUSIZES=X'8989',
               PSERVIC=X'020000000000185018507F00'
*                        0 1 2 3 4 5 6 7 8 9 1011
M32783S  MODEENT LOGMODE=M32783S,
               FMPROF=X'03',
               TSPROF=X'03',
               PRIPROT=X'B1',
               SECPROT=X'90',
               COMPROT=X'3080',
               RUSIZES=X'8989',
               PSERVIC=X'020000000000185020507F00'
*                        0 1 2 3 4 5 6 7 8 9 1011
...
*
*/* M3278?X  MODEL ? - WITH EXTENDED DATA STREAM SUPPORT
*
M32782X  MODEENT LOGMODE=M32782X,
               FMPROF=X'03',
               TSPROF=X'03',
               PRIPROT=X'B1',
               SECPROT=X'90',
               COMPROT=X'3080',
               RUSIZES=X'8989',
               PSERVIC=X'028000000000185018507F00'
*                        0 1 2 3 4 5 6 7 8 9 1011
...
M32784X  MODEENT LOGMODE=M32784X,
               FMPROF=X'03',
               TSPROF=X'03',
               PRIPROT=X'B1',
               SECPROT=X'90',
               COMPROT=X'3080',
               RUSIZES=X'8989',
               PSERVIC=X'02800000000018502B507F00'
*                        0 1 2 3 4 5 6 7 8 9 1011
...
```

Step 1: Customize Your Application Table

Before you begin to customize your Network Access application table, please read the overview, beginning on page 51.

1. Edit member PNAAPPLS of the SAMPLIB dataset.
2. PNAAPPLS contains sample @APPL macros for several different VTAM applications. Edit or delete these macros as appropriate.

   At the start, you may want to comment out all but one application. We recommend that you include an entry for a TSO application. Because TSO presents the fewest implementation problems, testing with TSO eliminates the most variables.

   Eventually you must add one or more macros for every application you want to appear on your Network Access menu. You may code one global macro statement to grant access to all applications.

   See “Using the @APPL macro,” on page 62, for information on the @APPL macro.
3. Code an @ATITLE macro, if you want, to give a title line to your Network Access menu. See “Using the @ATITLE macro,” on page 64, for instructions.
4. Use member PNAAPPLS in the CNTL dataset to assemble and link edit PNAAPPLS to the PIE/TSO load library. Your PNAAPPLS member must point to the @APPL and @ATITLE macros in the MACLIB dataset.
5. If you change the name of your Network Access menu table from PNAAPPLS, specify the new name with the PNAPARM parameter in the Network Access parameter dataset. (See “Step 8: Edit the Parameter Dataset,” on page 71.)
6. You may set up special Network Access menu tables for certain users, if you want. To do so, create each new table using steps 1 through 4. Then specify the table name with the MENU option in the PCSUSERS table. (See “Overriding Defaults for Users and Terminals,” on page 44, for more information.)

Using the @APPL macro

@APPL macro statements define which VTAM applications Network Access can access. See “Sample Definitions,” beginning on page 54, for a sample PNAAPPLS table.

FORMAT

```
@APPL  netname,
       [ LOGMOD={ ACCEPT   },]
              { NOACCEPT }
       MOD2ENT=model2entry,
       [,MOD3ENT=model3entry,]
       [,MOD4ENT=model4entry,]
       [,MOD5ENT=model5entry,]
       ,MOD2EXT=model2entry,
       [,MOD3EXT=model3entry,]
       [,MOD4EXT=model4entry,]
       [,MOD5EXT=model5entry,]
       [ERRMSG={YES},]
               {NO }
         [  ID=tso-userid,   ]
         [TERM=term-netname, ]
       [ITEXT=text,]
       [TEXT='text',]
       [USERS=users-table]
```

Parameter	Valid	Default
netname	1 to 8 characters	None
LOGMOD	ACCEPT or NOACCEPT	NOACCEPT
MOD2ENT-MOD5EXT	1 to 8 characters	None
ERRMSG	YES or NO	NO
ID	1 to 7 characters	*
TERM	1 to 8 characters	None
ITEXT	1 to 32 characters	None
TEXT	1 to 32 characters	None
USERS	1 to 8 characters	Global user ID

netname Specify the VTAM APPLID for the application. Do not use the ACBNAME, or Network Access will not be able to create cross domain sessions. If you want, code an asterisk to indicate any APPLID.

Applications you define individually will be displayed on the Network Access menu. The asterisk

entry will not be displayed on any Network Access menus.

LOGMOD Specify whether the application will accept the logmode Network Access passes to it at CINIT time. The default PNAAPPLS table already contains valid LOGMOD values for CICS, IMS, IDMS, TSO, OMEGAMON VTAM, PNMS, MODEL204, and NCCF version 2. Do not change these values unless you have a comprehensive understanding of the application's response to CINIT.

For LOGMOD=ACCEPT applications, code all the MODnENT and MODnEXT parameters for all terminal models you have installed. For LOGMOD=NOACCEPT applications, code only MOD2ENT and MOD2EXT.

MOD2ENT, MOD3ENT, MOD4ENT, MOD5ENT Specify the name of the logmode entries for SNA LU.2 models 2-5 3270 terminals. Member M3278X of the SAMPLIB dataset contains the supported logmodes for SNA LU.2 3270 models 2-5. The model 2 entry name in the sample is M32782S, the model 3 is M32783S, the model 4 is M32784S, the model 5 is M32785S.

MOD2EXT, MOD3EXT, MOD4EXT, MOD5EXT Specify the name of the logmode entry for SNA LU.2 models 2-5 3270 terminals with extended data stream support. Member M3278X of the SAMPLIB dataset contains the supported logmodes for SNA LU.2 3270 models 2-5. The model 2 entry name in the sample is M32782X, the model 3 is M32783X, the model 4 is M32784X, the model 5 is M32785X.

For each application, you may code ID or TERM, but not both. Network Access presents a menu of applications to users when they enter the ACCESS command without an application ID. All users or terminals included by the ID or TERM parameter will see this application on their menus. If you want to code multiple ID or TERM values for a single application, code separate @APPL macros for each one.

ID Specify which TSO users will see this application on their Network Access menus. Specify a complete or generic logon ID. Use a question mark (?) to match any character in a single position. Use an asterisk (*) to include all users.

TERM Specify which terminals will see this application on their Network Access menus. You may specify a complete or generic terminal ID (VTAM node name). Use a question mark (?) to match any character in a single position.

ERRMSG Specify whether Network Access error messages should interrupt full-screen displays. Code YES only for PROFS.

ITEXT Specify any initial text data to pass to the application when it is executed through the Network Access menu. This is the same as the USERDATA portion of the standard BIND image. You may use the variables discussed in the ACCESS command section of the PIE/TSO Command Reference.

TEXT Specify a comment to display on the Network Access menu next to the application name. Include your comment in quotes.

USERS Specify which user ID table to use with this application. You may specify a separate user ID table for each application, or you may use a single table for many applications. If you code multiple @APPL macros for a single application, you may specify a different user ID table for each one. If you omit USERS, Network Access will use the table specified in PARMLIB(NACCESS), the USERS parameter.

Using the @ATITLE macro

@ATITLE specifies a title line for your Network Access menu. It is not a required macro, but if you use it, place it before all @APPL macros.

FORMAT

```
@ATITLE title
```

Parameter	Valid	Default
title	1 to 79 characters	None

title Specify a title line to be placed on the Network Access menu. See the sample PNAAPPLS member for an example.

Step 2: Customize Your User ID Tables

In this step you will create one global user ID table, plus one specialized user ID table for each table referenced in the application table.

1. First create your global user ID table. You may edit member PNAUSERS of the SAMPLIB dataset.
2. Code the necessary @AUSER macros. See "Using the @AUSER Macro," on the next page.
3. Use member PNAUSERS in the CNTL dataset to assemble and link edit your user ID table to the PIE/TSO load library. When you assemble your table, be sure to point to the @AUSER macro in the MACLIB dataset.
4. If you change the name of your global user ID table from PNAUSERS, specify the new name with the USERS parameter of the Network Access parameter dataset. (See "Step 8: Edit the Parameter Dataset," on page 71.)
5. Now create a separate user ID table for each specialized table referenced in the application table. To do so, follow steps 1 through 3 for each table.

   You may put all your user ID tables in the same load library as long as they have unique member names.

Using the @AUSER macro

Each user ID table is composed of @AUSER macro statements. See "Sample Definitions," beginning on page 54, for examples of the PNAAPPLS table.

FORMAT

```
@AUSER  { userid    }
        { TERM=termid}
        [,MAXSESS=n]
        ,VT2POOL=mode12pool]
        [,VT3POOL=mode13pool]
        [,VT4POOL=mode14pool]
        [,VT5POOL=mode15pool]
        ,VX2POOL=mode12pool
        [,VX3POOL=mode13pool]
        [,VX4POOL=mode14pool]
        [,VX5POOL=mode15pool]
```

Parameter	Valid	Default
userid	1 to 7 characters	None
TERM	1 to 8 characters	None
MAXSESS	1 to 12	1
VT2POOL-VX5POOL	4 characters	None

Specify either userid or TERM.

userid Specify which TSO users can access the VTAM applications that list this user ID table in the application table. To specify all users, code an asterisk (*). To specify a generic, code the beginning character string. You may also use a question mark (?) to match any character in a single position.

TERM Specify the generic or actual name of the physical terminal from which a user may access this VTAM application.

MAXSESS Specify the maximum number of copies this user or terminal can open of each VTAM application that lists this user ID table.

For LOGMOD=ACCEPT applications, you need only code the VT2POOL and VX2POOL parameters. For LOGMOD=NOACCEPT applications, code all the VTnPOOL and VXnPOOL parameters for all terminal models you have installed.

VT2POOL, VT3POOL, VT4POOL, VT5POOL Specify the four character pool name Network Access should use when it opens its ACB. These parameters apply to model 2-5 terminals without extended data stream support.

VX2POOL, VX3POOL, VX4POOL, VX5POOL Specify the four character pool name Network Access should use when it opens its ACB. These parameters apply to model 2-5 terminals with extended data stream support.

☞ For both VTnPOOL and VXnPOOL, Network Access will generate the remaining four characters of the ACBNAME. Network Access will use the lowest number available. For instance, if you have VT2POOL=APC2 and two other users have Network Access sessions using APC20001 and APC20003, Network Access will use APC20002.

You may use the same virtual terminal pool in different VTAM domains. However you must use unique network names (the labels on the APPL statements) for each VTAM domain.

Step 3: Add a New Mode Table to VTAM

In this step you will add Network Access logmodes to VTAM. You may create your own logmodes or you can use the logmodes provided with PIE/TSO.

1. Edit or create your logmode member. You may use member M3278X of the SAMPLIB dataset to define logmodes for SNA LU.2 models 2-5 3270s. We suggest that you use this mode table to avoid confusion in initial setup of Network Access. With M3278X, Network Access will execute properly on local SNA 3270s, remote SNA 3270s, local non-SNA 3270s, and remote BSC 3270s.

   There are 2 entries for each physical terminal model. The first entry is for physical terminals that do not support extended data stream. The Query bit in the PSERVIC operand is x'00'. A second entry is for the same model terminal that supports extended data stream. The Query bit is on for these definitions.

   ☞ Do not change the FMPROF or TSPROF fields in the sample tables.

   If you create your own table, you must use SNA LU.2 3270 logmodes even if your physical terminals are non-SNA. This is because the ACCESS command looks like an SNA LU.2 to an application.

2. Assemble and link edit the logmode member to VTAMLIB. You may use member MODETABL of the CNTL dataset to assemble and link edit M3278X from SAMPLIB to SYS1.VTAMLIB.

Step 4: Add Virtual Terminals to VTAM

In this step, you will define Network Access' virtual terminals to VTAMLST. You will create a definition for each pool you identified in your user ID tables—parameters VTnPOOL and VXnPOOL. You may create your own definitions or use the definitions provided with PIE/TSO.

1. Add a new application major node (member) to VTAMLST. To use the PIE/TSO definitions, copy member PIE of the SAMPLIB dataset to VTAMLST.

   Member PIE defines 20 virtual terminals (APPLs) for establishing LU-LU sessions with Network Access. (The figure on page 69 lists part of this member.) These definitions refer to a mode table named M3278X. If you did not use M3278X in “Step 3: Add a New Mode Table to VTAM,” on page 67, then specify your mode table in the MODETAB keyword in each APPL statement.

   The nodes' last 4 characters must be numeric, beginning with 0001 and ending as high as 9999, but in all circumstances must be in sequential order (e.g., PIE10001, PIE10002, etc.).

   You may use the same virtual terminal pool in different VTAM domains. However you must use unique network names (the labels on the APPL statements) for each VTAM domain.

2. Be sure that you used the correct logmode entry names in your application table.

3. Vary the major node active from the console to activate the virtual terminals immediately:

   V NET,ACT,ID=member

4. Add the new table to the list of existing major nodes. To do so, add the member name to member ATCCONxx in VTAMLST.

   For example, if your current ATCCONxx member looks like this:

   ```
   TSOAPPL,SNA,NONSNA,CICS,CICSTEST
   ```

   and you add the PIE/TSO table, add PIE to the list:

   ```
   TSOAPPL,SNA,NONSNA,CICS,CICSTEST,PIE
   ```

MEMBER PIE

```
         VBUILD TYPE=APPL
PIE10001 APPL  ACBNAME=PIE10001,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10002 APPL  ACBNAME=PIE10002,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10003 APPL  ACBNAME=PIE10003,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10004 APPL  ACBNAME=PIE10004,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10005 APPL  ACBNAME=PIE10005,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE
PIE10006 APPL  ACBNAME=PIE10006,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10007 APPL  ACBNAME=PIE10007,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10008 APPL  ACBNAME=PIE10008,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10009 APPL  ACBNAME=PIE10009,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10010 APPL  ACBNAME=PIE10010,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10011 APPL  ACBNAME=PIE10011,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10012 APPL  ACBNAME=PIE10012,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10013 APPL  ACBNAME=PIE10013,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10014 APPL  ACBNAME=PIE10014,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10015 APPL  ACBNAME=PIE10015,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
PIE10016 APPL  ACBNAME=PIE10016,                          *
               MODETAB=M3278X,                            *
               EAS=1,PARSESS=NO,SESSLIM=YES,AUTH=(NVPACE)
```

Step 5: Update Terminal Tables for IMS

Older versions of IMS do not allow users to log on from undefined terminals. If you are running an older version of IMS, you must define the Network Access virtual terminals on these systems. You don't have to define the physical terminals.

1. Update the IMS terminal table. You may use member IMSSAMP of the SAMPLIB dataset.

IMSSAMP

```
      COMM  APPLID=IMS1, . . .
      TYPE  UNITTYPE=SLUTYPE2,     <== REQUIRED   X
            MODEL=2,               <== SUGGESTED  X
            FEAT=(NOCD),                          X
            OPTIONS=TRANRESP
      TERMINAL NAME=PIE10001
      NAME     PIE10001
```

Step 6: Authorize the ACCESS Command

ACCESS is an authorized command. In this step, you will add it to IKJTSOxx SYS1.PARMLIB.

1. Edit IKJTSOxx.
2. Add ACCESS to AUTHCMD and to AUTHTSF NAMES.
3. Use the TSO PARMLIB command to update the table dynamically:

   PARMLIB UPDATE(xx)

Step 7: Move Load Modules to SYS1.LPALIB

To improve performance move some of the Network Access load modules to SYS1.LPALIB. For more information, see“Move Load Modules to SYS1.LPALIB,” on page 26.

Step 8: Edit the Parameter Dataset

The Network Access parameter dataset points to your application table and your default user ID table. The parameter dataset is identified by the SYSIN DD statement in the PNAIPL procedure. (You will edit the PNAIPL procedure in the next step.) It must be a partitioned dataset member, with an LRECL of 80.

You can override the global application table for particular users and terminals. See "Overriding Defaults for Users and Terminals," beginning on page 44, for more information.

1. Edit member NACCESS of the PARMLIB dataset.
2. Identify your application and your default user ID table with the following parameters.

```
APPLS=tablename
USERS=tablename
```

Parameter	Valid	Default
APPLS	1 to 8 characters	PNAAPPLS
USERS	1 to 8 characters	PNAUSERS

Code each statement on its own line, beginning in column one. You may code comments—begin them with an asterisk (*) or begin the comment after the keyword parameter and its value, followed by at least one space.

APPLS Specify the load module name of the Network Access application table.

USERS Specify the load module name of the default user ID table.

EXAMPLE

```
APPLS=PNAAPPLS    NAME OF DEFAULT APPLICATIONS TABLE
USERS=PNAUSERS    NAME OF DEFAULT USERS TABLE
*
* CUSTOMER NAME
* USI
```

Step 9: Modify the PNAIPL Procedure

PNAIPL starts Network Access. It initializes control blocks and terminates immediately after initialization is complete. The distributed PNAIPL procedure is shown on page 73.

You can run PNAIPL from an operator console using the MVS START command or automatically through a COMMNDxx member. See “Automate Start-Up,” on page 90, for details.

1. Edit member PNAIPL of the PIE/TSO procedure library.
2. Update the PARMDS parameter if you have changed the PIE/TSO dataset names.
3. Supply the name of your APF authorized library with the APF parameter.

THE PNAIPL PROCEDURE

The parameters for the PNAIPL procedure are:

APF Specify the APF authorized library or link list library where the Network Access load modules reside. APF is used for both the STEPLIB and SYSLIB DD statements. You must supply a name. For example: APF='SYS1.LPALIB'

ACT Specify the action to be taken by the PNAIPL job. The default is START, which will activate Network Access. See “Starting and Stopping Network Access” in the PIE/TSO Administrator Guide for the other options available.

DUMP Specify a SYSOUT class for dumps generated if there is a problem with PNAIPL. The default is A.

MEMBER Specify the member name of the Network Access parameter dataset. The default is NACCESS.

PARMDS Specify the name of the partitioned dataset that contains the Network Access parameter dataset. The default is PIE.PTSnnn.PARMLIB.

SIZE Specify the region size for PNAIPL. The default of 512K is more than enough.

SOUT Specify the SYSOUT class for job output. Do not make this a purged class. The default is *.

The DD statements for the PNAIPL procedure are:

STEPLIB Identify the PIE/TSO APF authorized library. STEPLIB is required until you

- add PNAIPL to your link list and refresh LLA
- add PNAIPL to an LPA library and IPL with a CLPA
- add PNAIPL the link list library and IPL with an MLPA

SYSIN Identify the Network Access parameter dataset. The dataset must be a partitioned dataset, with an LRECL of 80. The default is PARMLIB. SYSIN is required.

SYSLIB Identify the APF authorized library containing the PNAPARM module. SYSLIB is required.

SYSPRINT Identify the PNAIPL processing log. If this DD statement is not opened properly, PNAIPL will send all messages to the operator console. This dataset has a logical record length of 121 and a RECFM of FBM. If no blocksize is provided a blocksize of 1210 will be used.

SYSPRINT is optional.

SYSUDUMP Identify the dump dataset. SYSUDUMP is optional.

PNAIPL EXAMPLE

```
//*=================================================
//*
//*    INITIALIZE PIE/NETWORK ACCESS AFTER AN IPL
//*
//*=================================================
//*       *** REQUIRED PARAMETER ***
//*=================================================
//PNAIPL  PROCEDURE APF=,     <== NAME OF APF AUTHORIZED
//*                           LIBRARY CONTAINING PIE/
//*                           NETWORK ACCESS LOAD MODULES
//*=================================================
//*       *** OPTIONAL PARAMETERS ***
//*=================================================
//  ACT=START,           <== ACTION TO TAKE, MAY BE
//*                           START, STOP, REPLACE OR
//*                           TEST
//  DUMP=A,              <== SYSOUT CLASS FOR THE
//*                           SYSUDUMP DD CARD
//  MEMBER=NACCESS,      <== MEMBER OF PIE/MULTITSO
//*                           PARAMETER DATASET TO USE
//*
//  PARMDS='PIE.PTS321.PARMLIB', CONTAINS PNA/SESSIONS
//*                           PARAMETERS
//  SIZE=512K,           <== REGION SIZE FOR PNAIPL
//  SOUT='*'             <== SYSOUT CLASS FOR PNAIPL
//*                           INFORMATION AND ERROR
//*                           MESSAGES
//*=================================================
//* *** EXECUTE THE PNAIPL PROGRAM
//*=================================================
//PNAIPL  EXEC PGM=PNAIPL,
//             PARM=&ACT,
//             REGION=&SIZE
//STEPLIB  DD  DISP=SHR,
//             DSN=&APF
//SYSLIB   DD  DISP=SHR,
//             DSN=&APF
//SYSPRINT DD  SYSOUT=&SOUT
//SYSUDUMP DD  SYSOUT=&DUMP
//SYSIN    DD  DISP=SHR,
//             DSN=&PARMDS(&MEMBER)
```

Step 10: Activate and Test Network Access

Network Access is now ready to be started.

1. Verify that PIE/TSO is active.
2. Start the PNAIPL procedure to initialize Network Access. From an MVS console, enter:
   `START PNAIPL,ACT=START`

   PNAIPL will execute and issue the message

   `PNA041I NETWORK ACCESS SUCCESSFULLY INITIALIZED`

   If you receive a different message, consult the *PIE/TSO Messages and Codes Manual* for an explanation and response.

   ☞ If PNAIPL abends with a System 306-C abend, the SYSLIB dataset is not APF authorized. The Network Access modules must be copied to an APF authorized library. Modify the SYSLIB DD statement in the PNAIPL procedure accordingly, before rerunning PNAIPL. If PNAIPL abends with a System 106-F abend, your APF authorized library is probably in the link list, and has gone into extents. Correct the problem before proceeding.
3. Once you receive message PNA041I, Network Access is active. To begin using it, log off and re-log on to TSO using your PIE/TSO logon procedure.
4. Test Network Access. Enter `ACCESS` from the MultiTSO menu. You should receive the Network Access menu.
5. Select a TSO application. Since TSO presents the fewest problems to Network Access, it is best to test it first. Then go on to test the rest of your VTAM applications.
6. Continue to test Network Access by following the procedures in the Network Access tutorial, in the *PIE/TSO User Guide* or in the *PIE/TSO Administrator Guide*.

   As you test Network Access, you may need to start and stop it frequently. See “Starting and Stopping Network Access” in the *PIE/TSO Administrator Guide* for more information.

Chapter 4
Implementing Other Components

In this chapter, we assume that you are familiar with PIE/TSO customization procedures. See "Chapter 2, Customizing MultiTSO" for more information when necessary.

CONTENTS

MultiTask

MultiTask requires no special installation, except that you must be running PIE/TSO version 3.2.1. To implement MultiTask, you need only code certain start-up options in your SESSIONS member. If you like, you may code overrides for specific users in your PCSUSERS table.

Global Options

1. Edit your PARMLIB(SESSIONS) dataset. (You may want to set up an alternate SESSIONS member while you are testing MultiTask, so that existing users will not be affected by your changes until you are ready.)
2. Specify `MTTCMP=YES`. (See "Set PIE/TSO Start-Up Options," beginning on page 31, for a complete explanation of this and the following parameters.)
3. Add `MTASK=YES` to authorize use of MultiTask. Add `ASYS=YES` to authorize use of asynchronous switching. These parameters do not turn MultiTask or asynchronous switching on in a users environment.
4. If you want your background or foreground sessions to use a special performance group, use the FGPGN and BGPGN parameters. Specify a performance group number. If you don't code FGPGN or BGPGN, the session will use the performance group the user logged on with or the performance group assigned to the application with the PGM component.

EXAMPLE

In this example, we specify that MultiTask is part of our license agreement with `MTTCMP=YES`. We authorize MultiTask for all users with `MTASK=YES` and `ASYS=YES`. We don't code BGPGN or FGPGN. So we leave the performance groups as they would be ordinarily.

```
. . .
NETCMP=YESNETWORK ACCESS COMPONENT
SESCMP=YESTSO SESSIONS COMPONENT
VIECMP=YESVIEW COMPONENT
MTTCMP=YESMULTITASK
*
PIE=YESALL USERS AUTHORIZED TO USE PIE
LOGDIR=YESIF LOGON DIRECTOR IS TO BE AVAIL
LOGOFF=YESLAST SESSION LOGOFF - LOGOFF USE
PGM=YESPGM AVAILABLE
MTASK=YESAUTHORIZE MULTITASK FOR ALL USERS
ASYS=YESAUTHORIZE ASYNCH. SWITCHING FOR ALL USERS
. . .
CPUIDSYOUR NEW PASSWORDS -
000000000000=XXXXXXXXXXX  CPUID=PASSWORD PAIR
```

Individual Options

To change MultiTask options for individual users, you may add the MTASK and ASYS parameters to your PCSUSERS table. Follow standard PCSUSERS procedures, detailed in "Overriding Defaults for Users and Terminals," beginning on page 44. See "Set PIE/TSO Start-Up Options," beginning on page 31, for a complete explanation of these parameters.

EXAMPLE

You want to authorize only a few users for MultiTask. Code `MTASK=NO` in your SESSIONS member. (You may code ASYS either YES or NO.) In your PCSUSERS table, code `MTASK=YES` and `ASYS=YES` for the users and terminals you want to authorize.

```
@SUSER SP,MTASK=YES,ASYS=YES
@SUSER T54,MTASK=YES,ASYS=YES
```

AutoLock

AutoLock locks terminals when they have been idle—the user has not pressed ENTER or a PF key—for a period of time you specify. To implement AutoLock, perform the following procedures.

1. Edit your PARMLIB(SESSIONS) dataset.
2. Code the LOCK, LINPUT, LKIGN, and MAXTRY options. See “Set PIE/TSO Start-Up Options,” beginning on page 31, for information on these parameters.
3. If you want AutoLock processing alone—you don’t want to use any other PIE/TSO features—also code `MAXSES=1`.

   In addition, turn off the MultiTSO menu. Code `SET MENU OFF` in each user’s logon procedure or initial CLIST or REXX EXEC.

Performance Group Manager

Performance Group Manager (PGM) allows you to set performance groups and domains by application ID, rather than by account code or user ID, as with IBM's SRM. You assign performance group and SWAP/NOSWAP values to application IDs in the PGM table. Use this table to change the values for special commands that you want to have greater or lesser priority during a user's logon.

PGM automatically assigns performance group numbers to applications when they are executed with the PIE/TSO PEXEC command or as aliases to PEXEC. (The PEXEC command is described in the *PIE/TSO Command Reference.)* When users switch between applications, PIE/TSO switches the performance group.

☞ If you are using DFP version 2, you can only assign 16 aliases to PEXEC. If you are using DFP version 3, you can assign up to 64.

If you like, you can create special PGM tables for individual users. To do so, create a separate PGM table and assemble it using PCSPGMT2 of the CNTL dataset. Be sure to specify the alternate table name in the MEMBER parameter. Then identify the alternate table in the PCSUSERS table. See “Overriding Defaults for Users and Terminals,” beginning on page 44, for more information.

1. Edit member PCSPGMTB of the SAMPLIB dataset.
2. Code your @SPGMTAB macros. (The format is described on page 82.) You may code @SPGMTAB macros in any order, except you must code a macro with TYPE=END last.
3. Assemble and link PCSPGMTB using member PCSPGMTB of the CNTL dataset.

The @SPGMTAB Macro

```
@SPGMTAB LNAME=logical-name,
         PNAME=physical-name,
         PGN=n,
           [NOSWAP={YES},]
                   {NO }

           TYPE={PGM }
                {CMD }
                {END }
```

Parameter	Valid	Default
LNAME	1 to 8 characters	None
PNAME	1 to 8 characters	None
PGN	any numeric	None
NOSWAP	YES or NO	NO
TYPE	CMD, PGM, or END	None

LNAME Specify the logical name of the command used on the PEXEC command.

PNAME Specify the physical name of the load module actually invoked. Use the module name invoked by the TSO Service Routine.

☞ If you are going to assign the command name as an alias to PEXEC, the LNAME and PNAME must be different. (If they are the same, the command will go into a loop.) If you want, you can assign a new LNAME and use the original load module name as the PNAME. Or you can use the original LNAME and copy the original load module to a different name. The first method requires some user training. The second method complicates maintenance somewhat. If you don't assign the LNAME as an alias to PEXEC, the LNAME and PNAME may be the same. When your users execute the command, they must execute it as a parameter to the PEXEC command. All three methods are shown in the examples below.

PGN Specify the performance group number to be assigned to the session while the PNAME module is running. Any numeric value will be accepted, however the system must recognize it as a valid PGN, or PEXEC will reject it.

NOSWAP Specify whether the address space is to be made non-swappable while the PNAME module is running. NOSWAP=YES will make the address space non-swappable. NOSWAP=NO will leave the address space swappable.

TYPE Identify the type of item coded. CMD indicates a TSO command processor. PGM indicates an MVS program. Use END for the last @SPGMTAB macro. For example: `@SPGMTAB TYPE=END`

EXAMPLE 1

This example shows how to execute an application as an alias to PEXEC using a different logical name.

1. When you code your @SPGMTAB macro, assign a new command name to the LNAME parameter and the actual load module name to the PNAME parameter.

   This example substitutes XACCESS for the PIE/TSO ACCESS command name.

   ```
   @SPGMTAB LNAME=XACCESS,PNAME=ACCESS,PGN=35,                    X
            NOSWAP=NO,TYPE=CMD
   @SPGMTAB,TYPE=END
   ```

2. Assemble and link PCSPGMTB.
3. Assign the LNAME (XACCESS) as an alias to PEXEC. Use the link edit JCL in member PEXEC of the CNTL dataset.
4. Inform your users of the new command name.

EXAMPLE 2

This example shows how to execute an application as an alias to PEXEC using a different physical name.

1. Rename the original load module. In our example, we rename the ACCESS module to XACCESS.
2. When you code your @SPGMTAB macro, assign the original command name to the LNAME parameter and the new load module name to the PNAME parameter.

   The following macro shows the LNAME as ACCESS and the new load module name, XACCESS, as the PNAME.

   ```
   @SPGMTAB LNAME=ACCESS,PNAME=XACCESS,PGN=4,                     X
            NOSWAP=YES,TYPE=CMD
   @SPGMTAB,TYPE=END
   ```

3. Assemble and link PCSPGMTB.
4. Assign the LNAME (ACCESS) as an alias to PEXEC. Use the link edit JCL in member PEXEC of the CNTL dataset.

EXAMPLE 3

This example shows how to place an application in the PGM table when it is not an alias to PEXEC.

1. When you code your @SPGMTAB macro, assign the original command name to the LNAME and PNAME parameters.

   The following macro shows ACCESS as the LNAME and PNAME.

```
@SPGMTAB LNAME=ACCESS,PNAME=ACCESS,PGN=4,                    X
         NOSWAP=YES,TYPE=CMD
@SPGMTAB,TYPE=END
```

2. Assemble and link PCSPGMTB.
3. Do not assign the LNAME (ACCESS) as an alias to PEXEC. Instead, inform your users that they should execute ACCESS as a parameter to PEXEC. For example: `PEXEC ACCESS`

View

1. Create the View dataset(s). You may have one global dataset and any individual user can have or share separate View datasets. All View datasets must be partitioned datasets, with logical record length 80. They can have any block size.
2. Preallocate the View dataset(s).

   Use the VIEWDS start-up option in your PARMLIB(SESSIONS) member to pre-allocate the global dataset.

   Use the VIEWDD start-up option to indicate what ddname you will use to pre-allocate individual View datasets. Then use that ddname to pre-allocate these individual datasets in users' initial CLISTs or REXX EXECs. For example:

   ```
   ALLOC FI(PIEVIEW) DA(dataset-name) SHR
   ```

Logon Director

The Logon Director allows PIE/CICS to log users on to TSO without requiring them to enter their user IDs and passwords. The Logon Director can also save TSO passwords in encrypted form, so that PIE/CICS can use them.

The Logon Director provides the following features.

- A single user sign on. Since PIE/CICS has already validated the user's password, the Logon Director can log this user on to TSO without the password prompt.
- Automatic Reconnect. The Logon Director determines if a logon reconnect is required and handles it invisibly to the user.
- Forced Reconnect. If the user is a PIE/TSO user who used the Logon Director for the original logon, Logon Director can disconnect the user from his original terminal and reconnect to this terminal.

For more information on the Logon Director, see the *PIE/CICS Resource Definition and Management Guide*, the MultiCICS and *NetGate User Manual*, and the *Dynamic Menus User Manual*.

1. The Logon Director is installed during normal PIE/TSO installation. If you would like to install the Logon Director only, use the sample JCL in member LINSTALL of the CNTL dataset.

   LINSTALL contains a portion of the full PIE/TSO installation. Its parameters have the same meanings as those of the SINSTALL job. See Appendix B for more information. Be sure that PCSDFLD@ is installed in a link list library.

   You cannot use INSTALLS to install the Logon Director only.

2. Edit your PARMLIB(SESSIONS) dataset. Specify `LOGCMP=YES`. Also set the LOGDIR and PASENC start-up options. (See “Set PIE/TSO Start-Up Options,” beginning on page 31, for a complete explanation of these parameters.)

Chapter 5
Putting PIE/TSO into Production

When you have finished customizing PIE/TSO, perform the procedures in this chapter to put your changes into production.

CONTENTS

Step 1: Set Up SMF Recording

If you would like to know what resource savings you are gaining by running PIE/TSO, you can set up PIE/TSO SMF recording.

⚠ WARNING

Use SMF recording only on MVS/ESA version 3 and above. It will cause abends on MVS/XA.

1. Include the SMFRTY start-up option in your PARMLIB(SESSIONS) member. To record data, specify an SMF record type (128-255) that is not being used by any other application.
   Example: `SMFRTY=227`

   To turn off recording, specify `SMFRTY=NONE`.

2. Update your SMFPARMxx member. Include the record type you specified in the SMFRTY start-up option in the TSO subsystem statement.
   Example: `SUBSYS(TSO,TYPE(227))`

3. To activate the new member issue `SET SMF=xx` from the console. Substitute the last two characters of the SMFPARMxx for `xx`.

4. Run PCSINIT with ACT=NEWPARM to put your SMFRTY start-up option into effect.
   Example: `S PCSINIT,ACT=NEWPARM`

   PIE/TSO will begin recording data for all users who log on after this point. When you have gathered enough data, run the analysis program. See the *PIE/TSO Administrator Guide* for procedures.

Step 2: Automate Start-Up

The PCSINIT procedure must run successfully before users can access PIE/TSO services. In addition, PNAIPL must run successfully before users can access Network Access services.

We recommend that you start both from a COMMNDxx member of SYS1.PARMLIB during IPL.

1. Add PCSINIT to a COMMNDxx member. For example: `COM='S PCSINIT'`
2. Add PNAIPL after PCSINIT. For example: `COM='S PNAIPL'`

Step 3: IPL or Restart PIE/TSO

You must restart PIE/TSO to put your changes into effect.

If you loaded PIE/TSO into an LPA library, you must IPL with a CLPA to load PIE/TSO modules into your LPA.

If you modified your IEFUTL module in “Step 3: Implement Security Features,” you must IPL with an MLPA or CLPA to implement your changes.

If you did neither, you can simply restart PIE/TSO.

1. Do one of the following.

- If you loaded PIE/TSO into an LPA library or you modified your IEFUTL module, IPL with an MLPA or CLPA now.
- If you did not load PIE/TSO into an LPA library or modify IEFUTL, restart PIE/TSO to implement your start-up option changes. Enter `S PCSINIT,ACT=START` from an operator console or equivalent.

Step 4: Testing PIE/TSO

To test your PIE/TSO system:

1. Log on with your PIE/TSO logon procedure.
2. Start up several sessions and applications.
3. Test switching.
4. Test various PIE/TSO commands, such as SWITCH, SET MTASK ON, etc. You may use the tutorial from the *PIE/TSO User Guide* or *Administrator Guide* as a testing guide, if you like.

Step 5: Modify PIE/TSO Users' Logon Procs

When you have completed customization and testing, you may introduce PIE/TSO to the user community. To do so, update users' logon procedures as described in “Step 5: Modify Your Logon Procedure,” on page 10.

Appendix A About the Product Tape

PIE/TSO is distributed on a standard labeled tape, 3480 cartridge non-IDRC, or 6250 BPI reel.

Resource Requirements

PIE/TSO 3.2.1 requires the following common storage.

- Pageable CSA for modules: 34K
- Pageable CSA for storage: 1K
- Pageable ECSA for modules: 18K
- Pageable ECSA for storage: 256 bytes per logged on PIE/TSO user
- Fixed ECSA for modules: 7K

☞ All common storage is in Key 2 for system integrity.

PIE/TSO 3.2.1 requires the following private storage.

- Low private below the line for modules: 124K
- High private below the line for storage: 40K
- Low private above the line for modules: 101K
- High private above the line for storage: 16K

The additional system overhead for TSO users while performing their work in a MultiTSO session is so slight that it cannot be measured. In fact PIE/TSO will decrease the system resources used for the initialization and termination of TSO applications.

However TSO users' working sets increase according to the number of TSO applications they have active at the same time. Make sure that your paging/swapping subsystem has enough space to accommodate this increase in storage. The rule of thumb is to multiply the expected average number of logged on PIE/TSO users by the expected number of active sessions by the current working set size of any average TSO user. This gives the total PIE/TSO related storage requirement in bytes.

PIE/TSO can cause an increase in page stealing on systems with paging problems if more than 40 PIE/TSO users are active.

Disk Space Requirements

Dataset	Kilobytes	Blocksize
Control library	1405	6160
Load library	1827	23200
Help library	703	6160
Macro library	984	6160
Panel library	328	6160
Table library	141	6160
Message library	94	6160
Sample library	843	6160
Parameter library	94	6160
Infocard library	187	6160
CLIST library	187	6160
View library	141	6160
Total: 314 3380 tracks		

☞ The View load modules are copied into the PIE/TSO load library. The View library contains sample panel images.

Appendix B Using SINSTALL

PIE/TSO provides two methods to unload datasets from the product tape. The recommended method is the INSTALLS CLIST. However, if you want to build your own installation job, you may. If you do so, we recommend that you use member SINSTALL of the CNTL dataset as a model. This appendix describes the installation job requirements and the parameters in the SINSTALL member.

Please read this section completely, before executing the SINSTALL job.

What the SINSTALL Job Does

The SINSTALL job must perform the following functions.

- Allocate and load all remaining PIE/TSO datasets from the product tape.
- Assemble and link edit the PCSDFLD@ load module to customize it to your MVS system level.
- Assemble and link edit the PCSVPSWD load module to allow the LOCK command to support your security system.
- Copy all members of the LOAD library to an APF authorized library.
- Copy the sample PCSINIT and PNAIPL (if you want Network Access support) to a procedure library.

Customizing the SINSTALL In-stream Procedure

You must customize SINSTALL to meet your installation's requirements.

1. Edit member SINSTALL of the CNTL dataset. Modify the following parameters.

   APF Identify the APF authorized library for the PIE/TSO load modules.

   CNTL Identify the PIE/TSO CNTL dataset created in “Step 1: Load File One,” on page 3.

   OPTIONS Identify your security system. Code ACF2, RACF, RACR (for RACROUTE), or TOPS (for TOP SECRET). If none of these are installed, code NONE. This parameter must be 4 bytes long.

   PROCLIB Identify the JES procedure library for the PCSINIT and PNAIPL procedures.

   VOLSER Specify the disk volume onto which to load PIE/TSO datasets.

2. You may modify the following parameters, if you want.

   APFDISP Specify the DISP parameter of the APF dataset above. The default is OLD.

   ASMCC Specify an acceptable condition code for the assembler step. The default is 4.

   DELETE Specify a condition code under which the DELETE step should execute. If you specify 0, no delete will be done. The job stream will allocate new datasets. If you code 4, all the PIE/TSO datasets will be deleted and reallocated. The default is 0.

   DU Specify a unit for the PIE/TSO permanent datasets. The default is SYSALLDA.

   MAC Identify the first macro library in the assembly SYSLIB concatenation. The default is SYS1.MODGEN.

   MAC1 Identify the second macro library in the assembly SYSLIB concatenation. The default is SYS1.MACLIB.

   ☞ The order of concatenation is critical. The macro and sample libraries must follow the system macro libraries. The BLOCK parameter specifies the largest blocksize in the concatenation.

   MAC2 Identify the security system macro library for PCSDFLD@ and PCSVPSWD assemblies. Change this parameter only if you run ACF2 or TOP SECRET. The default of SYS1.MACLIB is correct for RACF.

   PDISP Specify the DISP parameter for the PROCLIB dataset above. The default is OLD.

   Q1 Specify the first level qualifier for PIE/TSO datasets. The default is PIE.

   Q2 Specify the second level qualifier for PIE/TSO datasets. The default is PTS321.

   SIZE Specify the region size for the IEBCOPY step. The default is 2048K.

   SOUT Specify the SYSOUT class for job output. The default is *.

   TEMP Specify the unit for the SYSUT4 IEBCOPY DD statements. The default is VIO.

   TQ1 Specify the first level qualifier for PIE/TSO products on this tape. The default is TSCPIE.

   TQ2 Specify the second level qualifier for PIE/TSO products on this tape. The default is PTS321.

   TU Specify the unit for the PIE/TSO product tape. The default is TAPE.

VIO Specify the unit for temporary datasets created. The default is VIO.

3. Submit your SINSTALL job.

   ☞ The installation job assembles PCSDFLD@. PCSDFLD@ refers to IBM control blocks and avoids MVS level dependencies in most load modules.

   You may have several MNOTE statements during PCSDFLD@ assembly. This is normal, since IBM changes the IKJEFLWA macro in different releases of TSO/E. UNICOM Systems provides for this in the macros used in the assembly, but it is not possible to totally eliminate the impact of IBM changes. All MNOTEs can be ignored up to TSO/E Release 2.4.1.

   If you use the PIE/TSO load library as the APF authorized library, the COPYAPF step will end with a condition code of 8. That is because IEBCOPY will think that a “member select” is being attempted during a compress. If this occurs the IEBCOPY will fail with a parameter validation error before doing the compress or the copy. In the above case, 8 is an acceptable return code for this job step. PCSDFLD@ must reside in an APF authorized library or implicitly authorized library, such as link list.

Index

Symbols

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

PIE/TSO Quick Start Guide

This guide describes “the least you need to know” to use PIE/TSO. We recommend that you look over the tutorials in the PIE/TSO User Guide for more complete information on this material and for information on the many important features not described here.

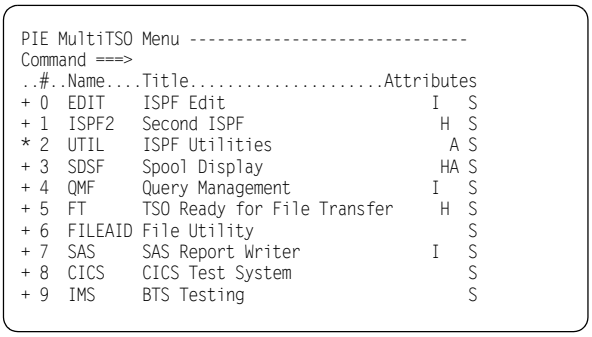
Using MultiTSO

PIE/TSO MultiTSO allows you to operate up to 12 TSO sessions on one terminal with one TSO logon.

Accessing MultiTSO

To access MultiTSO, log on to TSO using your standard logon method.

You will be presented with the MultiTSO menu, which displays information about your sessions and provides a command line for PIE/TSO commands.



Your system administrator may have set up an initial CLIST to open sessions automatically when you logon. If so, you’ll see those sessions displayed on the menu.

Your initial CLIST may automatically switch you to one of those sessions when you logon. If you are in a session, you can go to the MultiTSO menu by entering SWITCH at TSO READY.

Help

To access HELP, enter H on the MultiTSO menu command line.

Once you are in HELP, you may select any HELP member for information on a specific topic. To select a HELP member, enter its name on the HELP command line. For general information on PIE/TSO, enter PIE1.

To exit HELP, press PF3 twice.

Opening Sessions

To open a session, use the START command. Issue it from the MultiTSO menu or from TSO READY. The START command format is START [id] [tso-command]. Substitute a session's code (that is, its number or one letter code, such as A or B) for id. Substitute a TSO command and its parameters for tso-command. For example, to open session 1 and execute ISPF in it automatically enter START 1 ISPF.

You may omit either the ID (e.g., START ISPF) or the TSO command (START 1) or both (START). If you omit the ID, MultiTSO will open the next available session. If you omit the TSO command, MultiTSO will open the session and leave it at TSO READY.

Getting Around

Use any of the following methods to switch between sessions and the MultiTSO menu.

☞If you type in data but switch out before you press ENTER, that data may be lost. Press ENTER before you switch to ensure that all the data on the screen will be there when you return to the session. To prevent loss of data automatically, enter SCREEN SAVE from the MultiTSO menu command line.

SWITCHING FROM THE MULTITSO MENU

To switch to a session from the MultiTSO menu, just enter the session code or name in the command line. For example, to switch from the MultiTSO menu to session 2, enter 2 on the command line.

THE JUMP KEY

The jump key "jumps" from session to session in numerical order. By default the jump key is turned off. To turn the jump key on, set it to a PF key with the JMKEY command. JMKEY command format is JMKEY PFn. For example, enter JMKEY PF5 to turn the jump key on and to set it to PF5.

If you want to skip a session when you use the jump key, "hide" it with the HIDE command. For example, enter HIDE 1 to skip session 1. To unhide a session, use the UNHIDE command: UNHIDE 1.

THE SWITCH SEQUENCE

The switch sequence is the most versatile way to move around in PIE/TSO. The switch sequence consists of the switch character and the switch key. By default, the switch character is \ (backslash) and the switch key is ENTER. Your switch character and switch key are displayed on the right side of the MultiTSO menu.

To perform the switch sequence, type the switch character in the first position of an input field on your screen. Then press the switch key.

To switch between the MultiTSO menu and the last session you were in, issue \ [ENTER]. To switch to a particular session, specify the session code or name after the switch character. For example, to switch to session 2, issue \2 [ENTER]. To switch to a session named UTIL, issue \UTIL [ENTER].

Executing ISPF in More Than One Session

You can execute ISPF in more than one session at a time. To do so, execute MSPF instead of ISPF in the second and following ISPF sessions. (You may also execute MSPF in the first session.) For example, enter MSPF 3.2 to execute ISPF 3.2 in a session. You may combine the MSPF command with the START command to open a session and execute ISPF in it at the same time: START MSPF 3.2.

Asynchronous Switching

Asynchronous switching allows you to suspend your current application and switch to the MultiTSO menu when your application is still processing a task.

You could use asynchronous switching to move to the MultiTSO menu and cancel your task. Or, you could be in the middle of a long running command and get a call from someone needing information. You could suspend your task, get the information, and switch back to the task. When you return, your task will resume processing.

Before you can use asynchronous switching, you must turn it on. Enter SET ASYS ON ALL to turn it on in all sessions. Then, to switch out of a running session:

- From an SNA terminal: Press ATTN.
- From a non-SNA terminal: Press RESET and then PA1.

Changing Your MultiTSO Menu

You can change session names with the NAME command. For example, to change session 1's name to LIBR, enter NAME 1 LIBR.

You can also change a session's comments on the MultiTSO menu. To do so, simply type over the comment area. Your changes will remain until you logoff.

Locking Your Terminal

Use the LOCK command to secure your terminal while you are away from your desk. When your terminal is locked, no one can use it until your password is entered correctly. To lock your terminal, enter LOCK. To unlock your terminal, enter your password at the prompt. (Your terminal may get locked automatically after it has been idle. To unlock it, again, enter your password at the prompt.)

Exiting Sessions and Logging Off

To end a session, exit the application and enter LOGOFF at TSO READY.

Before you can log off TSO, you have to end all of your sessions. To take care of this with one command, enter CANCEL ALL. CANCEL ALL abends all your applications, ends all your sessions, exits PIE/TSO, and logs you off TSO.

⚠ WARNING
If there is any unsaved data in any session, it will be lost. For critical applications, be sure to save data and end normally before you use CANCEL ALL.

Using Network Access

Network Access allows you to access VTAM applications in your MultiTSO sessions.

1. To open a Network Access session, enter START ACCESS from the MultiTSO menu. MultiTSO opens a session and takes you to the Network Access menu.
2. To start an application from the menu, select it with the cursor in the command column on the left and type the letter S. Or type its menu number in the command line.

   When you press ENTER, Network Access will establish a session with the APPLID you have selected, and it will pass the information in the initial text area to the application. For instance, if the initial text area contains your user ID, Network Access will pass your user ID to the application automatically. The initial text area may contain variables (such as &ZUSER). See the ACCESS command in the PIE/TSO Command Reference for more information on these variables.
3. You may change the information in the initial text area, if you want.
4. When you are finished, press ENTER. Network Access will establish a session with your VTAM application.

Switching

Use the switch sequence or the jump key to switch from Network Access sessions.

Ending Network Access Sessions

To end your Network Access session, first log off your VTAM application. You will be returned to the Network Access menu. If you want, you can start a new VTAM application in the same session. To exit the session, issue the END command (PF3).

Long Running Tasks

Network Access sessions can process in the background while you work in other sessions. To exit out of a Network Access session running a task:

- If you are on an SNA terminal—press ATTN
- If you are on a non-SNA terminal—press the RESET key and use the switch sequence

You will switch to the MultiTSO menu. Your Network Access session will continue to process its task just as if your terminal displayed it. When you return, your screen will contain the latest updates.

Using MultiTask

MultiTask allows you to process one task in a background session (a session that you don't see), while you work in a foreground session (the session that is displayed on your screen). You must be authorized to use MultiTask.

1. You will use asynchronous switching to switch out of the background session while it is still processing a task. Before you start your long running task, turn asynchronous switching on—enter SET ASYS ON ALL.
2. Turn MultiTask on—enter SET MTASK ON id from the MultiTSO menu. Substitute the session's code or name for id. For example, to start MultiTask in session 3, enter SET MTASK ON 3. (Or enter M in the command column next to the session on the MultiTSO menu.)
3. Switch to the MultiTask session and begin your task.
4. To switch out of the session:

- If you have started a new application, wait for it to fully initialize. If you don't wait, the application could end or hang.
- If you are using an SNA terminal, press the ATTN key to exit. If you are using a non-SNA terminal, press RESET and then PA1.

5. You will go to the MultiTSO menu. From this point you can switch to any session and continue to work. When your background task is finished, PIE/TSO will send you a message.

Building PIE/TSO Sessions At Logon

If you execute an initial CLIST or REXX EXEC at logon, you can open sessions and customize your PIE/TSO environment automatically. You can issue any PIE/TSO command from a CLIST/EXEC. We review only a few here. See the PIE/TSO Command Reference for other commands.

Use the ALLOC command to open sessions in CLISTs/EXECs. The ALLOC command format is ALLOC [id] [tso-command].

You can name your sessions and assign comments to them to create a menu of sessions/applications. Use the NAME and NOTE commands to do this.

To change your switch key, switch character, and jump key, use the SWKEY, SWCHAR, and JMKEY commands, respectively.

When you are finished opening sessions and defining the environment, you can switch back to the MultiTSO menu or execute your first application in the session running the CLIST/EXEC. Use the SWITCH command to switch to the MultiTSO menu.

Code every command as needed to execute at TSO READY. Some commands, such as CREATE, execute at TSO READY by default. Most commands, such as ALLOC, need to be preceded by PIEEXEC. For example: PIEEXEC ALLOC

Be sure to name your CLIST/EXEC according to installation standards. Otherwise it will not execute at logon. See your PIE/TSO administrator for assistance.

EXAMPLE

```
PIEEXEC JMKEY PF21
PIEEXEC NAME   0 TSO
PIEEXEC NOTE   0 Native TSO
PIEEXEC ALLOCH 1 ISPF
PIEEXEC NAME   1 ISPF
PIEEXEC NOTE   1 ISPF
PIEEXEC ALLOC  2 MSPF 2
PIEEXEC NAME   2 EDIT
PIEEXEC NOTE   2 ISPF 2 (EDIT)
SWITCH
```

PIE/TSO User Guide

Release 3.2.1

UNICOM
SYSTEMS, INC.

PTSOUG321-01

Printed: October 2002

This manual applies to PIE/TSO release 3.2.1 and to all subsequent releases of the product until otherwise indicated by new editions or updates to this publication.

UNICOM Systems, Inc.
Unicom Plaza - Suite 310
15535 San Fernando Mission Blvd.
Mission Hills, California 91345

Contents

Preface

Using this Guide

What is this manual about?

The PIE/TSO User Guide contains a tutorial for the PIE/TSO components MultiTSO, Network Access, MultiTask, and View.

Chapter 1 briefly introduces you to PIE/TSO.

Chapter 2 teaches you to use MultiTSO and MultiTask.

Chapter 3 teaches you to use Network Access.

Chapter 4 teaches you to use the PIE/TSO special sessions—Help, Info, and View.

Chapter 5 introduces you to PIE/TSO commands.

For complete descriptions of PIE/TSO commands, see the PIE/TSO Command Reference. For advanced PIE/TSO features and for maintenance and operation information, see the PIE/TSO Administrator Guide. For installation and implementation, see the PIE/TSO Installation and Customization Guide.

Audience

This manual is written for PIE/TSO end users. We assume that you know how to use TSO and ISPF. You do not need to know anything about PIE/TSO.

Chapter 1
Introduction

CONTENTS

What is MultiTSO?

TSO and ISPF Without MultiTSO

With IBM TSO and ISPF you can run dozens of online applications and commands. If you are a typical user, you run several different functions throughout your day. For instance, if you are a programmer, you use one function to edit source code, another function to test programs, and another to review output. And typically you don't finish with the application you are using before you need information or functions from another. Instead, you need to alternate between several applications and datasets as you do your work.

With TSO alone, you have to stop the function you are running before you can begin another. With ISPF you fare a little better, because you can split the screen on a terminal display and alternate between two functions. For example, you can edit two files or edit a source file while looking at the compiler output.

Often, however, you need access to more than two functions at once, and if you want to split between an ISPF function and a TSO function, you're out of luck. The TSO function takes over until you quit it.

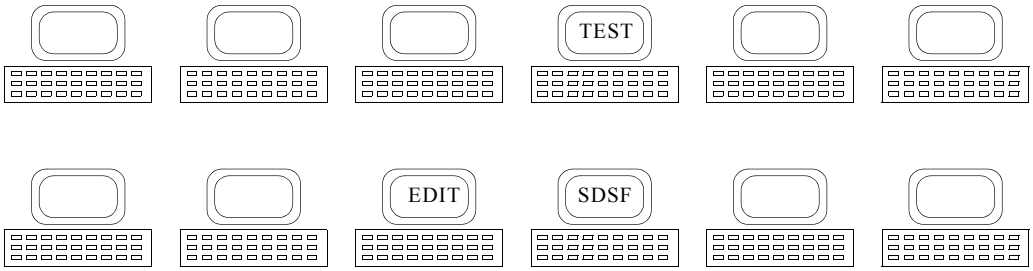
Some installations have tried to solve this problem by issuing two or more TSO logons to employees with critical need. But this is undesirable for security administration and networking.

TSO and ISPF With MultiTSO

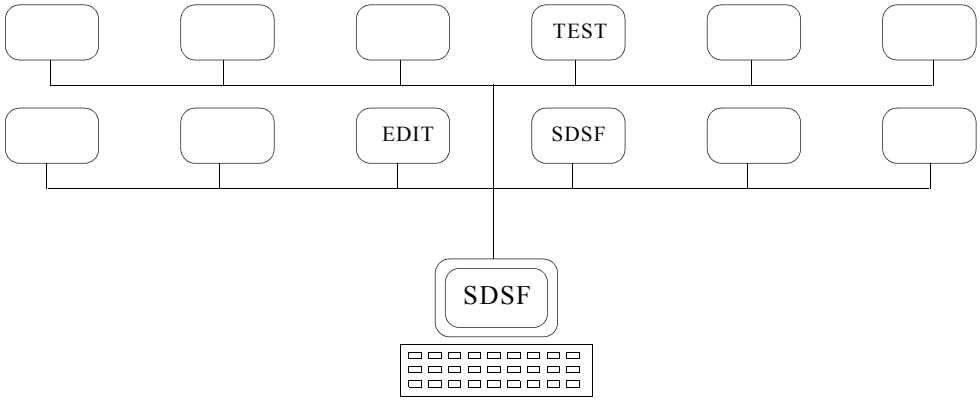
With MultiTSO, you log on to TSO with only one user ID and have access to up to twelve TSO sessions at once. You can alternate between any of these sessions at any time—even when an application is still processing a task.

MultiTSO supports everything TSO supports. So you can run FOCUS, SAS, SDSF, ASM, RMFMON, or any other TSO or ISPF command in a MultiTSO session. If you run ISPF in any or all of those sessions, you can use its split-screen capabilities. So it is possible to run twenty-four different commands at once, on the same terminal, with a single TSO ID.

To imagine MultiTSO capabilities, think of yourself sitting at a table with twelve terminals (or virtual terminals) in front of you. On one terminal you might want to run ISPF edit for your COBOL or Assembler source. On another terminal you might run a testing program. On a third terminal you might have SDSF running to see a job's output before it is printed. You're using three terminals, and you still have nine left over.



But instead of using twelve terminals, you use only one. Instead of logging on twelve times, taking up valuable MVS resources with each logon, you log on only once.



You never have to log off one function to move to another. You simply enter one of several PIE/TSO switching commands. When you return to a function, it will be just as you left it.

What is MultiTask?

With MultiTSO, you can switch out of any session—even a session that is processing a task—and work in another session. However with MultiTSO alone, processing can only occur in the session you see on your screen. Once you switch out of a session, it stops processing until you return to it.

This presents a problem when you execute a task that takes a long time to run, such as a QMF search of a DB2 database. Although you can switch out of the session now and then, if you want the task to get done, eventually you have switch to the session and wait the task out.

MultiTask solves this problem by allowing TSO to process a task in the background while you work in the foreground. As long as the foreground session is not active—that is, you have not pressed ENTER or a function key within five seconds—TSO will continue to process the task in the background.

So you can view output, type in data, write code, consider how to respond to data on the screen, and so on, while the task in the background runs undisturbed. If you press ENTER or a function key, MultiTask immediately returns control to the foreground session so that you don't have to wait. Then MultiTask waits five seconds, and if you haven't pressed ENTER or a function key in that time, it returns control back to the background session which continues to process as before.

What is Network Access?

Network Access allows you to run VTAM applications in your MultiTSO sessions. You can access any TSO, CICS, IMS, PROFS, etc., system on any CPU in your network. With Network Access, application programmers can have ISPF, SDSF, TSO, CICS, etc., available to edit, review output, submit jobs, test transactions, and compare results. Technical support personnel can maintain open sessions to TSO, online systems, fourth generation languages, problem databases, call logs, etc. End users can have fast access to multiple CICS systems or mixtures of CICS, IMS, TSO, etc. Performance analysts can monitor multiple CPUs from the same terminal.

What is View?

View captures and saves screen images from any VTAM application such as TSO, CICS, IMS, NCCF, CA-7, PROFS, etc. You can browse or change these images, and send them to other View users.

You can use View to

- help solve problems
- train new users and produce documentation
- analyze data efficiently

View can help you solve problems because it can simplify the communication process. Problems often require you to call for the assistance of another person (someone from the help desk, a programmer, technical support) on the telephone. You have to describe the problem, read out error messages, JCL, and so on, from your screen.

With View, you can send them a picture of the screen, and they can read it themselves. This saves time and effort and often can prevent the errors and oversights that occur in these kinds of situations. It's much easier for technical support to spot errors on your screen directly than to picture the problem mentally. And when they have solved the problem, they can send you a picture of the fix, instead of trying to describe it to you.

View can help with training and documentation because it provides a simple method of copying screens for manuals and visual aids.

Screen images also help you analyze data efficiently. Consider the following situations.

- Your installation is creating or modifying new screens. Your programmers need to communicate accurately and quickly with project planners and supervisors. With View, this kind of communication can take place in just a few seconds.
- You need to compare results from parallel tests. Instead of comparing the screens visually, you can send your screen images through a comparison program. This is both faster and more accurate than manual checks.
- Your performance analysts or systems programmers need to gather and compare information from multiple CPUs. Using View, they can readily correlate situations and save evidence for later analysis.

Chapter 2
Using MultiTSO

In this chapter, you'll learn the basic elements of MultiTSO. You'll learn how to log on, how to open, control, and end your sessions, and how to switch back and forth between sessions. You'll also learn to use MultiTask.

Each section provides exercises for you to try out the features for yourself. We recommend that you follow along with the exercises to learn the product “hands on.”

We assume that you are familiar with TSO, ISPF, and the IBM 3270 display terminal. If you are not, we strongly suggest that you contact your support group and become familiar with them before you proceed.

☞ MultiTSO is very flexible. You and your PIE/TSO administrator have a lot of room to change your working environment. So what we say here may not apply exactly to your installation, but the principles are the same, even if the screens and keys are different.

CONTENTS

Accessing PIE/TSO

To access PIE/TSO, log on to TSO using your standard logon method.

1. Log on to TSO now.
2. Press ENTER if TSO displays its *** prompt.

   Your logon procedure invokes PIE/TSO automatically. You'll probably see the MultiTSO menu, which looks like the screen below.

```
PIE MultiTSO Menu ---------------------------------------- (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
                                                            SwKey:   ENTER
                                                            Keys:    OFF
                                                            Mkeys:   M ON
                                                            Pfshow:  MKEYS
                                                            Asys:    OFF
                                                            Abbrev:  ON
                                                            ImpTSO:  ON
                                                            AE:      OFF
                                                            JMKey:   OFF
                                                            Screen:  SAVE
                                                            Ver :n03.02.01
                                                            Sessions.0-11
                                                            Mon   Oct 12
                                                            Date:  10/12/92
                                                            Time:  13:58:26
                                                            User:  JDOE
                                                            Term:  TERM4C15
                                                            Sysid: 820A

  F1=HELP      F2=START     F3=END       F4=CANCEL    F5=NEWS     F6=CREATE
  F7=          F8=          F9=         F10=SWKEY    F11=SWCHAR  F12=SWITCH
```

   Your administrator may have set up your logon procedure so that it opens sessions automatically. So when you log on, you see an application screen instead of the MultiTSO menu.

3. If you don't receive the MultiTSO menu, enter SWITCH from TSO READY or as a TSO command on the command line of your application. For example, from an ISPF application, enter: TSO SWITCH

   This should switch you to the MultiTSO menu. If it doesn't, see your PIE/TSO administrator.

   In the rest of this tutorial, we are going to assume that your logon invokes MultiTSO defaults and doesn't open any sessions.

The MultiTSO Menu

The MultiTSO menu is the starting point for all MultiTSO functions. You'll open your first MultiTSO session from this menu. You'll return to it again and again for information and central control. Finally, when you're ready to quit, you'll return to the MultiTSO menu for final clean up and to log off TSO.

The MultiTSO menu is divided into three sections: the header area, the information window, and the session information area.

The Header Area

The header area contains the first two lines of the screen and the PF key display at the bottom of the screen.

```
PIE MultiTSO Menu ----------------------------------------- (C) TSC, Inc 1991
Command ===>
```

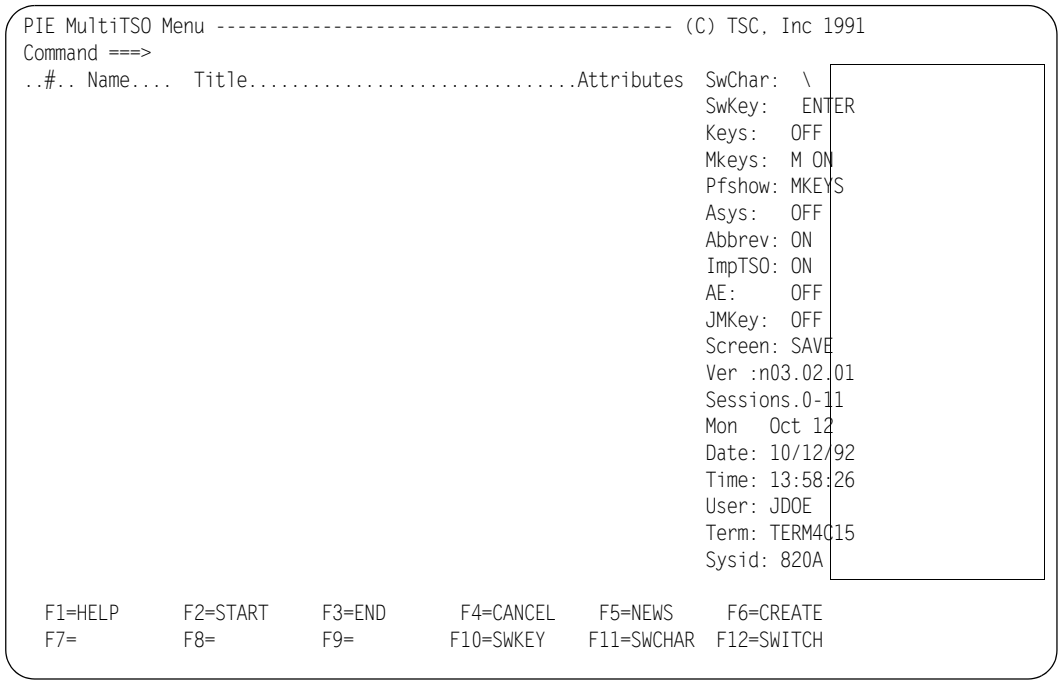
```
  F1=HELP      F2=START     F3=END       F4=CANCEL    F5=NEWS      F6=CREATE
  F7=          F8=          F9=          F10=SWKEY    F11=SWCHAR   F12=SWITCH
```

Of course, the first line provides the screen title and other identification.

The second line provides a command line. You may enter both PIE/TSO and normal TSO commands there. (Entering TSO commands is controlled by the PIE/TSO SET IMPTSO command. See the PIE/TSO Command Reference for details.)

The last line on the screen displays MultiTSO PF keys. There are two PF keys per command. PF keys 1 through 12 have the same functions, respectively, as PF keys 13 through 24. In this manual, we will only refer to PF keys 1 through 12.

The Information Window



The right side of the display is the MultiTSO information window. It displays information about your MultiTSO environment.

The following describes each field in this window. Some of the items described will be unfamiliar to you. We will describe the MultiTSO items as we go through the tutorial.

SwChar Identifies your current switch character.

SwKey Identifies your current switch key.

Keys Identifies whether your session PF keys are set on.

Mkeys Identifies how your MultiTSO menu PF keys are set.

Pfshow Identifies which PF keys are showing on your MultiTSO menu.

Asys Identifies whether you are authorized to perform asynchronous switching.

Abbrev Identifies whether you are authorized to identify sessions with shortened names (e.g., IS for ISPF).

ImpTSO Identifies how MultiTSO will respond if you execute non-PIE/TSO commands from the MultiTSO menu. If it is set to OFF, MultiTSO will issue a message stating that the command is invalid. If it is set to ON, MultiTSO will open a new session and issue the command as a TSO command.

AE Identifies whether auto-end is on or off. If it is on, auto-end will automatically end a session when the command executing in the session ends.

JMKey Identifies your current jump key.

Screen Specifies what to do if you switch to another session before you press ENTER or a PF key to enter any new data you may have typed on the screen. If Screen is set to SAVE, your new data will be saved. If it is set to NOSAVE, your new data will be lost.

Ver Identifies the MultiTSO version you are running.

Sessions Specifies the range of sessions you are allowed to access. In our example, 0-11 indicates that we can use all 12 sessions.

Date Specifies today's date in Day Month Date and mm/dd/yy format.

Time Specifies the current time.

User Identifies your TSO user ID.

Term Identifies the terminal you are working from.

Sysid Identifies your SMF system ID.

The Session Information Area

The middle section of the screen is the session information area. Initially all you will see is the title line for this section. As you open sessions, this area will display information about each session.

We will discuss this area in detail after opening some sessions.

```
..#.. Name....  Title.............................Attributes
```

Opening a Session

Now that you're familiar with the MultiTSO menu, let's open some sessions.

1. First, look at the SwChar and SwKey fields. Remember or write down what your switch character and switch keys are. This manual will assume your switch character is \ (backslash) and your switch key is ENTER.

2. Use the PIE/TSO START command to open a session. Enter START on the MultiTSO menu command line.

   The START command opens a session and switches to it—puts that session in control. You should see your initial TSO screen. This will be either the TSO READY prompt or, if you have IBM's TSO Session Manager, the Session Manager screen.

   ☞ If you switched right back to the MultiTSO menu, auto-end is on and is ending your session as soon as it hits TSO READY. For the purposes of this tutorial, turn auto-end off. Enter SET AE OFF from the MultiTSO menu command line. Now enter START again.

   ```
   READY

   _
   ```

3. From TSO READY, you can enter any TSO command—just as you would if you weren't using PIE/TSO. Let's start with a LISTC command to see what you have in your catalog. Enter LISTC. TSO will display your cataloged datasets.

4. Now we want to get back to the MultiTSO menu and open another session. MultiTSO provides several ways to switch between sessions. For now, we are going to use the switch character and switch key. This combination is called the switch sequence. Issue \ [ENTER]. (Substitute your own switch character and switch key.)

   You should be back at the MultiTSO menu. Your first session will be displayed in the information area.

   What have you just done? You have opened a session (probably session 0), used TSO from this session to list your catalog, and then switched back to the MultiTSO menu. Session 0 is now inactive—it's waiting for input from the terminal. It doesn't get any input until you switch back to it.

5. You can open a session and start a TSO application in it at the same time. To do so, all you have to do is type the TSO command and its parameters after START. Let's do that now. Enter START ISPF 2 from the MultiTSO menu command line. (If you don't use ISPF, substitute your installation's editor for ISPF. We'll assume you are using ISPF in our instructions.)

   You should be in an ISPF edit screen, because with this command you opened session 1 and executed ISPF 2 in one step.

6. Go back to the MultiTSO menu. Issue \ [ENTER] from the ISPF command line. (Substitute your own switch character and switch key.)
7. Let's open a third session. Enter START MSPF from the MultiTSO menu command line.

   You should be in ISPF again. MSPF is a special MultiTSO command that sets up ISPF files and then calls ISPF. It allows you to run ISPF in more than one session at a time. You must use MSPF to run ISPF if you already have a session running ISPF. You can also use MSPF the first time you start ISPF.
8. Switch back to the MultiTSO menu. Enter your switch sequence: \ [ENTER]

The Session Information Area

Your MultiTSO menu should look similar to the one below. It should have three sessions in the session information area. Let's go over the session information area now.

```
PIE MultiTSO Menu ------------------------------------------ (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   LISTC     LISTC                                      S  SwKey:   ENTER
+ 1   ISPF      ISPF 2                                     S  Keys:   OFF
* 2   MSPF      MSPF                                       S  Mkeys:  M ON
                                                             Pfshow: MKEYS
                                                             Asys:   OFF
                                                             Abbrev: ON
                                                             ImpTSO: ON
                                                             AE:     OFF
                                                             JMKey:  OFF
                                                             Screen: SAVE
                                                             Ver : 03.02.01
                                                             Sessions 0-11
                                                             Mon   Oct 12
                                                             Date: 10/12/92
                                                             Time: 14:01:34
                                                             User: JDOE
                                                             Term: TERM4C15
                                                             Sysid:  820A

  F1=HELP      F2=START     F3=END       F4=CANCEL    F5=NEWS     F6=CREATE
  F7=          F8=          F9=          F10=SWKEY    F11=SWCHAR  F12=SWITCH
```

The first column displays session status codes. Notice that there's an asterisk next to the 2. This means session 2 is the current session—that is, the last session you were in. The codes available in this column are:

(blank) Session not active—that is, you have opened it, but you haven't switched to it yet or you have ended it with the allocated attribute on.

+ Session is active—you have opened it and switched to it.

* Current session.

- You have executed the PIE/TSO commands END or ENDBACK in the session.

Displays the session code of the session described in the line. In our example, we have sessions 0, 1, and 2.

Name Displays the session name. If default naming is on, MultiTSO will give the session a name from the last TSO command you executed in the session. Notice that session 1 is named ISPF.

Title Displays comments. Notice that session 1 has the comment ISPF 2, and session 2 has the comment MSPF, but session 0 doesn't have a comment. MultiTSO uses the command portion of your START command as the initial comment. These comments reflect the commands you used to open these sessions.

You may add new comments or change comments at any time. If you are in the MultiTSO menu, just place the cursor in the comment field and overtype.

1. Try it now. Place the cursor in the comment area of session 0 and type: TSO Session
2. Now place the cursor in the comment area of session 2 and overtype the comment that is already there with: Second ISPF

   MultiTSO will save your comments and display them every time you switch to the MultiTSO menu.

Attributes Lists the attributes of every session. Possible attributes are:

M–MultiTask background session

I–Asynchronous switching active

H–Hidden session

K–Session PF keys active

A–Allocated session

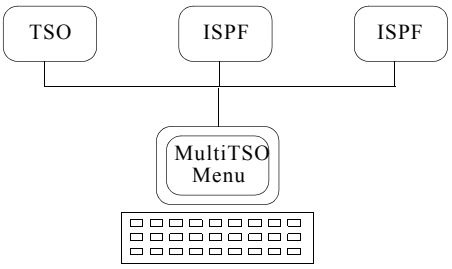
E–Auto-end active

S–Screen data will be saved when you switch out of the session

(Continue through the tutorial or turn to the glossary for an explanation of these attributes.)

Switching from Session to Session

So now you have three sessions. Session 0 is simply at TSO READY. Sessions 1 and 2 are running ISPF.



These three sessions are running in parallel. You can only see one of them at a time, and (unless you are running MultiTask) only the one you are looking at can do any processing. The other sessions are “suspended.”

To get to another session, you switch to it. You can think of sessions as channels on your TV and switching as changing channels. You can see only one channel at a time, but all channels are always available for viewing.

Because MultiTSO's primary function is to make it easy for you to get from one session to another, it provides several ways to switch. We'll discuss and practice the following switching methods:

- the switch sequence
- the jump key
- switching from the MultiTSO menu
- the SWITCH command
- switching from APL sessions

⚠ WARNING
If SCREEN (in your information window) is set to NOSAVE and you switch out of a session, you will lose any data that you have typed but haven't entered (by pressing ENTER or a PF key).

If you are running SCREEN NOSAVE, always press ENTER or a PF key before you switch out of a session. If SCREEN is set to SAVE, MultiTSO will automatically “press ENTER” before it switches.

For the purposes of this tutorial, it doesn't matter how your SCREEN option is set.

Using the Switch Sequence

The switch sequence is the most versatile switching tool in PIE/TSO. You can use it to switch to any session. You can even execute a command with it. You already used the switch sequence when you opened your sessions in the last section.

REQUIREMENTS

To perform the switch sequence, you type the switch character, then press the switch key.

You must type the switch character in the first position of an input field on your screen. You may use the TAB key to move to this position.

☞ If there is any data typed in this field but not yet entered (with ENTER or a PF key), it will be lost during the switch. This is true even when SCREEN is set to SAVE. So it is always a good idea to press ENTER or a function key before you use the switch sequence.

If you type the switch character in any other position, MultiTSO will ignore it, and it will be regarded as data by your application. So when you press the switch key, the switch character will be recorded as data, and your real data will be lost.

So, for example, if the input field is User: JDOE___, you can enter the switch character only as User: \DOE___. When you switch back, MultiTSO will restore JDOE to the field.

If you enter the switch character as User: JD\E___, the switch character will become part of the data in the field, and the O will be lost.

MultiTSO will ignore the switch character until you press the switch key. It will ignore the switch key if you don't type the switch character correctly first. That means when you press the switch key without the switch character, the program executing in that session will execute its normal function for that key. Let's say your switch key isn't set to ENTER, but to PF3. You are in your ISPF edit screen, and your PF3 key is assigned to END in ISPF. If you press PF3 before you type the switch character, ISPF will save and end out of your file, and you won't switch to another session. For this reason, it is best to avoid having critical application commands assigned to the same key as your switch key.

You can change your switch character and key. Your switch character can be any non-alphanumeric character on the 3270 keyboard. Your switch key can be ENTER or any PF key. To change your switch character, issue the SWCHAR command. For example, to change your switch character to #, enter SWCHAR #. To change your switch key, issue the SWKEY command. For example, to change your switch key to PF2, enter SWKEY 2. Enter either command from the MultiTSO menu, from TSO READY, or from an application as a TSO command.

Some applications will only allow you to input numerics on the screen. To use the switch sequence in this type of application, be sure to change your switch character to + or - before you enter the application. (These are the only characters that are valid both on these screens and as switch characters.)

OPTIONS

You can use the switch sequence to switch from any session to the MultiTSO menu or to any other session. You can also use it to switch from the MultiTSO menu to any session. (There are simpler ways to switch from the MultiTSO menu. You'll learn them later in this chapter.)

If you type the switch character by itself and press the switch key, you will switch back and forth between the MultiTSO menu and the current session. (The current session is the last active session. It is marked with an asterisk on the MultiTSO menu.) So to get to the MultiTSO menu from the current session, you would issue your switch sequence without any parameters: \ [ENTER].

To switch to another session, you will type the session ID after the switch character. For example, to switch to session 2, you would issue: \2 [ENTER] You can identify a session by code or by name. So if session 1 is named ISPF, you can switch to it by issuing \ISPF [ENTER].

You can also enter a PIE/TSO or TSO command after the switch character. If you enter a PIE/TSO command, MultiTSO will execute the command, but it won't switch you. For example, to hide all sessions from the current session, you could issue \HIDE ALL [ENTER].

If you enter a TSO command, MultiTSO will open a new session and execute the command in it. For example, to open a new session and execute SDSF in that session, issue \SDSF [ENTER]. (If you execute a command that is identical to a session name, MultiTSO will switch you to that session. It will not execute the command again.)

This means, that if you identify a session by its name and you mistype—you type a name that doesn’t exist—MultiTSO will interpret the name as a TSO command. It will create a new session and will execute the name as a command in TSO. So watch what you type!

☞ Executing TSO commands from the switch sequence requires that IMPTSO be set on. Set it on with the SET IMPTSO command. See the PIE/TSO Command Reference for details.

PRACTICE

1. From the MultiTSO menu, enter your switch sequence without any parameters: \ [ENTER]. You'll switch to session 2, the current session.
2. Issue \1 [ENTER]. You'll switch to session 1.
3. Issue \ [ENTER]. You'll switch to the MultiTSO menu. Notice that session 1 is now the current session.
4. Issue \0 [ENTER]. You'll switch to session 0.

   Notice that the display of cataloged datasets is gone. Because of TSO limitations, any line output or previous commands that were displayed on the terminal screen are lost when you switch from line mode commands. This is not true of full-screen commands.

   To avoid losing data from line mode commands, run them in Session Manager.
5. Now try executing commands with the switch sequence. Issue \NOTE 1 NEW COMMENT [ENTER].

   This will change your session 1 comment to “NEW COMMENT.”
6. Switch back to the menu to see. Enter \ [ENTER].
7. Issue \LISTA [ENTER].

   This will open a new session and issue the LISTA command in it.
8. Continue practicing the switch sequence until you can use it without having to think about it. When you are done, switch to the MultiTSO menu.

Using the Jump Key

The jump key “jumps” from session to session with just one keystroke. You don't have to enter anything on the screen. If you are in the MultiTSO menu, it jumps to the current session. Otherwise, it jumps to the next session in numeric order.

If you don't know which key is your jump key, look at the information area on the MultiTSO menu. Your JMKey is probably turned OFF, since this is the MultiTSO default.

1. Enable the jump key and set it to PF5 using the JMKEY command. Enter JMKEY PF5 from the MultiTSO menu command line.

   Your MultiTSO menu will display the jump key as: JMKey: PF5.

   ☞ The jump key is not assigned in both sets of PF keys. You can only use the single PF key displayed as your jump key.

2. You are in the MultiTSO menu. Notice which session is the current session. Now press PF5.

   You've switched to the current session.

3. Press PF5 again. You've switched to the next numeric session.

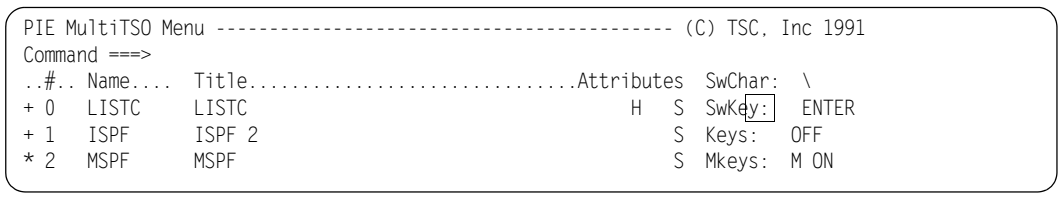
4. Continue pressing PF5 until you have seen all your sessions. Notice that you do not switch back to the MultiTSO menu.

5. You can hide sessions from the jump key. The jump key skips all hidden sessions. You can hide a session with the HIDE command. You can undo HIDE with the UNHIDE command.

   Press PF5 to get to session 0. It's at TSO READY, and you're not doing anything in particular in it. Let's hide it so you can switch between sessions 1 and 2 only. Enter: HIDE 0

6. Now press the PF5 key several times to see the new cycle. Session 0 is skipped.

7. Use the switch sequence to get back to the MultiTSO menu. Look at session 0's attributes. H is displayed for hidden.



8. Let's bring session 0 back into the loop. From the command line enter: UNHIDE 0

Notice that the H is gone from session 0's attributes.

```
PIE MultiTSO Menu ---------------------------------------- (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   LISTC     LISTC                                 H   S  SwKe     ENTER
+ 1   ISPF      ISPF 2                                    S  Keys:   OFF
* 2   MSPF      MSPF                                      S  Mkeys:  M ON
```

9. Press PF5 until you get to session 0.
10. Hide all your sessions. Enter: HIDE ALL

    This hides all your sessions in one statement. Now press the jump key. You've stayed in session 0. When there aren't any valid sessions, the jump key is ignored.
11. Practice using the jump key, hiding, and unhiding sessions. When you are done, use the switch sequence to switch to the MultiTSO menu.

Switching from the MultiTSO Menu

The easiest way to switch from the MultiTSO menu to a session is to type the session's name or code on the MultiTSO menu command line and press ENTER.

1. Try it now. Type 2 on the MultiTSO command line and press ENTER. You switch to session 2.

2. Use your switch sequence to switch back to the MultiTSO menu. Then enter 0 on the command line. You switch to session 0. Simple.

   If you are interested in the technical details, you are actually using the START command to switch. You may remember that START both opens a session and switches to it.

   We didn't try this earlier, but you can request a particular session with the START command. For example, you can enter START 4 to open session 4 and switch to it. If the session you request is already open, START will simply switch you to it. It won't open a new session.

   START is the default command on the MultiTSO menu. So if you type parameters on the MultiTSO menu command line press ENTER, MultiTSO will execute them as parameters to the START command.

   This boils down to the simple fact that when you enter a session code or name from the MultiTSO menu command line, you'll switch to that session.

3. Switch back to the MultiTSO menu and enter MSPF on the command line.

   You switch to session 1, which is named MSPF.

4. In all PIE/TSO commands, you can use an asterisk (*) instead of a session ID to get to the current session. Switch back to the MultiTSO menu and enter * on the command line.

   You switch back to session 1, the last session you accessed.

5. Practice switching from the MultiTSO menu command line. Finish in session 0, your TSO READY session.

Using the SWITCH Command

You can use the SWITCH command to switch to the MultiTSO menu or any session.

You can issue SWITCH from the MultiTSO menu or from TSO READY or from an application as a TSO command (e.g., from ISPF as TSO SWITCH). You can also switch to a session from the MultiTSO menu by placing the cursor in the command column next to the session, typing S, and pressing ENTER.

When you issue SWITCH from a session, you'll be switched to the MultiTSO menu. If you add a session ID, you'll switch to that session. When you enter SWITCH from the MultiTSO menu command line, you'll be switched to the current session.

1. From TSO READY, enter SWITCH. You're back at the MultiTSO menu.
2. At the command line, enter SWITCH. You're back at the current session.
3. This time, switch to session 2 by entering SWITCH 2.
4. Now go back to the MultiTSO menu. Because you are switching from ISPF, enter TSO SWITCH.
5. Move the cursor to session 1 on the menu. Type S and press ENTER. You'll switch to session 1.
6. Continue practicing with the SWITCH command until you have it. Finish at the MultiTSO menu.

Switching from APL/ADRS Sessions

MultiTSO fully supports APL and any products that are based on APL, such as ADRS or the Yale Full Screen Editor.

While in APL mode, valid switch characters are period (.), comma (,), and forward slash (/).

To switch out of APL sessions, use the following procedure.

1. Turn APL mode off.
2. Use the Home key to move the cursor to the first input field on the screen.

   If you prefer, you may clear the screen instead of moving the cursor to the Home position. When you switch back to the APL/ADRS session, all input will be redisplayed.
3. Enter the switch sequence to switch to the session you want.

   For ADRS sessions, use the same procedure. If the keyboard does not have APL mode, skip step 1.

Identifying Sessions

Sessions have both codes and names.

The code is displayed on the left side of the session on the MultiTSO menu. It can be the numbers 0 through 11, or it can be one alphabetic character. The alphabetic codes are:

- A–Session 10
- B–Session 11
- H–Help session
- I–INFO session
- V–View session
- X–MultiTSO menu

The session name is displayed in the Name column on the MultiTSO menu. If default naming is on (using the SET DFLTNAME command), MultiTSO will automatically give each session a name. That name will be the first word of the last command you entered in the session. For instance, if you executed MSPF 3.2 in a session, that session's name would be MSPF.

The trouble with default names is that they are always changing. Each time you execute a new command, the session gets a new name. You can solve this problem by assigning a permanent name to a session with the NAME command. Assigned names don't change until you change them with another NAME command.

1. Try it now. From the MultiTSO menu command line, enter NAME 1 EDIT.

   Your session 1 line should now read:

   ```
   + 1   EDIT      ISPF 2
   ```

2. If Abbrev is on, then you can abbreviate session names in commands and from the MultiTSO menu. If it isn't on already, turn it on now. Enter SET ABBREV ON.

3. Now switch to your EDIT session by typing E on the command line.

There are some tricks about session naming and name abbreviations. When you request a session, MultiTSO doesn't know if you are specifying a command, a session code, a session name, or an abbreviated name. It goes through a matching sequence every time you send it a string.

- First it tries to match your string with a command. (For example, if you type HIDE, MultiTSO will execute the HIDE command.)
- Next it tries to match a session code.
- Next it tries to find a session with the exact name you typed.
- Last it tries to find a session that begins with the characters you typed.

When you name a session or abbreviate a session name, you have to make sure that it will be the first match in this sequence.

For example, it wouldn't do any good to give a session the same name as a PIE/TSO command, because the command match comes first. For example, if you named a session HIDE, MultiTSO would try to execute the HIDE command every time you entered HIDE to identify the session. Moreover it wouldn't help to name a session HI, because that's the short form of the HIDE command.

In addition, you need to be careful about those alphabetic session codes: A, B, H, I, V, and X. All these session codes will match before a session name or abbreviation—so you shouldn't use them as session names or abbreviations.

Now suppose you have one session named ACTION and another named ACTIONA. You enter SWITCH ACTION. You will switch to ACTION, not ACTIONA. However, if you enter SWITCH ACT, you will switch to whichever session is first on the MultiTSO menu.

Whenever in doubt, use the whole session name or its code!

In addition to session names and codes, there are two variables for identifying sessions: * (asterisk) and ? (question mark). You've seen the * before. It identifies the current session. The ? identifies the last session you opened. It is useful in CLISTs and REXX EXECs and in user programs.

1. Enter * on the MultiTSO command line. You switch to the current session.
2. Switch back to the MultiTSO menu and enter ? on the command line. You switch to session 3 because that was the last session you opened.

Exiting Sessions and Logging Off

You've probably had more than enough of switching. How do you end sessions and end out of MultiTSO altogether?

There are several ways to terminate your sessions in MultiTSO. They are variations of the TSO LOGOFF and CANCEL commands.

1. To end a session, first exit the application in the session. Then enter the LOGOFF command at TSO READY. Try this now. Session 0 is already at TSO READY. So go to session 0 and enter LOGOFF.

   You will end up back in the MultiTSO menu. Notice that the line for session 0 is gone from the session information area.

2. Now enter LOGOFF from the MultiTSO menu command line.

You should see an error message like the one in the screen below.

```
PIE MultiTSO Menu ------------------------------------------ (C) TSC, Inc 1991
Command ===>
Please terminate active sessions first or enter CANCEL ALL      SwChar:  \
+ 0   LISTC     LISTC                                       S   SwKey:   ENTER
+ 1   EDIT      ISPF 2                                      S   Keys:   OFF
* 2   MSPF      MSPF                                        S   Mkeys:  M ON
                                                               Pfshow: MKEYS
                                                               Asys:   OFF
                                                               Abbrev: ON
                                                               ImpTSO: ON
                                                               AE:     OFF
                                                               JMKey:  PF5
                                                               Screen: SAVE
                                                               Ver : 03.02.01
                                                               Sessions.0-11
                                                               Mon   Oct 12
                                                               Date: 10/12/92
                                                               Time: 14:02:44
                                                               User: JDOE
                                                               Term: TERM4C15
                                                               Sysid:  820A

  F1=HELP      F2=START     F3=END       F4=CANCEL    F5=NEWS     F6=CREATE
  F7=          F8=          F9=         F10=SWKEY    F11=SWCHAR  F12=SWITCH
```

MultiTSO won't allow you to log off TSO until you have ended all your sessions.

Terminating up to 12 sessions individually can be time consuming. To terminate your sessions quickly, execute the CANCEL command from the MultiTSO menu. CANCEL will abnormally terminate the application in each canceled session.

You can cancel a single session or several sessions or cancel all your sessions at once with CANCEL ALL. CANCEL ALL also logs you off TSO.

⚠ WARNING
If you leave any unsaved data in any session, it will be lost when you cancel the session. If the ISPF recovery option is enabled, you may be able to recover your data the next time EDIT is invoked.

It is always best to switch into any session that may have unsaved changes or critical programs executing. Then you can save your changes and terminate programs normally in these sessions. When you are finished with critical sessions, you can safely use CANCEL ALL to abend all remaining sessions.

3. We don't have any critical sessions active in our tutorial—the CANCEL command is safe for our use. So to terminate session 1, from the MultiTSO menu command line, enter CANCEL 1.
4. We only have two sessions left. Let's create some new ones quickly to practice terminating them. From the MultiTSO menu command line, enter CREATE.

   Notice that MultiTSO opened a session but did not switch you to it. It also picked session number 1—the next available session ID.
5. Reissue the CREATE command twice, so that you have five sessions total.
6. Now use the CANCEL command to terminate sessions 0 and 1. From the MultiTSO menu command line enter CAN 0 1.
7. You can also cancel a session by entering P (for purge) next to the session in the column on the left side of the MultiTSO menu. Cancel session 2 this way now. Tab over to session 2, type P, and press ENTER.
8. Now use the CANCEL command to terminate your remaining sessions—3 and 4—and to log you TSO. From the MultiTSO menu command line enter CAN ALL.

   MultiTSO cancels all your sessions and logs you off TSO.
9. Log on again to continue with the tutorial.

Logging Off and Back On

Sometimes you need to end all your current sessions and to log on to TSO again with the same user ID. The LOGON command will do this in one step.

When you enter the LOGON command from the MultiTSO menu command line, MultiTSO will:

- cancel all your sessions (abend them, just as if you had executed CANCEL ALL)
- prompt you for your user ID and password

1. To try it, first create a few sessions. From the MultiTSO menu command line enter CREATE twice.
2. Now enter LOGON.
3. Enter your user ID.
4. Enter your password.

   You are back in MultiTSO and all your sessions have been canceled.

Asynchronous Switching

Asynchronous switching allows you to switch out of a session when the application running in it is still processing a task. When you switch out, you will go to the MultiTSO menu. From there you can perform any of your usual tasks. For instance you can switch to any of your existing sessions or open new sessions and run any TSO application.

The session you switch out of is suspended until you return to it. Once you return, the application will continue to process its task as if it had never been interrupted.

You can use asynchronous switching to:

- suspend a long running task to answer a question
- set up a dump dataset for a task that failed so you don’t have to recreate the problem
- cancel a task that has gone into a loop

There are several circumstances under which asynchronous switching will not work. It is disabled:

- at the VTAM page prompt (***)
- by certain enqueues
- by a system must complete condition
- by a step must complete condition

Most of the time you will not be able to tell that you are in a system or step must complete condition or that the enqueue in effect prohibits switching. So if asynchronous switching doesn't work, don't assume that it is not working properly. Simply wait a moment or two and try it again.

1. Before you can use asynchronous switching, you must turn it on.

   To turn it on in all sessions, enter SET ASYS ON ALL from the MultiTSO menu.

   To set asynchronous switching on in particular sessions, enter SET ASYS ON ids. Identify the sessions with the ids parameter. Then enter SET ASYS ON.

2. Switch to a session and start a long task, such as a long query of a database.

3. If your application is initializing, wait for it to finish. (If you switch out while an application is initializing, it will probably abend when you switch back.)

   While your long running task is processing, switch out of the session:

   - From an SNA terminal—press ATTN
   - From a non-SNA terminal—press RESET and then PA1

   ☞ When asynchronous switching is on, it overrides the normal use of your ATTN key (or RESET/PA1 keys) for the application. To attention out of your application, press ATTN twice (or RESET, PA1, PA1).

4. You have switched to the MultiTSO menu. Switch back to your suspended session, and it will resume processing.

5. Turn asynchronous switching off. Enter SET ASYS OFF.

Using MultiTask

MultiTask allows you to process a TSO application in a background session (a session that you don't see) while you work in a foreground session (a session displayed on your screen).

1. You must turn MultiTask on the session you want to be your background session. Enter SET MTASK ON id from the MultiTSO menu. Or place the cursor in the command column next to the session and enter M. (The session you used to query your database would be a good one to identify here.)

   MultiTask will be activated in the session you specify. Once you activate it, MultiTask will remain on in the session until you activate it in another session or you turn it off. To turn it off, enter SET MTASK OFF.

   ☞ You must be authorized to use MultiTask. If you are not, see your PIE/TSO administrator.

2. Now turn asynchronous switching on in your background session. Asynchronous switching must be on so that you can switch out of the session while the application in it is processing.

   Set asynchronous switching on for all sessions. Enter SET ASYS ON ALL.

3. Switch to your background session and start a long running task, such as a long query of a database.

4. Now use asynchronous switching to switch out of the session while the task is processing. (Remember not to switch out while an application is initializing.)

   - From an SNA terminal—press ATTN
   - From a non-SNA terminal—press RESET and then PA1

5. You are in the MultiTSO menu. You can open a new session and work in it, work in an existing session, anything.

   As you work in your foreground session, your background session will continue to process. (For details on background processing, see “More About MultiTask,” on page 32.)

6. Wait 30 seconds or a minute and switch back to your background session to see what is going on. You will see the latest screen updates. Switch back out to continue working in other sessions.

   ☞ With MultiTask, you can switch in and out of background tasks as often as you want. However don't do so too quickly. You can cause the task to fail.

More About MultiTask

USING MULTITASK EFFICIENTLY

Your foreground session has processing priority, so that you don't have to wait while you work. That means, whenever you press ENTER or a function key, MultiTask gives control to the foreground session.

MultiTask then waits five seconds before it reactivates the background session. The wait time helps minimize the number of times MultiTask switches control back and forth and increases efficiency.

So the more you press ENTER or a function key, the longer your background session is going to take to finish its task. You can type data, view it, and think about it as much as you choose. Only pressing ENTER or a function key affects background processing.

In general, you can expect a task running in the background session to take more wall-clock time to run than it would if it ran in your foreground session. How much longer depends on the amount of processing going on in your foreground session. For example, let's say your background task takes about eight minutes to complete when it isn't interrupted by a foreground session. But you switch out, and your foreground session takes control now and then for a total of four minutes. Your background task will now take twelve wall-clock minutes to run. CPU time is unaffected.

HOW DO I KNOW WHEN THE BACKGROUND TASK IS FINISHED?

When the background task is finished, MultiTask sends the message

```
PCS300I MULTITASK SESSION COMPLETE
```

to the foreground session to notify you.

HOW CAN I TELL WHICH SESSION IS MY BACKGROUND SESSION?

If you don't know which session is your background session, look at the MultiTSO menu. Your background session will have an “M” displayed as a session attribute.

```
PIE MultiTSO Menu ------------------------------------------ (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   LISTC     LISTC                                     S  SwKey:   ENTER
+ 1   EDIT      ISPF 2                                    S  Keys:   OFF
+ 2   MSPF      MSPF                                      S  Mkeys:  M ON
+ 3   QMF       DB2 Online Queries                    M   S  Pfshow: MKEYS
* V   VIEW                                              S  Asys:   ON
                                                           Abbrev: ON
```

WHAT HAPPENS IF I TURN MULTITASK OFF?

When you turn MultiTask off, the current background session stops processing. It will still exist, and it will resume processing as soon as it becomes the current session again.

Chapter 3
Using Network Access

This chapter explains how to use Network Access. We assume that you are familiar with the MultiTSO features described in Chapter 2, “Using MultiTSO."

CONTENTS

Invoking Network Access

Network Access allows you to access VTAM applications in your MultiTSO sessions. You invoke Network Access with the ACCESS command.

ACCESS is a TSO command. So you can issue it:

- from TSO READY
- in a CLIST or REXX EXEC
- from the MultiTSO menu (if SET IMPTSO=ON)

1. Let's open a Network Access session now. Enter START ACCESS on the MultiTSO menu command line.

   MultiTSO takes you to the Network Access menu, similar to the example below. Of course, your menu will be different. It will list the applications you have available at your installation. In fact, different users may have different Network Access menus.

```
PIE/TSO ---------------------- Network Access --------------- (C) TSC, Inc 1991
Cmd ==>
PF1/13 = Help   PF3/15 = END   PF7/19 = up   PF8/20 = down       Ver: 03.02.01
SELECT AN APPLICATION TO ACCESS
--#---System---- Description -------------------- Initial Text ----------------
  1   TSO        TIME SHARING OPTION
  12  JPODB      PRODUCTION DATABASE
  13  CICSA      PERSONNEL MANAGEMENT SYSTEM
  14  PROFS      ELECTRONIC MAIL
  15  IIN        IBM INFORMATION NETWORK

****************************** BOTTOM OF DATA ********************************
```

The Network Access Menu

The Network Access menu provides easy access to your VTAM applications.

It is divided into two sections. The first section presents the title line, a command line, and an information line where messages and PF keys are displayed.

The second section displays the VTAM applications available. The fields in this section are:

-- The command column. To select an application, you can type S in this column next to the application you want to select.

The APPLID menu number. You may enter this number in the command line to select an application.

System The VTAM APPLID.

Description A description of the application.

Initial Text The default parameters that will be executed if you access this application. (This will be explained in more depth later.)

Help

You can use the HELP command (PF1) to access PIE/TSO Help from the Network Access menu. See “Using the HELP Session,” on page 40, for information on PIE/TSO Help. The Help members for Network Access are NETACCSS and ACCESS.

Getting Around

Scroll up and down with the standard scrolling keys—PF7 (UP) to scroll up and PF8 (DOWN) to scroll down.

When you scroll, all but one data line will move up or down. To scroll only 12 data lines, issue HALF [PF7 or PF8] or H [PF7 or PF8] from the command line. To scroll to the beginning or end of the menu, issue MAXIMUM [PF7 or PF8] or M [PF7 or PF8] from the command line.

Exiting

To exit the Network Access menu, use the END command (PF3).

Starting an Application

You can start an application by selecting it from the menu.

1. Select the application with the cursor in the command column on the left. Type the letter S. Or you can select it by typing its menu number on the command line.

   In a moment, when you press ENTER, Network Access will establish a session with the APPLID you have selected. It will pass that application the information in the initial text area. For instance, if the initial text area for that application contains your user ID, Network Access will pass your user ID to the application automatically. Your initial text area may contain variables. For information on ACCESS variables, see “ACCESS” in the PIE/TSO Command Reference.

2. You may change the information in the initial text area, if you want. TAB over to the initial text area and type your user ID and logon procedure. (You can't supply passwords in the initial text area unless you are using the PIE Logon Director.)

   Your menu line will look something like the following:

```
s 1   CICS       CICS TEST             george proc(gfd037)
```

3. Now press ENTER.

   Network Access will execute the application. You can enter your password and use the application.

Switching

To switch from Network Access sessions, you can use the switch sequence. The switch sequence works exactly the same in Network Access sessions as it does in standard MultiTSO sessions.

However in some cases you may use Network Access to log on to TSO on another system or with another user ID. If your second TSO logon also accesses MultiTSO, you won't be able to use the same switch sequence to move around inside the second MultiTSO session.

To get around this, change either the switch key or switch character in the second MultiTSO session and use the new switch sequence to move around inside the second MultiTSO session. Use the original switch sequence to return to your original MultiTSO session.

Network Access and the MultiTSO Menu

You can run multiple Network Access sessions at the same time. PIE/TSO allows you to run up to twelve. Your installation may have set a lower limit.

Each Network Access session will be displayed on the MultiTSO menu. The screen below shows a MultiTSO menu with a second TSO session, two CICSTST sessions, and a CICSPRD session.

Notice that the Name column displays the VTAM APPLID for the CICS sessions (2 through 4). Yet the Name for the TSO session is TSOB010, even though the APPLID is TSOB. Because of MVS/TSO interfaces, the application name in the Name column and the TSO APPLID will be different. For non-TSO sessions, they will be the same.

If you like, you can use the NAME command to change the names of Network Access sessions.

```
PIE MultiTSO Menu ------------------------------------------ (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   LISTC     LISTC                                 I    S  SwKey:   ENTER
+ 1   TSOB010   ACCESS TSOB                           I    S  Keys:   OFF
+ 2   CICSTST1  ACCESS CICSTEST1                      I    S  Mkeys:  M ON
+ 3   CICSTST2  ACCESS CICSTEST2                      I    S  Pfshow: MKEYS
* 4   CICSPRO1  ACCESS CICSPROD1                      I    S  Asys:   ON
                                                              Abbrev: ON
                                                              ImpTSO: ON
                                                              AE:     OFF
                                                              JMKey:  OFF
                                                              Screen: SAVE
                                                              Ver : 03.02.01
                                                              Sessions.0-11
                                                              Mon   Oct 12
                                                              Date: 10/12/92
                                                              Time: 14:21:00
                                                              User: JDOE
                                                              Term: TERM4C15
                                                              Sysid:  820A

  F1=HELP      F2=START     F3=END       F4=CANCEL    F5=NEWS     F6=CREATE
  F8=          F9=          F10=SWKEY    F11=SWCHAR   F12=SWITCH
```

Starting a Session and Application in One Step

For quicker access, you can bypass the Network Access menu and start Network Access and the VTAM application in one step. To do so, you will issue the ACCESS command with parameters. The parameters are:

```
ACCESS [applid] [data]
```

Use applid to specify which VTAM application you want to use (for instance, ACCESS CICSTST1). Network Access will bypass the Network Access menu and connect you directly to the application you specify. You will see your normal initial screen for that application.

Use data to pass initial text to the VTAM application. For instance, to pass your user ID and logon procedure to CICSTST1 enter:

ACCESS CICSTST1 GEORGE PROC(GFD037)

Start a Network Access session and a VTAM application in one step. From a new session at TSO READY, enter ACCESS applid userid. (Substitute a valid VTAM APPLID and your user ID.) When you enter your password, you'll be ready to use the application.

Ending Network Access Sessions

To end a Network Access session, you must first log off your VTAM application.

If you started the APPLID from the Network Access menu, you will be returned to the Network Access menu. If you want, you can start a new VTAM application in the same session. To exit Network Access, you will use the END command (PF3).

1. Exit your first Network Access session now. Log off the application.
2. You are returned to the Network Access menu. Press PF3 to exit Network Access.
3. If you are still in a MultiTSO session, log off the session to exit it.

   + Your installation may allow you to use the CANCEL command to end Network Access sessions. However this method is not recommended. Some VTAM applications, including CICS and IMS, do not always sign you off when you cancel them. So it is better to sign off and end your Network Access session than to cancel it.

Long Running Tasks

Network Access sessions can process in the background while you work in other sessions. To exit out of a Network Access session running a task:

- If you are on an SNA terminal—press ATTN.
- If you are on a non-SNA terminal—press the RESET key and use the switch sequence.

You will switch to the MultiTSO menu. Your Network Access session will continue to process its task just as if your terminal displayed it. When you return, your screen will contain the latest updates.

(If you are at an SNA terminal and you want to use the ATTN key in your application, press ATTN twice. If you are at a non-SNA terminal, you must set your ATTN key function using the SET AATN or SET APFn command.)

SET AATTN

SET AATTN assigns the action of the ATTN key in Network Access sessions. Use it to determine whether ATTN is processed as normal in your VTAM APPLID or whether it switches you to the MultiTSO menu.

To set the ATTN key in Network Access sessions to the SWITCH command, enter: SET AATTN SWITCH

To pass the ATTN key through Network Access to the application, enter: SET AATTN ATTN

Chapter 4
Special Sessions

MultiTSO provides special sessions that don't count toward your twelve maximum sessions. The special sessions are:

- HELP–provides help on all PIE/TSO commands.
- INFO–provides general programming information, such as hex conversion tables.
- View–allows you to capture and manipulate screen images.

All of these sessions run in full-screen format. You will learn how to use these sessions in this chapter.

CONTENTS

Using the HELP Session

PIE/TSO provides help in a special session—the HELP or H session.

1. To open the HELP session, enter H from the MultiTSO menu command line. Your screen will look like the one below.

```
PIE MultiTSO ------------------- Tutorial ----------------- (C) TSC, Inc 1991
Command ===> ------------------------------------------------------------------
PF: 1 Help   3 END   4 RETURN   7 UP   8 DOWN   10 PREV   11 NEXT   12 SWITCH
************************** TABLE OF CONTENTS **********************************
?HELPDIR  ACCESS    ALLOC     ASYNCHSW  CANCEL    CLISTS    CLOSE     CMDBASIX
CMDLIST   CMDQUICK  CMDSYNTX  CREATE    DISC      DISPLAY   ENDBACK   ENDCMD
EXITING   FENCE     FREE      GLOSSARY  H         HELPSES   HIDE      I
INFO      INTRO     JMKEY     KEEP      LOCK      LOGON     MENU      MLTITASK
MPFL      MSPF      NAME      NETACCSS  NEWFEAT   NOTE      PEXEC     PFKEYS
PFSHOW    PIECHECK  PIEEXEC   PIESTART  PUSH      PUSHJUMP  RECALL    RESHOW
RESTART   SCREEN    SESSIDS  *SESSIONS  SETAATTN  SETABBRV  SETAE     SETAPF
SETASYS   SETCMDDF  SETCMDFL  SETDFTNM  SETHEADR  SETIMPTS  SETINFOW  SETKEYS
SETLCKIP  SETLCKTM  SETMENU   SETMKEYS  SETMPA3   SETMPA3L  SETMPF    SETMTASK
SETPARSR  SETPA3    SETPA3L   SETPF     SETPFL    SETPSWD   SETSESAT  SETTITLE
START     STOP      SWCHAR    SWITCH    SWITCHNG  SWKEY     SWTCHBCK  SWTCHNXT
UNHIDE    V         VIEW      VIEWCMDS
```

Scrolling in HELP

1. Scroll up and down with the normal scrolling keys—PF7 (UP) to scroll up and PF8 (DOWN) to scroll down.
2. When you scroll, all but one data line will move up or down. You can change the number of lines that scroll with the HALF or MAXIMUM parameters.

   To scroll only 12 data lines, issue HALF [PF7 or PF8] or H [PF7 or PF8] from the command line.

   To scroll to the beginning or end of the HELP member, issue MAXIMUM [PF7 or PF8] or M [PF7 or PF8] from the command line.

Selecting HELP Members

HELP offers several ways to access HELP members.

- HELP presents a list of members. Enter a member name from the HELP command line. For instance, VIEW is the HELP member on the View component. To request this member, enter VIEW on the HELP command line.

  Or you can view any member of this list by selecting it with the cursor and pressing ENTER. One member of the list will have an asterisk (*) next to its name. To select this member simply press ENTER.

- To get a guide to HELP members, enter ?HELPDIR.
- If you are already in a HELP member and you want to jump to the next member in the HELP list, press PF11. This is the same as entering the NEXT command. Press PF10 to jump back to the previous member. This is the same as entering the PREV command.
- Press PF3 to exit any HELP member.

Exiting HELP

To exit HELP, switch back to the MultiTSO menu or press PF4 (RETURN).

HELP on the Fly

If MultiTSO detects any errors in a command you submit, it will issue a one line error message. If you want more information on the command, press PF1.

Data Truncated

In some cases, you may notice that HELP does not display all of a HELP member. The last line of the displayed will read

```
DATA TRUNCATED - END OF INSTALLATION DEFINED BUFFER
```

The MultiTSO HELP processor uses a fixed size buffer dynamically obtained from your region. Your PIE/TSO administrator can limit the size of this buffer. If you see the above message, the HELP member is too large for this buffer.

Using the INFO Session

PIE/TSO also offers general information in the INFO or I session.

INFO members include:

- hexadecimal to decimal conversion
- translation of instruction OP Codes (similar to IBM's green card)

To open the INFO session, enter I from the MultiTSO menu command line. The INFO index will be displayed.

INFO also has a directory of members. To display this directory, press PF3.

Navigating in INFO is exactly the same as navigating in HELP. See “Using the HELP Session” for more information.

To exit the INFO session, switch to the MultiTSO menu or press PF4 (RETURN).

```
PIE MultiTSO -------------------- Tutorial ------------------ (C) TSC, Inc 1991
Command ===>
PF: 1 Help   3 END   4 RETURN   7 UP   8 DOWN   10 PREV   11 NEXT  12 SWITCH
****************************** TOP OF DATA ***********************************
*THIS IS AN INDEX OF MEMBERS IN INFOCARD          *

 MEMBERNAME     EXPLANATION
 ==========     ===========
 ALPHABET       International Phonetic Alphabet
 CODES1         x'00' - x'3F' op codes, decimal values
 CODES2         x'40' - x'7F' op codes, decimal values
 CODES3         x'80' - x'BF' op codes, decimal values
 CODES4         x'C0' - x'FF' op codes, decimal values
 HEX            hexadecimal <==> decimal conversion
 OPSTDHEX       Hex listing of opcodes for standard ESA/370 instructions
 OPSTDMEM       Listing by Mnemonics for standard ESA/370 instructions
 PIE            PIE commands reference card
 POWERS         Powers of 2, powers of 16
 PRTCHARS       Machine readable special character print standards.
 VIEW           VIEW commands reference card
****************************** END OF DATA ***********************************
```

Using View

The View Environment

View is a another PIE/TSO special session. Its session ID is V. You may only have one View session at a time.

Your View session consists of three components:

- the View menu
- the View buffers
- the View dataset

THE VIEW MENU

The View menu is a display of the members in your View dataset. It is the starting and ending point for all View functions.

THE VIEW BUFFERS

The View buffers are used to store unsaved screen images. You have two buffers and each can store one screen image at a time. One buffer, called the home buffer, stores a newly captured screen image. The other buffer, called the inter-user buffer, stores a screen image that you are sending to or receiving from another user.

Every time you send an image to one of these buffers, the new image replaces the old image. To prevent losing images in your buffers, we recommend you save them promptly. You will learn how to do that in this tutorial.

THE VIEW DATASET

When you save a screen image, it is saved as a member of the View dataset, a partitioned dataset.

You may use the default, global View dataset that anyone else can use. However it is probably better to have your own dataset, so you have total control over the screen image members.

If you want to allocate your own View dataset, you may allocate it manually or automatically, in your initial CLIST or REXX EXEC. You must allocate it before you start View. Before you allocate the dataset, first create it as a partitioned dataset with a logical record length of 80 and any block size.

Then to allocate it manually, from TSO READY, enter:

ALLOC FI(PIEVIEW) DA(dataset-name) SHR

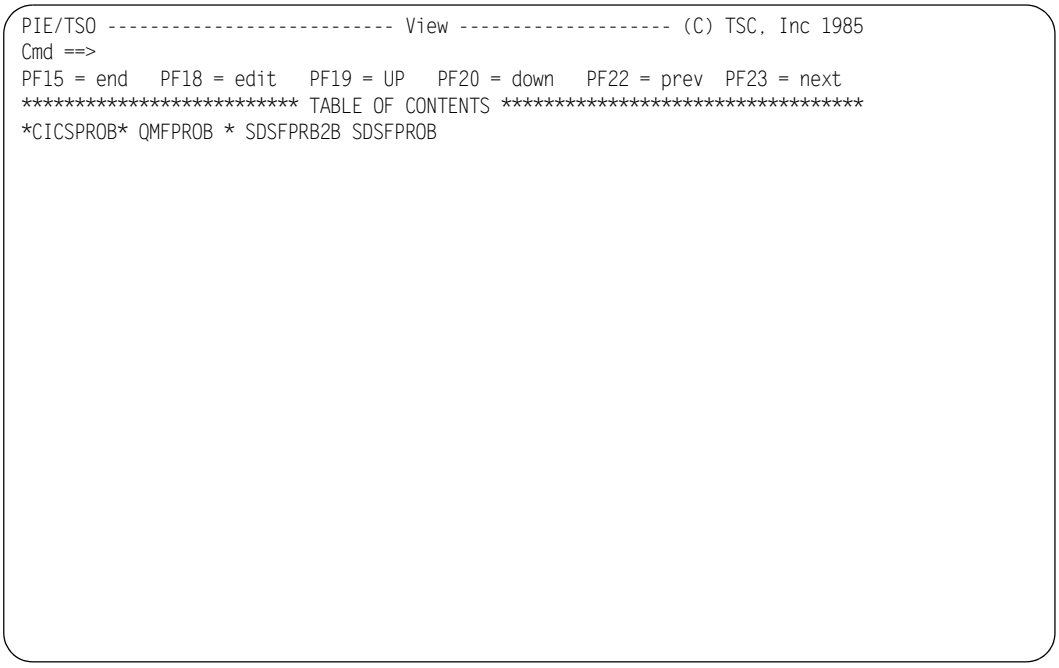
For information about allocating the View dataset in your initial CLIST/EXEC, see “Building Multiple Sessions Automatically,” beginning on page 62.

Starting View and the View Menu

You can open the View session with any session opening command (START, CREATE, etc.) and the V command as a parameter.

1. Open View now. On the MultiTSO command line, enter START V.

   You will get the View menu. The View menu is the first thing you will see whenever you open or switch to View.



   The View menu has a command line, a PF key help line, and a list of screen image members. This list contains the names of all your View dataset members—that is, of all your saved screen images.

   The first time you enter View, the first member will always be highlighted and marked with an asterisk. After that, the highlighting and asterisk will mark the last member you accessed.

   From this list, you can pick a screen image to browse by selecting it with the cursor and pressing PF12. To edit a screen image, you can select it and press the PF6 key for EDIT.

   Some View commands help you move around in this list and select members. They are:

- UP (PF7) which pages up through the list
- DOWN (PF8) which pages down through the list
- PREV (PF10) which moves the cursor to the previous member in the list
- NEXT (PF11) which moves the cursor to the next member in the list

   We'll be using these commands throughout the View tutorial.

2. Now switch back to the MultiTSO menu, using your switch sequence.

Your MultiTSO menu will show the View session at the bottom of its list.

```
PIE MultiTSO Menu ------------------------------------------ (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   LISTC     LISTC                                 I    S  SwKey:   ENTER
+ 1   EDIT      ISPF 2                                I    S  Keys:   OFF
+ 2   MSPF      MSPF                                  I    S  Mkeys:  M ON
+ 3   QMF       DB2 Online Queries                   MI    S  Pfshow: MKEYS
* V   VIEW                                          I    S  Asys:   ON
                                                             Abbrev: ON
                                                             ImpTSO: ON
                                                             AE:     OFF
                                                             JMKey:  OFF
                                                             Screen: SAVE
                                                             Ver : 03.02.01
                                                             Sessions.0-11
                                                             Mon   Oct 12
                                                             Date: 10/12/92
                                                             Time: 14:21:00
                                                             User: JDOE
                                                             Term: TERM4C15
                                                             Sysid:  820A

  F1=HELP      F2=START     F3=END       F4=CANCEL    F5=NEWS     F6=CREATE
  F7=          F8=          F9=         F10=SWKEY    F11=SWCHAR  F12=SWITCH
```

Capturing and Viewing Screen Images

Capturing a screen image is easy. You do it every time you switch from another PIE/TSO session to View. The screen image is placed in the home buffer.

You just used the V command to start View. Now you will use it to capture screen images. When you started View, you combined V with an opening command (START, etc.). To capture screen images, you combine V with the switch sequence.

1. Capture an image now. Switch to a MultiTSO or Network Access session running any application.
2. Place the cursor in the first position of an input field. Issue \V [ENTER]. (Use your own switch sequence.)

   You captured the screen image and switched to the View menu.

```
PIE/TSO -------------------------- View ------------------- (C) TSC, Inc 1985
Cmd ==>
PF15 = end   PF18 = edit   PF19 = UP   PF20 = down   PF22 = prev  PF23 = next
************************* TABLE OF CONTENTS *********************************
*CICSPROB* QMFPROB * SDSFPRB2B SDSFPROB
```

3. Let's look at the screen image you captured. Press PF6 for EDIT.

(Our sample image is from SDSF.)

```
SDSF DA 820A   PAGING  15.66 SIO   28.17 CPU  57.29%     LINE 22-42 (44)
COMMAND INPUT ===> \v                                     SCROLL ===> CSR
NP JOBNAME  STEPNAME PROCSTEP JOBID    OWNER    C POS DP PGN REAL PAGING   SIO
   JDJ3     TSOLJL$  TERM4A2  JPO05198 JDJ3       IN  84   2 2324  0.00   0.00
   KRP1     KRP1     KRP1     JPO05189 ++++++++   N/S 61   5 132   0.00   0.00
   DON3     $TSO01   TERM4C9  JPO05511 DON3       OUT FF   2 220   0.00   0.00
   LOM1     TSOTEST  TERM4BE  JPO06078 LOM1       OUT FF   2 436   0.00   0.00
   YDEBXBB  $PIEI    PIE10008 JPO05672 YDEBXBB    OUT FF   2   0   0.00   0.00
   POE1     $PIEI    PIE10002 JPO06242 POE1       IN  61   2 2216  0.00   0.00
   IEM1     TSOTEST  TERM4B9  JPO05400 IEM1       OUT FF   2 184   0.00   0.00
   TOP1     TSOTEST  PIE10011 JPO05391 TOP1       OUT FF   2 240   0.00   0.00
   LOM1VPOC LOM1VPOC LOM1VPOC JPO05394 ++++++++   OUT FF   5 276   0.00   0.00
   JPODB    JPODB    CICS     JPO05883 ++++++++   IN  51  33 1868  0.00   0.00
   NUH8     $PIEI    PIE11001 JPO06279 NUH8       OUT FF   2 1512  0.00   0.00
   NUH8     $PIEI    PIE10009 JPO05404 NUH8       OUT FF   2 1248  0.00   0.00
   CICSJPO  CICSJPO  CICS     JPO06268 ++++++++   IN  76   3 300   0.00   0.00
   SD080JP  $PIEI    PIE10004 JPO06076 SD080JP    OUT FF   2 1620  0.00   0.00
   CICSJPO  CICSJPO  CICS     JPO06303 ++++++++   IN  78   3 1240  11.92  14.18
   CICSJP3  CICSJP3  CICS     JPO05973 ++++++++   IN  75   3 1196  0.00   0.00
   KUO7     $PIEI    PIE10010 JPO05492 KUO7       OUT FF   2 192   0.00   0.00
   JCL0     TSOTEST  PIE10006 JPO06285 ++++++++   OUT FF   2 904   0.00   0.00
   SAS1     TSOSAS   TERM4AE  JPO06213 SAS1       OUT FF   2 628   0.00   0.00
   CICSJPO  CICSJPO  CICS     JPO05990 ++++++++   IN  76   3 648   0.00   0.00
```

Notice that everything is included in the image—even the \V you used to capture the image.

4. Modify your screen image. All protected fields on the captured screen image are now unprotected, and you can overtype anything.

   In our example, we've taken out the \V and replaced it with “da ostc.”

```
SDSF DA 820A   PAGING  15.66 SIO   28.17 CPU  57.29%     LINE 22-42 (44)
COMMAND INPUT ===> da ostc                                SCROLL ===> CSR
NP JOBNAME  STEPNAME PROCSTEP JOBID    OWNER    C POS DP PGN REAL PAGING   SIO
   JDJ3     TSOLJL$  TERM4A2  JPO05198 JDJ3       IN  84   2 2324  0.00   0.00
   KRP1     KRP1     KRP1     JPO05189 ++++++++   N/S 61   5 132   0.00   0.00
   DON3     $TSO01   TERM4C9  JPO05511 DON3       OUT FF   2 220   0.00   0.00
   LOM1     TSOTEST  TERM4BE  JPO06078 LOM1       OUT FF   2 436   0.00   0.00
   YDEBXBB  $PIEI    PIE10008 JPO05672 YDEBXBB    OUT FF   2   0   0.00   0.00
   POE1     $PIEI    PIE10002 JPO06242 POE1       IN  61   2 2216  0.00   0.00
   IEM1     TSOTEST  TERM4B9  JPO05400 IEM1       OUT FF   2 184   0.00   0.00
   TOP1     TSOTEST  PIE10011 JPO05391 TOP1       OUT FF   2 240   0.00   0.00
   LOM1VPOC LOM1VPOC LOM1VPOC JPO05394 ++++++++   OUT FF   5 276   0.00   0.00
   JPODB    JPODB    CICS     JPO05883 ++++++++   IN  51  33 1868  0.00   0.00
   NUH8     $PIEI    PIE11001 JPO06279 NUH8       OUT FF   2 1512  0.00   0.00
   NUH8     $PIEI    PIE10009 JPO05404 NUH8       OUT FF   2 1248  0.00   0.00
   CICSJPO  CICSJPO  CICS     JPO06268 ++++++++   IN  76   3 300   0.00   0.00
    .
    .
    .
```

Saving Images

Remember that the image we've been working with is in the home buffer only. If we captured another image, this one would be lost. To save the image, first view it with the EDIT command. Then issue the END command.

1. You are already viewing your screen image using the EDIT command. To save the screen image and the changes you have made to it, press PF3 for END.
2. View will ask you for a member name for your new screen image. Enter a member name. It can be up to 8 characters long.

   (If you enter the name of an existing member, you will get a "Member exists . . ." message. View will allow you to re-use a member name, but it will replace the old screen image with the new one. It will ask you to confirm the replacement before going forward. To replace the old member, press ENTER at the prompt.)

   View switches you back to the View menu. You can see the member you just created. (Use PF7 for UP or PF8 for DOWN, if it is not immediately visible.)

Exiting an Image Without Saving It

What if you don't want to save a screen image?

If you aren't viewing the image in the buffer, you don't have to do anything. The image will be lost when you capture another image.

If you are already viewing the image, you can exit without saving by using the END or RETURN commands.

1. First go back to the home buffer to view the screen image you just captured. Press PF6 for EDIT.
2. Now press PF3 for END.
3. You are presented with the member name prompt. Press PF3 again, to exit without saving.
4. Go back into the buffer—press PF6.
5. Press PF4 for RETURN.

   This exits you from the buffer screen image without the prompt. You can use RETURN to exit without saving at any time.

View Modes

EDIT MODE

As you viewed, modified, and saved the screen image you created, you did it all in edit mode. Edit mode, as its name implies, allows you to change an image. It also allows you to view an image you just captured and to save it.

Most commands are unavailable in edit mode. This is because every field in the screen image is unprotected, so View has no way of knowing whether you are issuing a command or editing the image. However you can issue some commands with PF keys. For instance, to issue the HELP command, you can't type HELP, but you can press PF1.

You can access edit mode with PF6 or the EDIT command. You can exit edit mode with PF3 (END) or PF4 (RETURN).

1. You've already practiced editing a newly captured image. Now edit the image you have saved. From the View menu, highlight your member name and press PF6 for EDIT.
2. Change the image.
3. Press PF3 for END.

BROWSE MODE

View's other mode is called browse mode. Browse has the opposite capabilities of edit. So you can't use browse to view a newly captured image or to change a screen image or to save an image. However you can issue View and PIE/TSO commands from browse—commands like SWITCH or SEND or RECEIVE. You can also use browse to view screen images that you don't want to change.

You are in browse mode when you:

- are in the View menu
- select a member from the View menu and press PF12

To get around in browse mode, use the following commands:

- DOWN and UP–to scroll down or up in the member
- NEXT and PREV–to highlight the next or previous member on the View menu
- END and RETURN–to end your browse and return to the View menu

1. You are in the View menu. Highlight a member name and press PF12.

   View will display the screen member.
2. Exit the screen image with PF3 END or PF4 RETURN.

Sending and Receiving Screen Images

Now that you know how to capture and save screen images, you can learn to send and receive them.

To send and receive screen images, you must be in browse mode. You can only send the last screen you captured. To send a screen image, capture it and issue the SEND command.

To receive one, use the RECEIVE command. Try it now.

1. Capture the screen you want to send.
2. Go to the View menu command line, and issue userid [PF5]. Substitute your user ID for `userid` to send the screen image to yourself.

   Your screen image has now been sent. You will receive a message to notify you that someone sent you a screen image.
3. Before you can look at the screen image you've been sent, you have to receive it. Press PF9 for RECEIVE.
4. The next time you receive a screen image, your old one will be lost. If you want to keep it, save it with PF3 END.

Switching and View

In View, there are two types of switching:

- between the current screen image and the View menu
- between other PIE/TSO sessions and View

+ If you are in edit mode, you can't do either kind of switch. Instead exit the screen image with PF3 (END) or PF4 (RETURN) to go to the View menu.

In browse mode, you can switch between the current screen image and the View menu, with the SWITCH command.

Because SWITCH is a View command, you can't use SWITCH to switch between View and other PIE/TSO sessions. Use your switch sequence instead.

Remember that whenever you switch from another PIE/TSO session to View, you capture the screen image you switch from and overlay the screen that was in the home buffer. So if you want to keep the image in the home buffer, be sure to save it before you switch out of View.

Closing View

You may close your View session with END or RETURN. You can also close your View session from the MultiTSO menu with the CANCEL command.

No matter how you end View, you will return to the MultiTSO menu. The screen images in your buffers will remain. If you start another View session, you can issue EDIT or RECEIVE to see them.

1. Close View now. Go to the View menu and press PF3 or PF4.
2. Restart View and switch back to the MultiTSO menu.
3. Enter CAN V.

Help In View

Access HELP from View as you would from any other session: start and switch to the HELP session or just press PF1. To get overall HELP for View, request the VIEW member. VIEW is the default from the View session.

View Commands and PF Keys

View has its own commands. Most of them are assigned to PF keys. As we stated earlier View has two modes: edit and browse. Some commands work in only browse mode. Others work in both.

BOTH MODES

The View commands that work in both modes are:

- HELP (PF1)–displays HELP information
- END (PF3)–ends activity or ends View
- RETURN (PF4)–cancels activity

BROWSE MODE

The View commands that work only in browse mode are:

- SEND (PF5)–sends screen image to another View user
- EDIT (PF6)–displays the member at cursor or in the home buffer
- UP (PF7)–scrolls up in member/menu
- DOWN (PF8)–scrolls down in member/menu
- RECEIVE (PF9)–displays a screen image sent to you
- PREV (PF10)–displays previous member
- NEXT (PF11)–displays next member
- BROWSE (PF12)–browses selected member
- SWITCH (no PF key)–switches between member and menu
- RECALL (no PF key)–displays the last command you entered in View

Printing Screen Images

To print a screen image, use your standard printing procedures and specify your View dataset and the appropriate screen member.

Deleting Screen Images

To delete a screen image, delete its member using your ISPF facilities.

Saving Into a Full View Dataset

If, while you are saving a screen image, you encounter either

```
SAVE WAS UNSUCCESSFUL
```

or

```
SAVE FAILED FOR dataset-name
```

(often accompanied by a system B37/D37/E37 abend), you can resolve the situation without necessarily ending your View session.

First switch out of View and compress the dataset (for example, with ISPF/PDF option 3.1).

If this does not reclaim enough space for your save, you can either expand the dataset or allocate a new one.

1. To allocate a new dataset, first free the one that is already in use. Access a TSO session and enter FREE FI(PIEVIEW).
2. Now, before you switch back to View, allocate your new dataset with:

   ```
   ALLOC FI(PIEVIEW) DA(dataset-name) SHR
   ```
3. Now you may switch back to View and save your screen image.

Chapter 5
Command Primer

In "Chapter 2 Using MultiTSO" you learned the basics of getting in and out of MultiTSO and moving around in it. While we used several MultiTSO commands in the process, we didn't discuss the commands in any depth. This chapter briefs you on the most basic commands. It teaches you how to execute commands automatically at logon in an initial CLIST or REXX EXEC. It also lists all the commands in categories, such as switching, creating sessions, etc.

See the PIE/TSO Command Reference for detailed information on any command.

CONTENTS

Entering Commands

How

You can enter PIE/TSO commands in several ways:

- by typing them on a command line
- by pressing PF keys
- from TSO CLISTs and REXX EXECs

If you enter a command using a PF key and you want to enter any parameters, type the parameters on the command line before you press the PF key. For instance, to execute CANCEL ALL, you could issue ALL [PF4] from the MultiTSO menu command line.

Where

Where you enter PIE/TSO commands depends on the command. Some must be entered from the MultiTSO menu. These commands are:

ALLOC	H	RESHOW
ALLOCH	I	RESTART
CANCEL	KEEP	All SET commands
CLOSE	LOGOFF	SWITCH BACK
DISC	LOGON	SWITCH NEXT
DISPLAY	NAME	V
END	PFSHOW	
ENDBACK	PUSH	
FENCE	PUSHJUMP	
FREE	RECALL	

If you want to execute one of these commands from a session, you can specify the command as a parameter to the PIEEXEC command and execute PIEEXEC as a TSO command. For instance, to execute ALLOC from TSO READY, enter PIEEXEC ALLOC.

You could also execute the command from within a session using the switch sequence. For instance, to execute ALLOC from any session, enter \ALLOC [ENTER]. (Substitute your own switch sequence.)

You don't need PIEEXEC to execute the following commands. You may execute them directly as TSO commands.

ACCESS	MSPF and MULTISPF	START and STARTH
@ALLOC	NOTE	STOP
@CAN	PEXEC	SWCHAR
CREATE and CREATEH	PIECHECK	SWITCH
HIDE	PIESTART	SWKEY
JMKEY	SCREEN	UNHIDE
LOCK	All @SET commands	

The PIE/TSO Command Reference describes each command in detail. Those descriptions will indicate where you can enter the command.

Identifying Sessions

Any time you need to identify a session, you can use either its name or its code. If abbreviations are on, then you can also identify a session with the first characters of its name.

See “Identifying Sessions,” on page 23, for more information.

The Basic Commands

This section briefly outlines the most basic PIE/TSO commands to get you started quickly. For a complete description of any command, see the PIE/TSO Command Reference.

Some commands were reviewed in Chapter 2, “Using MultiTSO." Those commands are omitted in this section.

1. To begin, open a session. Enter START MSPF 2.

DISC

DISC allows you to disconnect your TSO logon. You can use disconnect to move your logon and sessions to another terminal. You can also use it when someone wants to log on at your terminal. You can disconnect, allow the other person to log on, and then reconnect when he or she is finished.

1. To use DISC, enter DISC from the MultiTSO menu command line.
2. To reconnect, log on to TSO using the Reconnect option at the bottom of the logon screen. Your session 0 is just the way you left it.

LOCK

You can use the LOCK command to secure your terminal while you are away from your desk. When your terminal is locked, no one can enter data until your password is entered correctly.

1. To lock your terminal, enter LOCK.

To unlock your terminal, enter your password at the prompt.

CLOSE

Each session takes resources from your region. So if you have several sessions open and you need to run a resource demanding application like APL2, you might not have enough resources remaining.

To regain needed resources you can end a session with the CLOSE command. When you use CLOSE to end a session, MultiTSO leaves the session on the MultiTSO menu. So when you want to go back to the session, you have only to select it from the MultiTSO menu. MultiTSO will re-open the session using the command executed when the session was opened the first time. For example, if you first opened session 1 with START %SAS, MultiTSO will re-open session 1 with START %SAS.

1. To close session 0, enter CLOSE 0. (You may also close a session by placing the cursor next to the session in the column on the left side of the MultiTSO menu, typing C, and pressing ENTER.)

   If you want to close all your sessions, enter CLOSE ALL.

2. Re-open session 0 now. Enter 0 on the MultiTSO menu command line. Notice that MultiTSO automatically runs MSPF 2—the command you entered with the START command above.

FENCE

Some commands, such as APL2, may take the entire region for work space when they initialize. When that happens, there is no storage left to open new sessions. The FENCE command reserves a predetermined amount of storage for such cases.

This amount is set with the FENCE start-up option. The distributed default is 256K. You may change this default with the FENCE command.

1. Change the amount of storage reserved to 1M. Enter FENCE 1M from the MultiTSO menu.

1. Before you start the storage intensive application, enter FENCE ON. MultiTSO will reserve storage space for new sessions.

2. Open a session and start the application.

   When you switch out of the session the reserved storage will be freed. You will have memory available, and you may open a new session.

RECALL

In the MultiTSO menu, you can use RECALL to redisplay the last command you entered. Then you can modify the command and enter it again.

1. To use the recall command, enter R or RECALL on the MultiTSO menu command line.

SCREEN

SCREEN controls two things:

- whether data typed on a screen but not yet entered (with ENTER or a PF key) is saved when you

switch out of a session

- whether the MultiTSO menu header line is displayed

Obviously the first function is what makes SCREEN a basic command. If you have data on your screen that you haven't entered by pressing ENTER or a PF key and you switch to another session, what happens to the data?

If SCREEN is set to SAVE, PIE/TSO effectively “presses ENTER,” and the data is saved.

Sometimes, however, pressing ENTER has negative effects in an application. For instance, it can start other processing or bring you to a menu. In these cases, you must set SCREEN to NOSAVE. With NOSAVE, unentered data is lost.

1. To save data when you switch, enter SCREEN SAVE.
2. To avoid the ENTER when you switch, enter SCREEN NOSAVE.

Setting Function Keys

You can define PF and PA keys as shortcuts to perform any PIE/TSO command. You can activate these keys in the MultiTSO menu or in any MultiTSO session. When you are in a session, the MultiTSO function key overrides the application's use for that function key. You can use MultiTSO function keys to:

- change menu attributes
- perform switching functions
- switch to a specific session

If you like, you can have two sets of function keys: MKEYS and SKEYS. The MKEYS operate only in the MultiTSO menu. The SKEYS can operate in both sessions and the MultiTSO menu.

Using Both Sets of PF Keys

If you want to have one set of keys in the MultiTSO menu and another set of keys in your sessions, you can set up both sets of function keys, and activate the MKEYS on the MultiTSO menu and the SKEYS in your sessions.

1. To do so, first set up your MKEYS. Assign PF key functions with the SET MPFn command. For instance, to set PF2 to switch to your EDIT session, enter SET MPF2 SWITCH EDIT.

   + Session variables can be useful in PF keys. To identify the current session, use an asterisk (e.g., SET MPF2 CLOSE *). To identify the last session you opened, use a question mark (e.g., SET MPF2 SWITCH ?).

2. You may also set the PA3 key for the MultiTSO menu. Use the SET MPA3 command. For example: SET MPA3 RECALL
3. Now activate the MKEYS on the MultiTSO menu. Enter SET MKEYS MKEYS and SET MKEYS ON.
4. Now set your up your SKEYS. Assign PF key functions with the SET PFn command. SET PFn is just like the SET MPFn command. So, to set PF2 to switch back to MultiTSO menu, enter

SET PF2 SWITCH.

5. Set the PA3 key with SET PA3. For example: SET PA3 SWITCH BACK. (SWITCH BACK switches you between the last two sessions you accessed.)
6. Now enter SET KEYS ON ALL to activate the SKEYS in your sessions.

Using the Same Set of PF Keys

If you want to use the same set of keys in both the MultiTSO menu and your sessions, set up only the SKEYS and activate the SKEYS in both your sessions and the MultiTSO menu.

1. To start, set up your SKEYS. Assign PF key functions with the SET PFn command. For instance, to set PF2 to switch between your current session and the MultiTSO menu, enter SET PF2 SWITCH.
2. If you like, set the PA3 key using the SET PA3 command. For example, to set PA3 to SWITCH BACK, enter SET PA3 SWITCH BACK.
3. Activate the SKEYS on the MultiTSO menu. Enter SET MKEYS SKEYS and SET MKEYS ON.
4. Activate the SKEYS in your sessions. Enter SET KEYS ON ALL.

Displaying PF Keys

Use the PFSHOW command to display PF keys on your MultiTSO menu.

1. Enter PFSHOW ON to turn the PF key display on.
2. Enter PFSHOW PF1 to display PF keys 1 through 12. Enter PFSHOW PF13 to display PF keys 13 through 24. Enter PFSHOW PA1 to display your PA keys
3. Enter PFSHOW MKEYS to display the MKEYS. Enter PFSHOW SKEYS to display the SKEYS.

You can change the label on any PF key. Use the SET MPFnL command to change MKEYS PF key labels. Use the SET PFnL command to change SKEYS PF key labels. Use the PA3L and MPA3L commands to change PA3 key labels. For example, to change the MKEYS PF2 label to EDIT, enter SET MPF2L EDIT.

Building Multiple Sessions Automatically

If you execute an initial CLIST or REXX EXEC at logon, you can open sessions and start applications automatically, name your sessions and create comments for them, set up your PF keys, change your switch sequence, and so on. That way, your PIE/TSO environment is ready every time you log on. The MultiTSO menu will display your opened sessions, comments, etc. To use an application, all you have to do is enter the session code on the command line.

```
PIE MultiTSO Menu ---------------------------------------- (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
  1   ISPF      ISPF                                 HKA S  SwKey:  ENTER
  2   EDIT      ISPF 2 (EDIT)                         KA S  Keys:   ON
  3   DSLIST    ISPF 3.4 (DSLIST)                     KA S  Mkeys:  M ON
  4   CICSTEST  Test CICS                             KA S  Pfshow: MKEYS
  5   INFO      Customer Service Database             KA S  Asys:   OFF
  6   NAMENU    Network Access Menu                   KA S  Abbrev: ON
                                                            ImpTSO: ON
                                                            AE:     OFF
                                                            JMKey:  PF21
                                                            Screen: SAVE
                                                            Ver : 03.02.01
                                                            Sessions.0-11
                                                            Mon   Oct 12
                                                            Date: 10/12/92
                                                            Time: 14:01:34
                                                            User: JDOE
                                                            Term: TERM4C15
                                                            Sysid:  820A

  F1=HELP      F2=START     F3=CAN *    F4=ISPF      F5=NEWS     F6=CREATE
  F7=DISC      F8=          F9=        F10=SWKEY    F11=SWCHAR  F12=SWITCH
```

The Basics

You can issue all PIE/TSO commands from CLISTs and REXX EXECs. We will show you a few basic commands in this section. There are many more commands available. To get a feel for the possibilities, see " Categorizing the Commands" on page 67. See the PIE/TSO Command Reference for specific command requirements.

You'll be better able to create initial CLISTs and EXECs if you understand how the commands are executed. Your logon procedure starts a session and executes the CLIST/EXEC in that session. Your commands execute at TSO READY; so you must code every command in TSO format. Do not code abbreviations—code the full command name. Remember that some commands, such as ALLOC, must be preceded with PIEEXEC to execute at TSO READY. Other commands, such as CREATE, execute at TSO READY by default. For example, to create one session and allocate another, you would code:

```
CREATE
PIEEXEC ALLOC
```

(Again, refer to the PIE/TSO Command Reference for specific command requirements.)

Be sure to name your CLIST/EXEC according to your installation's standards, or it will not execute

when you log on. See your PIE/TSO administrator for assistance.

Hints For Creating an Effective CLIST/EXEC

1. Your main objective in an initial CLIST/EXEC is probably to open sessions automatically. There are several session opening commands, but ALLOC is the best one for CLISTs/EXECs. That is because ALLOC opens sessions without switching to them. (Once you switch out of the session executing the CLIST/EXEC, it stops executing until you return to it.) Also, sessions opened with ALLOC remain on the MultiTSO menu even after you end them. So your MultiTSO menu doesn't change when you end a session.
2. When you open a session, you can start a TSO command in it, too. In the menu on the previous page, we started an application in every session we opened, except session 0, which is native TSO. We started ISPF in session 1 (PIEEXEC ALLOC 1 MSPF), ISPF 2 in session 2 (PIEEXEC ALLOC 2 MSPF 2), and so forth. The TSO command will execute when you switch to the session the first time.
3. Never execute ISPF in a CLIST or EXEC. Always execute MSPF. This ensures that MultiTSO can execute ISPF successfully no matter which session you switch to first.
4. To create the session menu effect, you can give each session a name and a comment. Use the NAME and NOTE commands to do this. For example, we used the following commands to name and define session 2:

   ```
   PIEEXEC NAME 2 EDIT
   PIEEXEC NOTE 2 ISPF 2 (EDIT)
   ```

5. To change your switch key, switch character, and jump key, use the SWKEY, SWCHAR, and JMKEY commands, respectively. For example:

   ```
   SWKEY ENTER
   SWCHAR \
   JMKEY PF21
   ```

6. You may want to change your PF keys as well. See " Setting Function Keys" on page 59, for these commands. Example:

   ```
   @SET KEYS ON ALL
   @SET MKEYS ON
   @SET MPF4 SWITCH ISPF
   @SET PF4   SWITCH ISPF
   @SET MPF3 CAN ALL
   @SET MPF7 DISC
   ```

7. When you are finished opening sessions and defining the environment, you may want to switch back to the MultiTSO menu. That way you are presented with the MultiTSO menu when you log on, instead of the session in which the CLIST/EXEC executed. Use the SWITCH command to switch to the MultiTSO menu: SWITCH. Or cancel the session your CLIST/EXEC is running in with PIEEXEC CANCEL *.
8. At times you will want to interrupt your CLIST/EXEC and execute something in another session. To do so, use any switching command (such as SWITCH or START) to switch to the other session. When you switch back to the session executing the CLIST/EXEC, it will resume

processing.

9. You can set up several ISPF sessions with different options. Pass parameters for these options just as you would pass them to ISPSTART. For example, to allocate a session that will present panel ZSDSFOP2, code

```
ALLOC MSPF PANEL(ZSDSFOP2) NEWAPPL(ISF)
```

   To allocate a session that will present the RMFWDM command, code

```
ALLOC MSPF CMD(RMFWDM)
```

   To allocate a session that will present the program ICEPMD01, code

```
ALLOC MSPF PGM(ICEPMD01) NEWAPPL(ICE) NOCHECK
```

10. You may want to simplify access to tools you don't use very often. In this case, you want pre-allocation to be automated, but you don't want to waste time and resources on these tools every time you log on. To accomplish this, create a separate CLIST/EXEC for the tool. Then when you allocate the session for it, execute that CLIST or EXEC. The CLIST/EXEC will only execute if you switch to the session.

    For example, to set up INFOSYS with two databases, create a CLIST named INFO containing the following statements.

```
ALLOC DDN(VSAM) DSN('INFO.SUPPORT') SHR
ALLOC DDN(VSAM2) DSN('INFO.VSAM') SHR
PEXEC PGM BLGOZD   '&PARM.'
FREE DDN(VSAM VSAM2)
```

    This will allocate your databases, start INFOSYS, and, when you exit the session, free the datasets. Then in your initial CLIST, allocate the INFOSYS session as follows:

```
PIEEXEC ALLOC  %INFO
PIEEXEC NAME ? INFO
PIEEXEC NOTE ? Customer Service Database
```

    (The ? is a variable for the last session opened.)

11. If you want to fence storage from an application automatically, you must execute the application from a separate CLIST/EXEC and code FENCE ON in that CLIST/EXEC before you start the application. (If you code FENCE ON in your initial CLIST/EXEC, you can't ensure that it will apply to the correct application. It will apply only to the application running in the first session you switch to.) For example, you could start APL2 with the following CLIST:

```
PIEEXEC FENCE ON
APL2
```

12. To automate View, pre-allocate your View dataset, then allocate your View session, as follows:

    ALLOC FI(PIEVIEW) DA(dataset-name) SHR
    PIEEXEC V

13. PIESTART is a special command that is useful in CLISTs/EXECs or can be used in place of them.

    When you execute PIESTART, you indicate the session by name and you may include a

command to execute in the session. If the session does not yet exist, PIESTART will execute START and NAME, to open and switch to the next available session and to give it the name you indicated. If you include a command, PIESTART will execute that command in the session.

If the session already exists, PIESTART will switch you to that session. It will ignore any command you include to execute in the session.

PIESTART format is PIESTART //name/command.

14. PIECHECK determines whether PIE/TSO is active. It passes a return code to notify you of PIE/TSO status. The return codes are listed in the PIE/TSO Command Reference. With PIECHECK your initial CLIST/EXEC can perform special processing if MultiTSO is not active.
15. You can place comments in your CLIST or REXX EXEC. Precede your comment with /*. End it with */.

SAMPLE CLIST

The following is a sample initial CLIST. It activates the jump key and sets it to PF21. It sets PF keys for the MultiTSO menu and for sessions and turns them on. SET AE ON ALL prevents you from seeing TSO READY when you end an application.

The CLIST allocates sessions 1 through 6. In this case, your logon procedure starts session 0. The CLIST also names these sessions and creates menu comments for them. When all is finished, it cancels the current session and switches you back to the MultiTSO menu.

```
JMKEY PF21
@SET MKEYS ON
@SET MPF3  CAN *
@SET MPF7  DISC
@SET KEYS  ON ALL
@SET PF4   SWITCH ISPF
@SET PF11  LOCK
@SET AE ON ALL
PIEEXEC ALLOCH 1 MSPF 2
PIEEXEC NAME   1 EDIT
PIEEXEC NOTE   1 ISPF 2 (EDIT)
PIEEXEC ALLOC  2 MSPF CMD(RMFWDM)
PIEEXEC NAME   2 RMF3
PIEEXEC NOTE   2 RMF Monitor III
PIEEXEC ALLOC  3 MSPF 3.4
PIEEXEC NAME   3 DSLIST
PIEEXEC NOTE   3 ISPF 3.4 (DSLIST)
PIEEXEC ALLOC  4 ACCESS CICSTEST &&ZSNON
PIEEXEC NAME   4 CICSTEST
PIEEXEC NOTE   4 Test CICS
PIEEXEC ALLOC  5 %INFOMVS
PIEEXEC NAME   5 INFO
PIEEXEC NOTE   5 Customer Service Database
PIEEXEC ALLOC  6 ACCESS
PIEEXEC NAME   6 NAMENU
```

```
PIEEXEC NOTE   6 Network Access Menu
PIEEXEC CANCEL *
```

SAMPLE REXX EXEC

The following sample is found in member PIEINIT3 of the PIE/TSO CLIST dataset.

```
/* REXX */
"PIEEXEC"
/*************************************************************/
/* Change the MultiTSO Title                                 */
/*************************************************************/
"PIEEXEC SET TITLE TSO Application Menu"
/*************************************************************/
/* Control MultiTSO Menu display                             */
/*************************************************************/
"PIEEXEC PFSHOW OFF"
"PIEEXEC SET NOTEPAD OFF"
/*************************************************************/
/* Assign MultiTSO PF Keys                                   */
/*************************************************************/
"PIEEXEC SET PF2 SPF"
"PIEEXEC SET PF3 DIDOCS"
"PIEEXEC SET PF4 OM"
"PIEEXEC SET PF5 TSCDB"
"PIEEXEC SET PF10 ENDBACK"
"PIEEXEC SET PF11 SET ASYS OPPOSITE"
"PIEEXEC SET PF12 SET KEYS OPPOSITE"
"PIEEXEC SET PF13 SWITCH BACK"
/*************************************************************/
/* Make the MultiTSO Menu Keys the same as the session keys */
/*************************************************************/
"PIEEXEC SET MKEYS SKEYS
/*************************************************************/
/* Assign Specific labels to some PF keys                    */
/*************************************************************/
"PIEEXEC SET PF11L ASYS OPP"
"PIEEXEC SET PF12L KEYS OPP"
"PIEEXEC SET PF13L BACK"
/*************************************************************/
/* Assign Switch Key and Character                          */
/*************************************************************/
"PIEEXEC SWKEY ENTER"
"PIEEXEC SWCHAR {"
/*************************************************************/
/* Turn Jump Key OFF                                         */
/*************************************************************/
"PIEEXEC JMKEY OFF"
/*************************************************************/
/* Allocate and assign names to sessions                     */
/*************************************************************/
"PIEEXEC ALLOCH DIDOCS"
```

```
"PIEEXEC NAME 1 DIDOCS"
"PIEEXEC ALLOCH %MYSPF PGM(OMNI#ISP)"
"PIEEXEC NAME 2 OMNI"
"PIEEXEC ALLOC %MYSPF"
"PIEEXEC NAME 3 SPF"
"PIEEXEC ALLOCH %MYIPCS"
"PIEEXEC NAME 4 IPCS"
"PIEEXEC ALLOCH %MYSPF RMF3"
"PIEEXEC NAME 5 RMF3"
"PIEEXEC ALLOC ACCESS TSCDB &ZSNON"
"PIEEXEC NAME 6 TSCDB"
"PIEEXEC ALLOCH RMFMON"
"PIEEXEC NAME 7 RMF"
/*************************************************************/
/* Assign notes                                              */
/*************************************************************/
"PIEEXEC NOTE tscdb Electronic Mail System"
"PIEEXEC NOTE spf ISPF/Catalyst"
"PIEEXEC NOTE didocs MVS System Console"
"PIEEXEC NOTE omni OmniJES Controller"
"PIEEXEC NOTE rmf3 RMF Monitor III"
"PIEEXEC NOTE rmf RMFMON"
"PIEEXEC NOTE ipcs Dump and Trace Analysis"
/*************************************************************/
/* Control Asynchronous Switching                            */
/*************************************************************/
"PIEEXEC SET ASYS ON ALL"
/*************************************************************/
/* Control Autoend                                           */
/*************************************************************/
"PIEEXEC SET AE ON ALL"
```

Categorizing the Commands

Often there are two or more PIE/TSO commands that perform the same general task but in different ways. For instance, the CREATE and ALLOC commands both open new sessions without switching to them. However when you end a session opened with CREATE, the session completely disappears. When you end a session opened with ALLOC, the session remains on the MultiTSO menu.

Use this section to get an idea of what commands you can use to perform certain tasks.

Opening Sessions

START Opens a session and switches to it.

STARTH Opens a hidden session and switches to it.

ALLOC Opens a session with the allocated attribute. When you end the session, it will remain on the MultiTSO menu.

ALLOCH Opens a hidden session with the allocated attribute. When you end the session it will remain on the MultiTSO menu.

CREATE Opens a session.

CREATEH Opens a hidden session.

SET CMDDEFLT Sets a default TSO command to execute when you open a session.

SET IMPTSO Determines how MultiTSO will handle non-PIE/TSO commands entered from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PIESTART Starts and names the session specified. If the session already exists, PIESTART switches to it.

Ending Sessions and Logging Off

LOGOFF Logs you off a session. If no other sessions are active, it also logs you off TSO.

CANCEL Abends the application in a session and ends the session.

CLOSE Cancels a session but leaves the session allocated on the MultiTSO menu.

DISC Disconnects your TSO logon.

KEEP Assigns the allocated attribute to a session. When you end the session, it will remain on the MultiTSO menu.

FREE Removes the allocated attribute. When you end the session, it will be removed from the MultiTSO menu.

SET AE Sets the auto-end attribute. When the application running in an auto-end session terminates, the session also terminates.

SET CMDFINAL Sets a default TSO command to execute when you log off all sessions.

RESTART Cancels a session and reopens it automatically.

LOGON Cancels all your sessions and logs you back on to TSO.

STOP Ends a session from a CLIST or REXX EXEC.

Switching

SWITCH Switches to the MultiTSO menu or to the specified session.

SWITCH BACK Switches between the last two sessions accessed.

SWITCH NEXT Switches forward through sessions in numerical order.

SWCHAR Sets the switch character.

SWKEY Sets the switch key.

JMKEY Sets the jump key.

HIDE Hides a session, removing it from the jump key cycle.

UNHIDE Unhides a session, returning it to the jump key cycle.

END Pushes the current session to the bottom of the stack and switches to the last session accessed.

PUSH Alias for END.

ENDBACK Pushes the current session to the bottom of the stack and switches to the session you accessed before the current session.

PUSHJUMP Alias for ENDBACK.

SET ASYS Enables asynchronous switching.

Identifying Sessions

NAME Assigns a name to a session.

SET DFLTNAME Determines whether PIE/TSO will assign names to sessions automatically.

NOTE Modifies a session's comment.

SET PARSER Changes the way you identify sessions.

SET ABBREV Determines whether you can use abbreviations for session names.

Controlling Function Keys

PFSHOW Controls the PF key display on the MultiTSO menu.

SET MKEYS Activates MultiTSO menu PF keys.

SET MPFn Sets MultiTSO menu PF keys.

SET MPFnL Assigns labels to MultiTSO menu PF keys.

SET MPA3 Sets the PA3 key on the MultiTSO menu.

SET MPA3L Assigns a label to the MultiTSO menu PA3 key.

SET KEYS Activates session PF keys.

SET PFn Sets session PF keys.

SET PFnL Assigns labels to session PF keys.

SET PA3 Sets the PA3 key for use in a session.

SET PA3L Assigns a label to the session PA3 key.

Executing Commands

PIEEXEC Prefix for PIE/TSO commands so that you can enter them from TSO READY and from applications that accept TSO commands.

RECALL Displays the last command entered from the MultiTSO menu.

Your MultiTSO Environment

SCREEN Saves data typed on a screen but not yet entered when you switch out of a session. Also displays the MultiTSO menu header line.

LOCK Prevents unauthorized use of your terminal while you are not using it.

SET LOCKINPUT Determines whether data typed on your screen but not entered (with ENTER or a PF key) before your terminal is locked will be passed back to the screen after your terminal is unlocked.

SET LOCKTIME Determines the amount of time your terminal must be inactive before it is automatically locked.

Controlling the MultiTSO Menu

SET INFOWINDOW Controls the MultiTSO menu information window.

SET SESATT Controls the session attribute display.

SET TITLE Controls the title line.

SET HEADER Controls the header line.

RESHOW Refreshes the MultiTSO menu.

SET MENU Controls the MultiTSO menu display.

Miscellaneous

MSPF and MULTISPF Allow you to run copies of ISPF in multiple sessions at once.

FENCE Reserves storage space from applications that take all available storage in your region, so that you can open more sessions later.

PEXEC Signals PGM to apply the PGN and SWAP/NOSWAP values assigned to the application.

PIECHECK Checks whether PIE/TSO is running.

DISPLAY Displays information about your environment.

MultiTask

SET MTASK Turns MultiTask on and off.

Network Access

ACCESS Invokes Network Access.

SET AATTN Sets the attention key action in Network Access sessions.

SET APFn Sets the attention key function to a PF key in Network Access sessions.

SET PASSWORD Sets your password for ACCESS variables.

Special Sessions

H Opens the MultiTSO HELP session.

I Opens the INFO session.

V Opens View and captures screen images when it is executed from a session with the switch sequence.

Glossary

*****: Indicates the current session, both on the MultiTSO menu display and in PIE/TSO commands.

Allocated session: Session created with ALLOC, @ALLOC, or ALLOCH or assigned the allocated attribute with the KEEP command. When you end an allocated session, it will remain on the MultiTSO menu. To reopen it, simply type the session code on the MultiTSO menu command line.

Alternate parameters: Parameters which are mutually exclusive.

Application table: The table that defines VTAM applications to Network Access.

Asynchronous switching: Switching out of a session while it is running a task. Asynchronous switching must be turned on in a session before you have the ability to switch out while that session is processing.

Attributes: Session characteristics, such as active session, MultiTask active, hidden session, etc.

Auto-end (AE): A session attribute. When auto-end is on for a session, the session closes when it reaches TSO READY. If you are opening an auto-end session, you must include a TSO command in the session opening command, to avoid having the session end immediately after you open it.

Background session: A session with MultiTask on. The background session continues to process a task off-screen while you work in the foreground session.

Browse mode: A View mode in which you can view a saved screen image and perform commands.

Character: Alpha-numeric character—A to Z, 0 to 9.

Closing a session: Terminating a session with the CLOSE command. This leaves the session on the MultiTSO menu.

Current screen image: The saved screen image you last displayed.

Current session: The session you are in or, if you are in the MultiTSO menu, the last session you were in.

Edit mode: A View mode in which you can view and edit screen images.

Ending a session: Terminating a session.

ENTER: The ENTER key.

Enter: Type the boldfaced characters and press ENTER.

Foreground session: The session you work in and see on your screen while MultiTask processes another task in the background session.

H session: The HELP session.

Hidden session: A session hidden from the jump key with the HIDE command.

Home buffer: A View buffer that can contain a captured screen image.

I session: The INFO session.

INFO session: A special session displaying IBM green card information and general information about PIE/TSO.

Information window: The portion of the MultiTSO menu displaying information about your PIE/TSO environment, such as the switch key, version, etc.

Installation job: The job that unloads the PIE/TSO product tape, assembles and links load modules, etc.

Inter-user buffer: A View buffer that contains a screen image to be sent to another user or a screen image received from another user.

Issue: Type the boldfaced characters and press the key in brackets (e.g., [ENTER] or [PF2]).

Jump key: The PF key you can use to jump from session to session. The jump key skips the MultiTSO menu and hidden sessions.

MSPF and MULTISPF: MultiTSO commands which allow you to run ISPF in multiple sessions at the same time.

MultiTask: A PIE/TSO component that processes tasks in a background session while allowing you to use other sessions in the foreground.

MultiTask background session: See “Background session.”

MultiTask foreground session: See “Foreground session.”

MultiTSO menu PF keys: See “MKEYS.”

MKEYS: The set of PF and PA keys defined for use in the MultiTSO menu. You may only activate these keys in the MultiTSO menu. You may not activate them in sessions.

Native ISPF datasets: ISPF/PDF datasets which must be preallocated and which cannot be shared between sessions.

Network Access session: A session running the ACCESS command.

Opening a session: Bringing a session into being with START, CREATE, ALLOC, etc.

Parameter dataset: The member containing Network Access start-up options.

Screen: A field on the MultiTSO menu and a command. The SCREEN command determines whether or not to save data that has been typed on your screen but not yet entered

(by pressing ENTER or a PF key) when you switch out of a session. The field on the MultiTSO menu displays how this data will be handled.

Session ID: The number, one character code, or name of a session.

Session name: The name assigned manually to a session with the NAME command or assigned automatically when SET DFLTNAME is ON. It is displayed on the MultiTSO menu in the Name column.

Session number: The number or, sometimes, one character code that refers to a session. It can be 0 through 11, A, B, V, H, I, or X (for the MultiTSO menu). It is displayed on the MultiTSO menu in the # column.

Screen image: A copy of a screen that you captured using View.

Session attributes: Session characteristics, such as active session, MultiTask active, hidden session, etc.

Session information area: The portion of the MultiTSO menu displaying information about your sessions.

SESSIONS member: The member containing PIE/TSO start-up options.

Sessions PF keys: See “SKEYS.”

SKEYS: The set of PF and PA keys defined for use in PIE/TSO sessions. You may activate these keys in particular sessions or all sessions. You may also activate them in the MultiTSO menu.

Special session: A PIE/TSO session that does not count toward your maximum number of sessions.

Stack: The order in which you access MultiTSO sessions. You can change this order with the END and ENDBACK commands.

Start-up options: Customize the PIE/TSO environment. They are recorded in the SESSIONS member of the PARMLIB dataset.

Switch character: The character you use as part of your switch sequence. Our examples assume it is a backslash (\). It can be any non-alphanumeric character on the 3270 keyboard.

Switch key: The key you use to execute the switch sequence. Our examples assume it is ENTER.

Switch sequence: A method of switching between sessions and the MultiTSO menu. You can also use it to execute MultiTSO commands.

Type: Type the boldfaced characters without pressing ENTER.

User ID table: The table that defines users, terminals, and virtual terminal pools for Network Access applications.

View: A feature of PIE/TSO that allows you to make copies of screens, edit them, save them, and send them to other users.

Index

Symbols

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

PIE/TSO
Command Reference

Release 3.2.1

UNICOM
SYSTEMS, INC.

PTSOCR321-01

Printed: October 2002

This manual applies to PIE/TSO release 3.2.1 and to all subsequent releases of the product until otherwise indicated by new editions or updates to this publication.

UNICOM Systems, Inc.
Unicom Plaza - Suite 310
15535 San Fernando Mission Blvd.
Mission Hills, California 91345

Contents

Preface

Using this Guide

What is this manual about?

The PIE/TSO Command Reference is a reference guide to the PIE/TSO commands that are available to all users. The PIE/TSO commands available only with special authorization (TRACE CARO, TRACE ACCESS, and LOCATE) are described in the PIE/TSO Administrator Guide.

Chapter 1 briefly instructs you in the use of PIE/TSO commands.

Chapter 2 describes general PIE/TSO commands, in alphabetical order.

Chapter 3 describes View commands, in alphabetical order.

Who is this manual for?

This manual is written for all PIE/TSO users. We assume that you know how to use TSO and ISPF. We also assume that you are familiar with PIE/TSO. For a guide to PIE/TSO features, see the PIE/TSO User Guide.

Chapter 1
Command Conventions

This chapter reviews general command syntax. It also lists all the commands in categories, such as switching, creating sessions, etc.

Chapter 2, “General Commands,” presents each command in detail. Commands are listed in alphabetical order. The categorical list at the end of this chapter can be helpful as a guide when you turn to the alphabetical listing.

CONTENTS

Command Syntax

A PIE/TSO command consists of a command name, followed by parameters.

```
COMMAND parameters
```

You may enter PIE/TSO commands in upper or lower case or a mixture of both.

Separate parameters from the command name and from each other with spaces.

Always type parameters in the order they are given in the format box. You may omit parameters, but don't change their order.

Typographical Conventions

Format boxes, like the one above, present the format of each command visually. The following section describes the conventions used in those boxes.

UPPER AND LOWER-CASE

You can abbreviate almost all command names and some keywords. The abbreviation is shown in capital letters. Other letters are lower case, and you may omit them when you type in the command.

For instance:

```
CANcel
```

indicates that CAN is the abbreviation of the CANCEL command. You can enter CAN or CANCEL on the command line. Look at the SWKEY command:

```
SwKey
```

SK is the abbreviation.

☞ Abbreviations are only valid from the MultiTSO menu command line, with PIEEXEC (a PIE/TSO command prefix), and with the switch sequence.

UPPER-CASE AND LOWER-CASE ITALICS IN PARAMETERS

Some parameters must be entered as shown in the format box. These are keyword parameters. However most parameters are not keywords. They are simply names for data that you must supply.

If any part of a parameter is in upper-case, that parameter is a keyword. Code either the abbreviation or the entire parameter. Do not substitute a value for it. For instance:

```
CANcel ALL
```

indicates that the ALL parameter is a keyword. You must code ALL.

If the parameter is in lower-case italic, don't code the parameter itself, but substitute a valid value in its place. For instance:

```
CANcel id
```

indicates that you must substitute a session ID for the id parameter. So you would code CANCEL 7 to cancel session 7.

UNDERLINING

If a parameter is underlined, it is the default parameter for the command.

For instance:

```
Display [SYSTEM|id|STORAGE]
```

indicates that SYSTEM is the default. If you want DISPLAY SYSTEM, you may enter only DISPLAY.

BRACKETS AND BRACES AND NEITHER

Some parameters are required—you must supply them for the command to work. Some are optional—you can enter them or omit them as you like.

If a parameter is in brackets, it is optional.

```
Start [id] [command]
```

Both the id and command parameters are optional.

If a parameter is not in brackets, it is required.

```
SwKey n
```

In addition, if a parameter is in braces, it is required.

```
SwKey {n}
```

STACKING PARAMETERS AND STRAIGHT LINES

Sometimes several parameters are mutually exclusive. That is, you can specify only one of those parameters when you enter the command. For instance, with the CANCEL command, you can enter CANCEL 2 or CANCEL ALL but never CANCEL 2 ALL.

In format boxes, mutually exclusive parameters are stacked on top of each other in one set of brackets or braces.

If the parameters are required, they are shown in braces. You must supply one parameter from the set.

```
SET ASYS {ON       }
         {OFF      }
         {OPPOSITE }
```

If the parameters are optional, they are shown in brackets. You can supply one if you like.

```
SET ASYS [ids]
         [ALL]
```

At times, mutually exclusive parameters are separated by a straight line (|).

```
SET ASYS {ON|OFF|ALL} [ids|ALL]
```

Parameter Tables

Command parameters are shown in tables for quick reference. Parameter tables indicate what values are valid for the parameter and the parameter's default.

Parameter	Valid	Default
id	0 to 11 or name	None
command	Any TSO command	None

Examples

Examples are given for most commands and procedures. Examples are shown in heavy type: NOTE 1 EDIT

Entering Commands

How

You can enter PIE/TSO commands in several ways:

- by typing them on a command line
- by pressing PF keys
- from TSO CLISTs and REXX EXECs

If you enter a command using a PF key and you want to enter any parameters, type the parameters on the command line before you press the PF key. For instance, to execute CANCEL ALL, you could issue ALL [PF4] from the MultiTSO menu command line.

Where

Where you enter PIE/TSO commands depends on the command. Some must be entered from the MultiTSO menu. These commands are:

ALLOC	FREE	PUSHJUMP
ALLOCH	H	RECALL
CANCEL	I	RESHOW
CLOSE	KEEP	RESTART
DISC	LOGOFF	All SET commands
DISPLAY	LOGON	SWITCH BACK
END	NAME	SWITCH NEXT
ENDBACK	PFSHOW	V
FENCE	PUSH	

If you want to execute one of these commands from within a session, you can specify the command as a parameter to the PIEEXEC command, and execute PIEEXEC as a TSO command. For instance, to execute the ALLOC command from TSO READY, enter PIEEXEC ALLOC.

You could also execute the command from within a session using the switch sequence. For instance, to execute ALLOC from any session, enter \ALLOC [ENTER]. (Substitute your own switch sequence.)

You don't need PIEEXEC to execute the following commands. You may execute them directly as TSO commands.

ACCESS	MSPF and MULTISPF	START and STARTH
@ALLOC	NOTE	STOP
@CAN	PEXEC	SWCHAR
CREATE and CREATEH	PIECHECK	SWITCH
HIDE	PIESTART	SWKEY
JMKEY	SCREEN	UNHIDE
LOCK	All @SET commands	

Chapter 2 describes each command in detail. These descriptions will indicate where you

can enter the command.

Identifying Sessions

Sessions have both codes and names.

Session codes are one alpha-numeric character (except for the codes 10 and 11, which are two). They are shown in the # column of the MultiTSO menu. Session numbers are 0 through 11. The alphabetic codes are:

- A–Session 10
- B–Session 11
- H–Help session
- I–INFO session
- V–View session
- X–MultiTSO menu

Anytime you need to identify a session, you can use either its name or its code. (Throughout the PIE/TSO manuals, the term ID refers to both session names and codes.) If abbreviations are on, then you can also identify a session with the first characters of its name.

When you request a session, MultiTSO doesn't know if you are specifying a command, a session code, a session name, or an abbreviated name. It goes through a matching sequence every time you send it a string.

- First it tries to match your string with a command. (For example, if you type HIDE, MultiTSO will execute the HIDE command.)
- Next it tries to match a session code.
- Next it tries to find a session with the exact name you typed.
- Last it tries to find a session that begins with the characters you typed.

So when you name a session or abbreviate a session name, you have to make sure that it will be the first match in this sequence. For example, don't abbreviate a session name to A, B, H, I, V, or X. You'll get the session the code refers to. Whenever in doubt, use the whole session name or its code!

In addition to session names and codes, there are two variables for identifying sessions. The * (asterisk) identifies the current session. The ? (question mark) identifies the last session you opened.

Categorizing the Commands

Often there are two or more PIE/TSO commands that perform the same general task but in different ways. For instance, the CREATE and ALLOC commands both open new sessions without switching to them. However when you end a session opened with CREATE, the session completely disappears. When you end a session opened with ALLOC, the session remains on the MultiTSO menu.

Use this section to get an idea of what commands you can use to perform certain tasks.

Opening Sessions

START Opens a session and switches to it.

STARTH Opens a hidden session and switches to it.

ALLOC Opens a session with the allocated attribute. When you end the session, it will remain on the MultiTSO menu.

ALLOCH Opens a hidden session with the allocated attribute. When you end the session it will remain on the MultiTSO menu.

CREATE Opens a session.

CREATEH Opens a hidden session.

SET CMDDEFLT Sets a default TSO command to execute when you open a session.

SET IMPTSO Determines how MultiTSO will handle non-PIE/TSO commands entered from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PIESTART Starts and names the session specified. If the session already exists, PIESTART switches to it.

Ending Sessions and Logging Off

LOGOFF Logs you off a session. If no other sessions are active, it also logs you off TSO.

CANCEL Abends the application in a session and ends the session.

CLOSE Cancels a session but leaves the session allocated on the MultiTSO menu.

DISC Disconnects your TSO logon.

KEEP Assigns the allocated attribute to a session. When you end the session, it will remain on the MultiTSO menu.

FREE Removes the allocated attribute. When you end the session, it will be removed from the MultiTSO menu.

SET AE Sets the auto-end attribute. When the application running in an auto-end session terminates, the session also terminates.

SET CMDFINAL Sets a default TSO command to execute when you log off all sessions.

RESTART Cancels a session and reopens it automatically.

LOGON Cancels all your sessions and logs you back on to TSO.

STOP Ends a session from a CLIST or REXX EXEC.

Switching

SWITCH Switches to the MultiTSO menu or to the specified session.

SWITCH BACK Switches between the last two sessions accessed.

SWITCH NEXT Switches forward through sessions in numerical order.

SWCHAR Sets the switch character.

SWKEY Sets the switch key.

JMKEY Sets the jump key.

HIDE Hides a session, removing it from the jump key cycle.

UNHIDE Unhides a session, returning it to the jump key cycle.

END Pushes the current session to the bottom of the stack and switches to the last session accessed.

PUSH Alias for END.

ENDBACK Pushes the current session to the bottom of the stack and switches to the session you accessed before the current session.

PUSHJUMP Alias for ENDBACK.

SET ASYS Enables asynchronous switching.

Identifying Sessions

NAME Assigns a name to a session.

SET DFLTNAME Determines whether PIE/TSO will assign names to sessions automatically.

NOTE Modifies a session's comment.

SET PARSER Changes the way you identify sessions.

SET ABBREV Determines whether you can use abbreviations for session names.

Controlling Function Keys

PFSHOW Controls the PF key display on the MultiTSO menu.

SET MKEYS Activates MultiTSO menu PF keys.

SET MPFn Sets MultiTSO menu PF keys.

SET MPFnL Assigns labels to MultiTSO menu PF keys.

SET MPA3 Sets the PA3 key on the MultiTSO menu.

SET MPA3L Assigns a label to the MultiTSO menu PA3 key.

SET KEYS Activates session PF keys.

SET PFn Sets session PF keys.

SET PFnL Assigns labels to session PF keys.

SET PA3 Sets the PA3 key for use in a session.

SET PA3L Assigns a label to the session PA3 key.

Executing Commands

PIEEXEC Prefix for PIE/TSO commands so that you can enter them from TSO READY and from applications that accept TSO commands.

RECALL Displays the last command entered from the MultiTSO menu.

Your MultiTSO Environment

SCREEN Saves data typed on a screen but not yet entered when you switch out of a session. Also displays the MultiTSO menu header line.

LOCK Prevents unauthorized use of your terminal while you are not using it.

SET LOCKINPUT Determines whether data typed on your screen but not entered (with ENTER or a PF key) before your terminal is locked will be passed back to the screen after your terminal is unlocked.

SET LOCKTIME Determines the amount of time your terminal must be inactive before it is automatically locked.

Controlling the MultiTSO Menu

SET INFOWINDOW Controls the MultiTSO menu information window.

SET SESATT Controls the session attribute display.

SET TITLE Controls the title line.

SET HEADER Controls the header line.

RESHOW Refreshes the MultiTSO menu.

SET MENU Controls the MultiTSO menu display.

Miscellaneous

MSPF and MULTISPF Allow you to run copies of ISPF in multiple sessions at once.

FENCE Reserves storage space from applications that take all available storage in your region so that you can open more sessions later.

PEXEC Signals PGM to apply the PGN and SWAP/NOSWAP values assigned to the application.

PIECHECK Checks whether PIE/TSO is running.

DISPLAY Displays information about your environment.

MultiTask

SET MTASK Turns MultiTask on and off.

Network Access

ACCESS Invokes Network Access.

SET AATTN Sets the attention key action in Network Access sessions.

SET APFn Sets the attention key function to a PF key in Network Access sessions.

SET PASSWORD Sets your password for ACCESS variables.

Special Sessions

H Opens the MultiTSO HELP session.

I Opens the INFO session.

V Opens View and captures screen images when it is executed from a session with the switch sequence.

Chapter 2
General Commands

This chapter details all the PIE/TSO commands general users can input from the MultiTSO menu, from TSO READY, or using the switch sequence or PIEEXEC command. These commands affect all PIE/TSO components. This chapter does not explain the commands you can input from the View command line.

For the following administrator commands, see the PIE/TSO Administrator Guide.

LOCATE
TRACE CARO
TRACE ACCESS

Commands are listed in alphabetical order. For information on command syntax, on entering commands, and for a categorical listing of commands, see Chapter 1, “Command Conventions."

CONENTS

ACCESS

ACCESS invokes Network Access. It allows you to access any VTAM application (CICS, IMS, PROFS, etc.) from PIE/TSO. You can switch to and from ACCESS sessions, just as you switch to and from regular MultiTSO sessions. For more information about using Network Access, please see the PIE/TSO User Guide.

ACCESS is ignored if Network Access is inactive.

FORMAT

```
ACCESS [applid]  [ data      ]
                 [ variables ]
```

Issue ACCESS as a TSO command. You may execute it from the MultiTSO menu only if SET IMPTSO is ON. To execute ACCESS when auto-end is on, issue it with a session opening command (e.g., START ACCESS).

PARAMETERS

applid Specify an APPLID to log on to. If you omit the APPLID, you will get the Network Access menu, and you may select an APPLID from the menu.

data Specify logon data to pass to your APPLID (for example, your user ID).

variables Use any of the following PIE/TSO and PIE/CICS variables to pass user-dependent information to the APPLID.

&ZUSER sends the user ID to a TSO APPLID.

&ZPSWD sends the user's password to a TSO APPLID.

&ZTSNON sends the user ID and password to an APPLID using PIE/TSO.

&ZTESNON sends the user ID and password to an APPLID using PIE/TSO. Use this variable if your installation encrypts sign on strings.

&ZSNON sends the user ID and password to an APPLID using PIE/CICS.

&ZESNON sends the user ID and password to an APPLID using PIE/CICS. Use this variable if your installation encrypts sign on strings.

☞ You can only send passwords to APPLIDs using the Logon Director. Users may set their passwords for any of these variables using the PIE/TSO SET PASSWORD command.

These are PIE/TSO and PIE/CICS variables, not CLIST, REXX, or ISPF variables. If you want to pass the &ZPSWD variable in a CLIST, for example, you would have to code &&ZPSWD in your CLIST.

Parameter	Valid	Default
applid	1 to 8 character APPLID	None
data	1 to 256 characters	None
variables	1 to 256 characters	None

EXAMPLES

To invoke Network Access and receive the Network Access menu, enter ACCESS from TSO READY.

To invoke Network Access and access the CICSTEST APPLID directly, enter ACCESS CICSTEST.

You want to start CICSPROD and pass your user ID and password to the CICS region so that PIE/CICS can automatically sign you on. Enter ACCESS CICSPROD &ZSNON.

You want to log on to your TSO system with a second user ID. (You might want to do this, for instance, if you have more than one TSO user ID or if a couple of programmers are trying to solve a problem from a single terminal.) Your second logon has the user ID JSMITH. Enter ACCESS TSO JSMITH.

ALLOC, @ALLOC, and ALLOCH

ALLOC opens a new session and assigns the allocated attribute to the session. When you normally end an allocated session, the session remains on the MultiTSO menu. To restart the session, simply enter its ID on the MultiTSO menu command line. MultiTSO will re-open the session and execute the TSO command used when you originally opened the session.

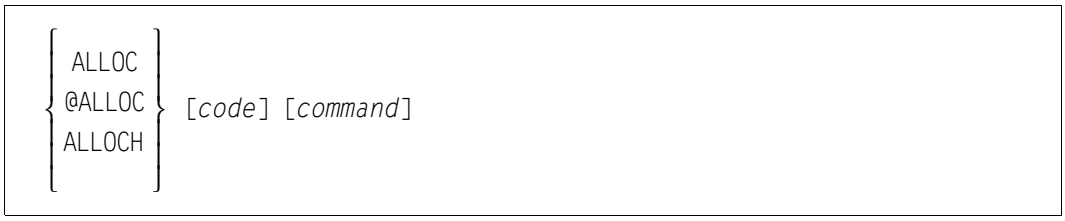
ALLOC does not switch to the session it opens.

@ALLOC is an alias for ALLOC and performs exactly the same function.

ALLOCH performs the same function as ALLOC, except it opens a hidden session.

The primary use of these commands is to open sessions from CLISTs and REXX EXECs.

FORMAT



Issue @ALLOC from the MultiTSO menu or as a TSO command. Issue ALLOC and ALLOCH from the MultiTSO menu or with PIEEXEC or with the switch sequence.

PARAMETERS

code Specify the code of the session you want to open. If you omit this parameter, MultiTSO will open the next inactive session. If you specify the ID of a session that is already active, MultiTSO will ignore the command.

command Specify a command to start in your new session, along with any relevant parameters. The command will execute after you switch to the session the first time. If you omit this parameter, MultiTSO will leave the session at the TSO READY prompt. If the session is already active, MultiTSO will ignore this parameter.

☞ If auto-end is on for this session and the application you start with the command parameter fails or you omit the command parameter entirely, auto-end will end your session automatically. You will be returned to the MultiTSO menu without a message. If this occurs, set auto-end off for the session.

Parameter	Valid	Default
code	0 to 11, A, or B	Next Available
command	1 to 256 character TSO command	None

EXAMPLES

To allocate session 6, enter ALLOC 6 from the MultiTSO menu.

To allocate the next session available and start ISPF in it, enter @ALLOC ISPF from TSO.

You want to allocate a hidden session with session ID 4 and prepare it to run FOCUS. Enter PIEEXEC ALLOCH 4 FOCUS DA('FOCUS.DATA') from the TSO READY prompt.

CANCEL and @CAN

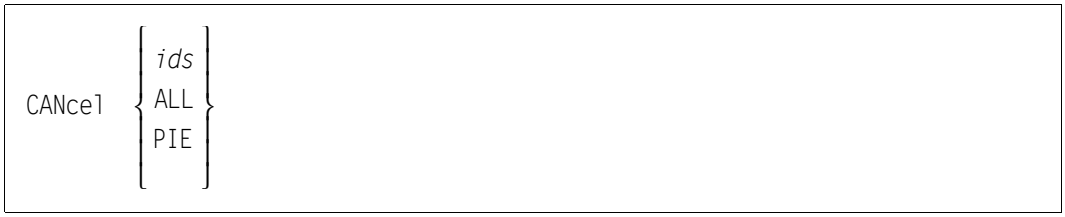
CANCEL abnormally terminates the application running in a session and ends the session. If you cancel all sessions, it will also log you off TSO.

⚠ WARNING
Because canceled sessions are abnormally terminated, you will lose all unsaved data in those sessions.

Sessions are abended with a system 222 abend. It appears as if the MVS console operator canceled the session with the MVS CANCEL command. However the MVS console operator will see no messages unless you cancel all sessions. Then the operator will see a message stating that you have logged off.

Your installation may have elected not to allow Network Access sessions to be canceled. In that case, you must normally terminate all Network Access sessions before executing CANCEL ALL or CANCEL PIE. If you specify a Network Access session by ID, you will get an error message.

FORMAT



Issue CANCEL from the MultiTSO menu, with PIEEXEC, or with the switch sequence. Issue @CAN as a TSO command or from the MultiTSO menu.

If you are in the MultiTSO menu, you may execute CANCEL with PF4 or PF16. First enter the parameter on the command line; then press the PF key. You may also cancel a session by placing the cursor next to the session in the command column, typing P (for purge), and pressing ENTER.

PARAMETERS

ids Specify the IDs of the sessions you want to cancel.

ALL Cancels all sessions and logs you off TSO.

PIE Cancels all sessions and logs you off PIE/TSO. You are returned to a normal TSO environment without PIE/TSO functions. To return to PIE/TSO, log off and back on to TSO.

Parameter	Valid	Default
ids	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None
ALL	ALL	None
PIE	PIE	None

EXAMPLES

You want to cancel all your sessions and log off TSO. Enter CAN ALL from the MultiTSO menu.

To cancel session 2, enter @CAN 2 from the TSO READY prompt.

You want to cancel all your sessions and quit PIE/TSO, but remain logged on to TSO. Enter PIE [PF4] from the MultiTSO menu.

CLOSE

CLOSE abends the application running in a session and ends the session. It leaves the session allocated on the MultiTSO menu.

To reopen a closed session, all you have to do is enter the session ID from the MultiTSO menu command line. MultiTSO will open the session with the same parameters you used to open it initially. For example, if you opened the session with START ISPF, when you close and reopen the session, MultiTSO will start the session and execute ISPF in it.

Use CLOSE when you need to free up resources for another session but you want to go back to the closed session later.

FORMAT

```
CLOSE { ids }
      { ALL }
```

Issue CLOSE from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may also close a session by placing the cursor next to the session in the command column, typing C, and pressing ENTER.

PARAMETERS

ids Specify the IDs of the sessions you want to close.

ALL Closes all sessions. It does not log you off TSO or sign you off PIE/TSO.

Parameter	Valid	Default
ids	0 to 11, A, B, *, ?, or 1 to 8 character name	None
ALL	ALL	None

EXAMPLES

To close the session named ISPF, enter CLOSE ISPF.

To close session 2, enter CLOSE 2.

To close all sessions, enter CLOSE ALL.

CREATE and CREATEH

CREATE opens a new session without switching to it.

CREATE is similar to the ALLOC command, with the following difference. When you normally end a session opened with ALLOC, the session is immediately re-allocated, just as if you had reentered the ALLOC command. When you end a session opened with CREATE, the session is wiped out of existence.

CREATEH performs the same functions as CREATE, except that it opens a hidden session.

CREATE and CREATEH are especially useful in CLISTs and REXX EXECs.

FORMAT

```
{ CREATE  }
{ CREATEH }  [code] [command]
```

Issue CREATE and CREATEH as TSO commands or from the MultiTSO menu. You may also issue CREATE from the MultiTSO menu with PF6 or PF18. Type any parameters before pressing the PF key.

PARAMETERS

code Specify the code of the session you want to open. If you omit the parameter, MultiTSO will open the next inactive session. If you specify the ID of the session that is already active, MultiTSO will ignore the command.

command Specify a command to start in your new session, along with any relevant parameters. The command will execute when you first switch to the session. If you omit this parameter, MultiTSO will leave the session at the TSO READY prompt. If the session is already active, MultiTSO will ignore this parameter.

☞ If auto-end is on for this session and the application you start with the command parameter fails or you omit the command parameter entirely, auto-end will end your session automatically. You will be returned to the MultiTSO menu without a message. If this occurs, set auto-end off for the session.

Parameter	Valid	Default
code	0 to 11, A, or B	Next Available
command	1 to 256 character TSO command	None

EXAMPLES

To create the next available session and prepare it to run ISPF, enter CREATE ISPF.

To open session 1 as a hidden session, enter CREATEH 1.

To open session 4 and run FOCUS in it, enter CREATE 4 FOCUS DA('FOCUS.DATA') from the MultiTSO menu.

DISC

DISC disconnects your TSO logon. Use it when you want to move to a different terminal and resume your TSO logon. All sessions and unsaved data will remain intact. To reconnect, log on using the Reconnect option at the bottom of your logon screen.

Imagine, for example, you need help from another programmer in debugging a problem. You enter DISC from your current terminal. You go to the other programmer's office and log on from the terminal there, using the Reconnect option. All your sessions will resume on the new terminal, intact.

You can also use DISC when someone needs to log on at the terminal you are using. You can disconnect, allow them to log on, and then reconnect when they are finished.

FORMAT

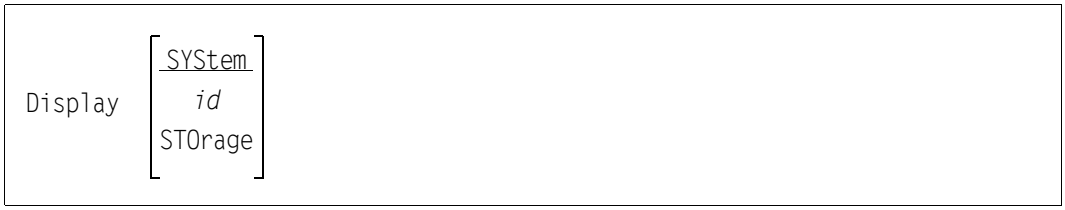
```
DISC
```

Issue DISC from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

DISPLAY

DISPLAY displays information about your sessions, the CPU, and storage utilization. If Network Access is active, it will display the name of the virtual terminal in use.

FORMAT



Issue DISPLAY from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PARAMETERS

SYSTEM Displays the level of MVS running on this CPU. This is the default option.

id Specify a session ID to display information about a particular session.

STORAGE Displays the region size, the total available storage, and the largest amount of contiguous storage available.

Parameter	Valid	Default
SYSTEM	SYS or SYSTEM	SYSTEM
id	0 to 11, A, B, *, ?, or 1 to 8 character name	None
STORAGE	STO or STORAGE	None

EXAMPLE

Session 6 is a Network Access session to CICS. CICS has not responded to the last transaction entered. You want to call the Network Control Center to find out why. The NCC operator needs to know your virtual terminal ID. To find out, you switch to the MultiTSO menu and enter DISPLAY 6. The response will be

```
ACCESS SESSION 6 - PIE10001 TO CICS - FROM 10:45
```

Your virtual terminal ID is PIE10001. You accessed CICS at 10:45.

Now, you have 5 sessions active and you want to open another that needs 200K. Do you have that much left? To find out, switch to the MultiTSO menu and enter D STO. The response will tell you how much storage you have left in your TSO region. For example:

```
Region = 06500K, Available = 03296K, Contig = 03232K
```

END

END pushes the current session to the bottom of the stack and switches you to the session that was second from the top of the old stack.

What does this mean?

The stack is the order in which you have switched to sessions and the MultiTSO menu. Let's say you start sessions 0, 1, 2, 3, and 4 in order. Your initial stack order is:

MultiTSO menu, 0, 1, 2, 3, 4

where the MultiTSO menu is the bottom of the stack and session 4 is the top of the stack. Let's say you switched from session 4 to session 2. Your new stack would be:

MultiTSO menu, 0, 1, 3, 4, 2

When you switch to session 2, you move it to the top of the stack. Everything else moves down one.

Now, END pushes the current session to the bottom of the stack and switches you to the session that was second from the top of the old stack. Let's say you issue END from session 2, your new stack order would be:

2, MultiTSO menu, 0, 1, 3, 4

Session 2 is pushed to the bottom of the stack and you are switched to session 4, the second to the top session from the old stack.

FORMAT

```
END
```

Issue END from the session with a sessions PF key, PIEEXEC, or the switch sequence.

ENDBACK

ENDBACK is similar to the END command. It pushes the current session to the bottom of the stack and switches you to the session that is third from the top of the old stack.

See END for an explanation of the stack concept.

To put the ENDBACK command in practical terms, let's say your stack is:

MultiTSO menu, 0, 1, 2, 3, 4

where the MultiTSO menu is the bottom of the stack and 4 is the top. You ENDBACK from session 4. Your new stack order would be:

4, MultiTSO menu, 0, 1, 3, 2

Session 4 is pushed to the bottom of the stack, and you are switched to session 2, the session that was third from the top in the old stack. Then you ENDBACK from session 2. Your new stack order is:

2, 4, MultiTSO menu, 0, 3, 1

Session 2 goes to the bottom of the stack, and you access session 1, which was third from the top in the old stack.

FORMAT

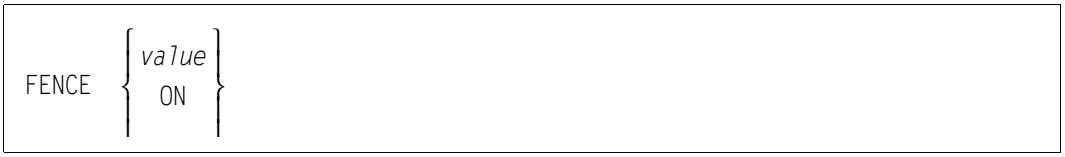
```
ENDBACK
```

Issue ENDBACK from the session with a sessions PF key, PIEEXEC, or the switch sequence.

FENCE

Some TSO applications, such as APL2, take all the available storage in a region when they initialize. This leaves you no storage to open new sessions. However you can use FENCE to reserve some storage from those applications so that you can open new sessions later.

FORMAT



Issue FENCE from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PARAMETERS

value Changes the amount of storage fenced. You may specify up to 2M. If you omit this value, the FENCE start-up option determines how much storage is reserved. The default is 256K. You must change the FENCE value before you issue FENCE ON.

ON Reserves the storage. When you switch out of the session the reserved storage will be freed.

Parameter	Valid	Default
value	1 to 2048[K], 1M, or 2M	See text
ON	ON	None

EXAMPLE

Our example uses APL2, since APL2 will take all available storage in the region.

1. You want to change the amount of storage fenced to 1M from the default amount set by the FENCE start-up option. Enter FENCE 1024 before entering FENCE ON.
2. Before you start APL2, enter FENCE ON from the MultiTSO menu. MultiTSO will reserve 1M storage in your region.
3. Open a session and start APL2.

   When you switch out of the session the reserved storage will be freed. You will have memory available, and you may open a new session.

FREE

FREE removes the allocated attribute from a session. When you end the session, it will be removed from the MultiTSO menu.

FREE is the default when you open your session with START, STARTH, CREATE, or CREATEH. Use FREE if you opened the session with ALLOC or ALLOCH or to undo a previous KEEP command.

FORMAT

```
FREE  { ids }
      { ALL }
```

Issue FREE from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PARAMETERS

ids Specify the IDs of the sessions you want to free.

ALL Frees all sessions.

Parameter	Valid	Default
ids	0 to 11, A, B, *, ?, or 1 to 8 character name	None
ALL	ALL	None

EXAMPLES

To free all sessions, enter FREE ALL.

To free sessions 5 and 9, enter FREE 5 9.

H

H starts PIE/TSO's full-screen HELP. HELP runs in a special session that does not count towards your maximum number of sessions. Its session ID is H. For more information on the HELP session, see the PIE/TSO User Guide.

FORMAT

```
H
```

Issue H from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

HIDE

HIDE "hides" sessions from the jump key. Suppose you have sessions 0, 1, 2, and 3 open, and none are hidden. Your jump key will cycle through all sessions. If you hide session 2, your jump key will skip session 2 and cycle through sessions 0, 1, and 3 only.

HIDE is useful when you want to switch quickly between a few sessions. You can hide all other sessions temporarily.

A session remains hidden until you end it, log off TSO, or unhide it with the UNHIDE command.

FORMAT

```
HIde { ids }
     { ALL }
```

Issue HIDE as a TSO command or from the MultiTSO menu.

PARAMETERS

ids Specify the IDs of the sessions you want to hide.

ALL Hides all sessions.

Parameter	Valid	Default
ids	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None
ALL	ALL	None

EXAMPLES

To hide all sessions, enter HIDE ALL from TSO READY.

To hide sessions 3, 4, and 10, enter HI 3 4 10 from the MultiTSO menu.

I

I starts the INFO session. INFO is a special session that does not count towards your maximum number of sessions. The INFO session ID is I. See the PIE/TSO User Guide for more information on the INFO session.

FORMAT

```
I
```

Issue I from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

JMKEY

JMKEY activates the jump key and assigns it to a PF key. This setting will remain in effect until you change it or until you log off TSO.

By default, your jump key is off.

JMKEY does not duplicate the jump PF key in the high or low set of PF keys. So, for instance, PF12 means only PF12, not PF24 too.

The jump key overrides the corresponding PF key set in your application. It also overrides MultiTSO menu and sessions PF keys.

The default for JMKEY is set with the JMKEY start-up option.

FORMAT

```
JMKey key
```

Issue JMKEY from the MultiTSO menu or as a TSO command.

PARAMETERS

key Specify any PF key number, either in the format PF1 or 1.

Parameter	Valid	Default
key	PF1 to PF24 or 1 to 24	None

EXAMPLE

To activate your jump key and set it to PF4, enter JMK 4 from the MultiTSO menu.

KEEP

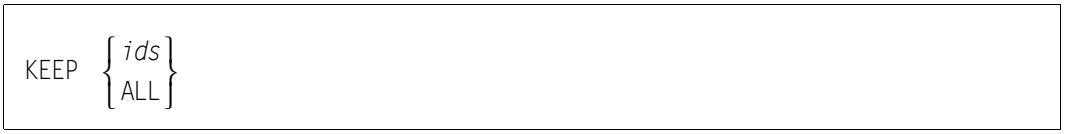
KEEP assigns the allocated attribute to a session. When you end an allocated session, it will remain on the MultiTSO menu.

To re-open an allocated session, all you have to do is enter the session ID from the MultiTSO menu command line. MultiTSO will open the session with the same parameters you used to open it initially. For example, if you opened the session with START ISPF, when you reopen the session, MultiTSO will start the session and execute ISPF in it.

KEEP is the default if you open the session with ALLOC or ALLOCH. Use it when you open a session with START, STARTH, CREATE, or CREATEH.

To undo KEEP, issue FREE.

FORMAT



Issue KEEP from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PARAMETERS

ids Specify the IDs of the sessions you want to keep.

ALL Keeps all sessions.

Parameter	Valid	Default
ids	0 to 11, A, B, ?, *, or 1 to 8 character name	None
ALL	ALL	None

EXAMPLES

To keep all sessions, enter KEEP ALL.

To keep sessions 5 and 9, enter KEEP 5 9.

LOCK

LOCK prevents access to any session until the correct user password is entered. When your terminal is locked, you can walk away from it, and no one will be able to use your user ID.

When you issue the LOCK command, you will get the message:

```
PCS371A - TERMINAL LOCKED, PLEASE CONFIRM PASSWORD FOR userid
```

To unlock your terminal, enter your password. MultiTSO will verify your password with your security system. If you enter the correct password, your terminal will be unlocked. Your sessions will be just the way you left them.

If you enter the wrong password, you will get the message:

```
PASSWORD NOT AUTHORIZED FOR USERID
```

You may then try to enter the password again.

Your PIE/TSO administrator set a limit to the number of times you can input passwords incorrectly. If that limit is exceeded, MultiTSO will disconnect your logon to prevent unauthorized access.

Your PIE/TSO administrator can set up LOCK to be invoked automatically. If this is the case, your terminal will automatically lock if it is idle—you haven't pressed ENTER or a function key—for the specified period of time.

FORMAT

```
LOCK msgtext
```

Issue LOCK from the MultiTSO menu or as a TSO command.

PARAMETER

msgtext Specify any message to display on your locked terminal. Use this parameter to let people know where you are, when you will be back, etc.

Parameter	Valid	Default
msgtext	1 to 95 characters	None

EXAMPLES

To lock your terminal, enter LOCK.

To lock your terminal and leave a message for those who come to call on you, enter:
LOCK I WILL RETURN AT THREE PM

LOGON

LOGON cancels all sessions, logs you off, and logs you back on to TSO. You will not see the VTAM logo.

LOGON is helpful if you want to change your logon procedure or you just want to start afresh with PIE/TSO.

FORMAT

```
LOGON [userid] [parms]
```

Issue LOGON from the MultiTSO menu, with PIEEXEC, or with the switch sequence. If you issue LOGON any other way, you will be executing the TSO LOGON command.

PARAMETERS

userid Specify a user ID to change the user ID you are logged on with.

parms Specify any TSO logon parameters—for example, your logon procedure. Most security systems will reject your password.

Parameter	Valid	Default
userid	valid user ID	None
parms	logon parameters	None

EXAMPLE

You want to end all your MultiTSO sessions and log on as JSMITH. Enter LOGON JSMITH from the MultiTSO menu.

MSPF and MULTISPF

MSPF allows you to run ISPF in multiple sessions.

If you run MSPF in your first session, MultiTSO will invoke ISPF directly. If you run MSPF in any other session, MSPF will create preallocated ISPF utility datasets, different from the default datasets so that your sessions will not interfere with each other. Then it will start ISPF.

MSPF is an alias for MULTISPF.

FORMAT

```
{MULTISPF}
{ MSPF  }  [parms]
```

Issue MULTISPF or MSPF as a TSO command. You may execute them from the MultiTSO menu only if SET IMPTSO is on. To execute them when auto-end is on, issue them with a session opening command (e.g., START MSPF).

PARAMETER

parms Specify any ISPF command parameters. They will be passed directly to ISPF.

Parameter	Valid	Default
parms	ISPF parameters	None

EXAMPLE

To invoke ISPF and go to ISPF menu option 2, enter MSPF 2 from TSO READY.

If you like, you can pass parameters to ISPSTART. For example, to allocate a session that will present panel ZSDSFOP2, code

ALLOC MSPF PANEL(ZSDSFOP2) NEWAPPL(ISF)

To allocate a session that will present the RMFWDM command, code

ALLOC MSPF CMD(RMFWDM)

To allocate a session that will present the program ICEPMD01, code

ALLOC MSPF PGM(ICEPMD01) NEWAPPL(ICE) NOCHECK

NAME

NAME assigns names to sessions. You can assign any session a descriptive name to make it easier to identify. Then you can use the session name wherever a session ID is needed.

So instead of having to remember that SDSF is running in session 2 and ISPF EDIT in session 3, you can name the sessions SDSF and EDIT. Then when you want to switch to the EDIT session, you can issue the switch command with the name EDIT:
\EDIT [ENTER]

If you set up MultiTSO PF keys, you can assign the name of your most commonly used applications to PF keys, and switch to those applications by pressing the key.

By default, PIE/TSO assigns names to sessions automatically from the TSO commands executed in the sessions. (This is controlled with the SET DFLTNAME command.) These names change as you execute new TSO commands in the session. For example, if you issue LISTA in session 2, session 2's name will be LISTA. If you later issue the LISTC command in that session, its name will change to LISTC.

The names you assign with NAME remain until you change them with NAME.

FORMAT

```
NAME id name
```

Issue NAME from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PARAMETERS

id Specify the ID of the session you want to name.

name Specify the session's new name. To remove the name from a session, leave this field blank.

Parameter	Valid	Default
id	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None
name	1 to 8 characters	None

EXAMPLES

You are running ISPF EDIT in session 0. To name session 0 EDIT, enter NAME 0 EDIT.

Your session 0 line might read:

```
PIE MultiTSO Menu ---------------------------------------- (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   EDIT      ISPF 2                                    S  SwKey:   ENTER
```

To remove the name you assigned to session 0, enter NAME 0.

Note that if SET DFLTNAME is on, the session's name will revert to the default name. If that was the case, your session 0 line might read:

```
PIE MultiTSO Menu ---------------------------------------- (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   MSPF      ISPF 2                                    S  SwKey:   ENTER
```

NOTE

NOTE assigns a comment to a session for display on the MultiTSO menu. Use this comment to help identify what is running in the session.

By default, MultiTSO will use the command information you specify when you open a session with ALLOC, CREATE, START, etc. You can use NOTE to change these comments.

FORMAT

```
NOTE id [comments]
```

Issue NOTE from the MultiTSO menu or as a TSO command. When you are in the MultiTSO menu, you can create and modify comments by typing over them.

PARAMETERS

id Specify the session you want to assign a comment to.

comments Specify your comment.

Parameter	Valid	Default
id	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None
comments	1 to 40 characters	None

EXAMPLE

To identify RMF3 and PROFS, enter NOTE 1 RMF Monitor III and NOTE 2 Electronic Mail System.

```
PIE MultiTSO Menu ------------------------------------------ (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
+ 0   LISTC     LISTC                                    S  SwKey:   ENTER
+ 1   RMF3      RMF Monitor III                          S  Keys:   OFF
* 2   PROFS     Electronic Mail System                   S  Mkeys:  M ON
```

PEXEC

PEXEC is an interface to the PIE/TSO Performance Group Manager (PGM).

PGM contains a table that assigns performance group numbers and SWAP/NOSWAP status to non-ISPF applications. These assignments are set up during installation. For more information on them, see the PIE/TSO Installation and Customization Guide.

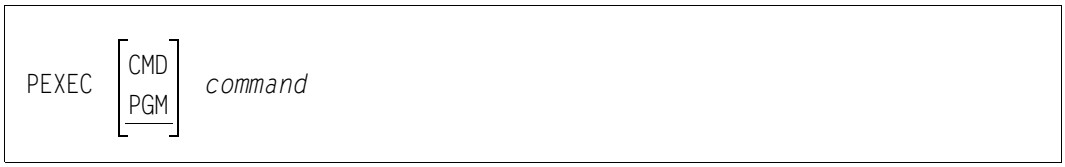
PGM does not change the performance group and SWAP/NOSWAP assignments for an application unless you tell it to. You can tell PGM to change the assignment in either of two ways:

- by executing the application with the PEXEC command
- by defining the application as an alias to PEXEC

If the application is an alias to PEXEC, PGM will change the performance group assignments automatically when the application is requested. If it is not an alias, you must execute it as a parameter to the PEXEC command.

When you exit the application, your performance group number and swap values will be reset to their original values. The performance group number and swap values are maintained by session and are set and reset as you switch from session to session.

FORMAT



Issue PEXEC as a TSO command. You may also call it as an MVS program. You may execute PEXEC from the MultiTSO menu only if SET IMPTSO is on. To execute it when auto-end is on, issue it with a session opening command (e.g., START PEXEC CMD ACCESS).

PARAMETERS

CMD Identifies the command as a TSO command processor, CLIST, or REXX EXEC.

PGM Identifies the command as an MVS program.

command Specify the name or alias of the command and any relevant parameters.

If you enter an invalid command, PEXEC will display error messages providing information about the error. These messages are explained in the IBM manual TSO Guide to Writing a Terminal Monitor Program or a Command Processor SC28-1136 in the chapter on invoking other programs, commands, or CLISTs with the TSO Service Routine.

Parameter	Valid	Default
CMD|PGM	CMD or PGM	PGM
command	1 to 256 character TSO command	None

EXAMPLE

You want to execute ACCESS with the CICS170 APPLID. You want to use the performance group values in the PGM table. Enter PEXEC CMD ACCESS CICS170.

ACCESS will start and the performance group number and swap values will be set as specified in the PGM table.

PFSHOW

PFSHOW displays the function key help line on the MultiTSO menu. You may display the MultiTSO menu keys there, or you may display the PF keys that are active in your sessions.

The default for PFSHOW is set with the PFSHOW start-up option.

FORMAT

```
PFSHOW [ ON     ]
       [ OFF    ]
       [ type   ]
       [ range  ]
```

Issue PFSHOW from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PARAMETERS

ON|OFF Displays or turns off the PF key help line. If you omit all PFSHOW parameters, MultiTSO will turn the display off if it is on or turn it on if it is off.

type Indicate which keys to display. Specify either MKEYS to display the menu function keys or SKEYS to display the session function keys. If you omit this parameter, it will default to the type you used the last time you executed the PFSHOW command. If this is the first time you are executing PFSHOW during this logon, it will default to MKEYS.

range Indicate the range of keys to display. PA2 displays the PA2 and PA3 keys. PF1 and F1 display PF keys 1 through 12. PF13 and F13 display PF keys 13 through 24. If you omit this parameter, it will default to the range you used the last time you executed PFSHOW. If this is the first time you are executing PFSHOW, it will default to PF1.

Parameter	Valid	Default
ON|OFF	ON or OFF	See text
type	MKEYS or SKEYS	MKEYS
range	PA2, PF1, F1, PF13, or F13	PF1

EXAMPLES

To turn off the function key display, enter PFSHOW OFF.

To toggle the PF key display on and off, enter PFSHOW.

To display the session keys, enter PFSHOW SKEYS.

To display PF keys 13 through 24, enter either PFSHOW PF13 or PFSHOW F13.

PIECHECK

PIECHECK determines whether PIE/TSO is active. It passes a return code to notify you of PIE/TSO status. It is helpful in CLISTs, EXECs, and programs. For instance, you can use PIECHECK in an initial CLIST to allocate certain datasets only if PIE/TSO is active.

FORMAT

```
PIECHECK [code]
```

Issue PIECHECK from CLISTs or REXX EXECs as a TSO command. Call it from an assembler program through the LINK macro and clear register 1 to zero. If you don't, PIECHECK will abend with an 0C4.

PARAMETER

code Specify the one character code of the session you want to check. For session 10, specify A. For session 11, specify B.

Parameter	Valid	Default
code	0 to 9, A, B, H, I, V, *, ?	Current session

RETURN CODES

If you issue PIECHECK without a session ID, the return code will indicate which session is the current active session. The possible return codes are:

- 0–MultiTSO not active
- 240–session 0 current (hex value F0)
- 241–session 1 current (hex value F1)
- 242–session 2 current (hex value F2)
- 243–session 3 current (hex value F3)
- 244–session 4 current (hex value F4)
- 245–session 5 current (hex value F5)
- 246–session 6 current (hex value F6)
- 247–session 7 current (hex value F7)
- 248–session 8 current (hex value F8)
- 249–session 9 current (hex value F9)
- 193–session 10 current (hex value C1)
- 194–session 11 current (hex value C2)

If you issue PIECHECK with a session ID, the return code will indicate the status of that session. The possible return codes are:

- 0–MultiTSO not active
- 4–session is not open
- 8–session is open
- 12–session ID is not valid

PIEEXEC

The PIEEXEC command allows you to issue PIE/TSO commands from TSO READY or from applications as TSO commands. For instance, you can execute the FENCE command from TSO READY as PIEEXEC FENCE, and it would execute just as though you had entered it directly from the MultiTSO menu.

You can also use PIEEXEC to avoid assigning aliases to PIE/TSO commands that have the same name as TSO commands. When you execute the command using PIEEXEC, PIE/TSO will execute its own command. For instance, if you want to execute PIE/TSO CANCEL instead of TSO CANCEL, you can enter PIEEXEC CANCEL ALL. This will execute PIE/TSO's CANCEL ALL command, just as if you had entered CANCEL ALL from the MultiTSO menu.

PIEEXEC is not necessary for the following commands:

ACCESS*	MSPF and MULTISPF*	START and STARTH
@ALLOC	NOTE	STOP*
@CAN	PEXEC*	SWCHAR
CREATE and CREATEH	PIECHECK*	SWITCH
HIDE	PIESTART*	SWKEY
JMKEY	SCREEN	UNHIDE
LOCK	All @SET commands	

The commands marked with an asterisk (*) are TSO command processors. Normally you would execute these commands directly from TSO READY or from an application as TSO commands. The rest of the commands are aliases to PIEEXEC—so you don't need to use PIEEXEC when you execute them.

If you execute any of these commands with the PIEEXEC command, MultiTSO will open a new session and execute the command from that session. (This is controlled by the SET IMPTSO command.)

PIEEXEC is also useful in CLISTs and REXX EXECs. If you use PIEEXEC in a CLIST/EXEC and the command executes without error, the CLIST/EXEC will resume execution when the command completes. If an error is encountered executing the command, the MultiTSO menu will be displayed with the error message.

FORMAT

```
PIEEXEC command
```

Issue PIEEXEC as a TSO command from a session or in a CLIST or REXX EXEC.

PARAMETERS

command Specify a PIE/TSO command with its parameters.

Parameter	Valid	Default
command	Any PIE/TSO command and parameters	None

EXAMPLE

You want to cancel all your sessions and log off TSO. You are in a TSO session, but you want to use PIE/TSO CANCEL, not TSO CANCEL. Enter PIEEXEC CANCEL ALL.

PIESTART

PIESTART is a special command that is useful in CLISTs, REXX EXECs, PF keys, and ISPF menus to open or switch to a session.

When you execute PIESTART, you identify the session by name. If the session does not yet exist, PIESTART will execute START and NAME, to open and switch to the next available session and to give it the name you indicated. If the session already exists, PIESTART will switch you to it.

FORMAT

```
PIESTART //name/command
```

Issue PIESTART as a TSO command.

PARAMETERS

name Specify the session name. If the session exists, MultiTSO will switch you to it. If it doesn't exist, MultiTSO will start the session and give it the name you specify here.

command Specify a TSO command and its parameters. If the session already exists, MultiTSO will ignore this command.

Parameter	Valid	Default
name	1 to 8 characters	None
command	1 to 256 character TSO command	None

EXAMPLE

You want to go to an ISPF session named EDIT in a CLIST, but you have no way of knowing whether it exists. You code PIESTART //EDIT/MSPF 3.2.

PUSH

PUSH is an alias for END. See “END” for more information.

PUSHJUMP

PUSHJUMP is an alias for ENDBACK. See “ENDBACK” for more information.

RECALL

RECALL displays the last command you executed from the MultiTSO menu or from a session. You can then change the command and execute it again.

FORMAT

```
Recall
```

Issue RECALL from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

RESHOW

RESHOW refreshes the MultiTSO menu.

FORMAT

```
RESHOW
```

Issue RESHOW from the MultiTSO menu.

RESTART

RESTART is the equivalent of executing CANCEL and START. It abends the transaction running in a session, exits the session, and then restarts the session. RESTART will not re-execute an initial command unless you specify the command as a parameter to RESTART.

FORMAT

```
REStart id [command]
```

Issue RESTART from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

PARAMETERS

id Specify the ID of the session you want to restart.

command Specify an initial TSO command and its parameters for sessions 0-11.

☞ If auto-end is on for this session and the application you start with the command parameter fails or you omit the command parameter entirely, auto-end will end your session automatically. You will be returned to the MultiTSO menu without a message. If this occurs, set auto-end off for the session.

Parameter	Valid	Default
id	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None
command	1 to 256 character TSO command	None

EXAMPLE

To cancel and then re-start session 6, enter RESTART 6. When it has been restarted, the session will end up at the TSO READY prompt.

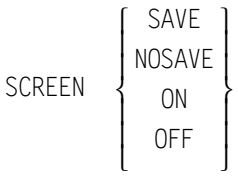
To restart session 4 and run ISPF 2 in it, enter RESTART 4 ISPF 2.

SCREEN

It is possible to type data on your screen and then switch out of the session before you press ENTER or a PF key to save the data you typed. SCREEN determines how that data is treated. You may either keep or discard it.

You may also use SCREEN to control the display of the MultiTSO menu header line.

FORMAT



Issue SCREEN from the MultiTSO menu or as a TSO command.

PARAMETERS

SAVE|NOSAVE Determines what happens if you have data on your screen that you haven't saved by pressing ENTER or a PF key and you switch to another session. If you specify SAVE, MultiTSO will effectively “press ENTER” and save your data before it switches out of the session. Use NOSAVE if pressing ENTER has unwanted effects in your application. If you specify NOSAVE, all unentered data will be discarded when you switch.

☞ If you are running SCREEN NOSAVE, always press ENTER or a PF key before you switch out of the session.

ON|OFF Displays or turns off the header line.

Parameter	Valid	Default
SAVE|NOSAVE	SAVE or NOSAVE	None
ON|OFF	ON or OFF	None

EXAMPLES

To display the header line, enter SCREEN ON.

To keep unsaved data when you switch out of a session, enter SCREEN SAVE.

SET AATTN

SET AATTN assigns the action of the ATTN key in Network Access sessions. This allows you to control whether the ATTN key will be processed as usual by the APPLID or whether it will switch you to the MultiTSO menu.

FORMAT

```
SET AATTN {SWITCH}
          { ATTN }
```

Issue SET AATTN from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET AATTN as a TSO command.

PARAMETERS

SWITCH Sets the ATTN key to the SWITCH command in Network Access sessions.

ATTN Restores the ATTN key to its original use by each application in Network Access sessions. Use it to revert the ATTN key from SWITCH.

Parameter	Valid	Default
SWITCH|ATTN	SWITCH or ATTN	None

EXAMPLES

To set the ATTN key in Network Access sessions to the SWITCH command, enter SET AATTN SWITCH.

To pass the ATTN key through Network Access to the application, enter SET AATTN ATTN.

SET ABBREV

SET ABBREV allows you to use abbreviations for session names.

If abbreviations are turned off, you must specify the entire session name to identify a session. If abbreviations are on, you can identify a session by the first character or characters in its name.

Abbreviating names is obviously an effort-saver, but it can be confusing. You may not always receive the session you expect when you abbreviate a name. For the rules for matching names with sessions, see “Identifying Sessions,” on page 6.

Whenever you are in doubt, use the full session name or its code.

The default for SET ABBREV is set with the ABBREV start-up option.

FORMAT

```
SET ABBREV { ON  }
           { OFF }
```

Issue SET ABBREV from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET ABBREV as a TSO command.

PARAMETER

ON|OFF Enables or disables abbreviations.

Parameter	Valid	Default
ON|OFF	ON or OFF	None

EXAMPLES

To allow abbreviated session names from the MultiTSO menu, enter SET ABBREV ON.

To prohibit abbreviated session names from TSO READY, enter @SET ABBREV OFF.

SET AE

SET AE controls the auto-end attribute. With auto-end on, MultiTSO automatically ends the session if you log off the application running in it or if the application abends. After it ends the session, MultiTSO switches you to the last session you were in.

Auto-end avoids the TSO READY prompt and the "blank screen." If you set up sessions automatically with initial CLISTs or REXX EXECs, and set auto-end on for all sessions, the need for user expertise is greatly reduced. Users don't have to know how to start an application, and they don't have to know what to do at TSO READY.

The default for SET AE is set with the AE start-up option.

FORMAT

```
SET AE { ON  } [ ids ]
       { OFF } [ ALL ]
```

Issue SET AE from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET AE as a TSO command.

PARAMETERS

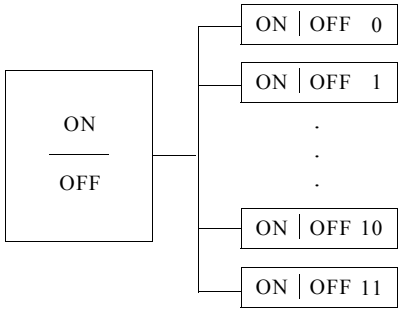
SET AE works on a breaker concept, with a main breaker and individual breakers for each session. For auto-end to occur in a session, the main breaker and the session breaker must both be on. You can turn the main breaker on or off and leave the session breakers as they are. You can change session breakers without affecting the main breaker.

ON|OFF Turns a breaker on or off.

ALL|ids Specifies what breakers to turn on or off.

Use ALL to change the settings for the main breaker and for all sessions (e.g., SET AE ON ALL).

If you want to change specific session breakers, specify their session IDs (e.g., SET AE ON 1). When you specify an ID, you change only the session breaker—you do not change the main



To change the main breaker only, omit both ids and ALL (e.g., SET AE ON).

Parameter	Valid	Default
ON|OFF	ON or OFF	None
ALL|ids	ALL, 0 to 11, A, B, ?, *, or 1 to 8 character name	None

EXAMPLES

To set auto-end on only in the session named ISPF, enter SET AE ON ISPF and SET AE ON.

To set auto-end on in all sessions, enter @SET AE ON ALL.

SET APFn

SET APFn assigns the ATTN key function to a PF key in Network Access sessions. It is useful for non-SNA terminals so that the ATTN key can be passed to the Network Access SNA terminal.

SET APFn does not affect PF keys in other sessions.

FORMAT

```
SET APFn [ATTN]
```

Issue SET APFn from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET APFn as a TSO command.

PARAMETERS

n Specifies which PF key to act as the ATTN key.

ATTN Sets the specified PF key to the ATTN key. If you omit it, the PF key will return to its normal function.

Parameter	Valid	Default
n	1 to 24	None
ATTN	ATTN	None

EXAMPLES

To make PF1 act like the attention key in Network Access sessions, enter SET APF1 ATTN.

To return the PF1 key to its normal function in the Network Access session, enter SET APF1.

SET ASYS

SET ASYS allows you to suspend your current application and switch to the MultiTSO menu when your application is still processing.

You could use asynchronous switching to move to the MultiTSO menu and cancel your task. Or, you could be in the middle of a long running task and get a call from someone needing information. You could suspend your task, get the information, and switch back to the task. When you return, your task will resume processing.

To switch out of a running session:

- From an SNA terminal—press ATTN.
- From a non-SNA terminal—press RESET and then PA1.

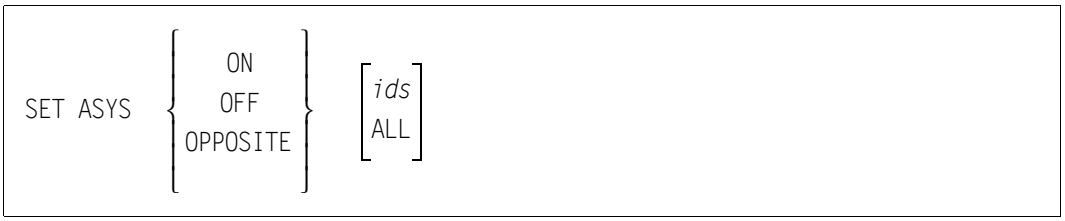
When asynchronous switching is on, it overrides the normal use of your ATTN key (or RESET/PA1 keys) for the session in which it is active. To attention out of your application, press ATTN twice or press RESET, PA1, PA1.

There are several circumstances under which asynchronous switching will not work. It is disabled:

- at the VTAM page prompt (***)
- by certain enqueues
- by a system must complete condition
- by a step must complete condition

Most of the time you will not be able to tell that you are in a system or step must complete condition or that the enqueue in effect prohibits switching. So, if asynchronous switching doesn't work, don't assume that it is not functioning properly. Simply wait a moment or two and try it again.

FORMAT



Issue SET ASYS from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET ASYS as a TSO command.

PARAMETERS

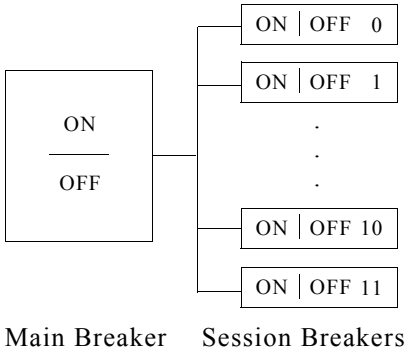
SET ASYS works on a breaker concept, with a main breaker and individual breakers for each session. For asynchronous switching to occur in a session, the main breaker and the session breaker must both be on. You can turn the main breaker on or off and leave the session breakers as they are. You can change session breakers without affecting the main breaker.

ON|OFF|OPPOSITE Turns a breaker on or off. You can assign OPPOSITE to a PF key and toggle asynchronous switching on and off.

ALL|ids Specifies what breakers to turn on or off.

Use ALL to change the settings for the main breaker and for all sessions (e.g., SET ASYS ON ALL).

If you want to change specific session breakers, specify their session IDs (e.g., SET ASYS ON 1). When you specify an ID, you change only the session breaker—you do not change the main



To change the main breaker only, omit both ids and ALL (e.g., SET ASYS ON).

Parameter	Valid	Default
ON|OFF|OPPOSITE	ON, OFF, or OPPOSITE	None
ALL|ids	ALL, 0 to 11, A, B, ?, *, or 1 to 8 character name	None

EXAMPLES

To set up session 2 as an asynchronous switching session, enter SET ASYS ON 2. Now to activate asynchronous switching for session 2, enter SET ASYS ON.

To disable asynchronous switching from all sessions, enter SET ASYS OFF ALL.

SET CMDDEFLT

SET CMDDEFLT sets up a default TSO command to execute in every session you open. You may want to put this command in your initial CLIST or REXX EXEC.

You can override the default command by specifying a TSO command as a parameter when you open the session with ALLOC, CREATE, or START.

The default for SET CMDDEFLT is set with the DCMD start-up option.

FORMAT

```
SET CMDDEFLT command
```

Issue SET CMDDEFLT from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET CMDDEFLT as a TSO command.

PARAMETER

command Specify the default TSO command and its parameters.

Parameter	Valid	Default
command	1 to 256 character TSO command	None

EXAMPLE

Enter SET CMDDEFLT MSPF to automatically start MSPF whenever you open a new session.

SET CMDFINAL

SET CMDFINAL sets up a default final TSO command to execute when you have logged off all your sessions. The final command doesn't execute if you use CANCEL to end your last session.

You might want to put this command in your initial CLIST or REXX EXEC.

The default for SET CMDFINAL is set with the FCMD start-up option.

FORMAT

```
SET CMDFINAL command
```

Issue SET CMDFINAL from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET CMDFINAL as a TSO command.

PARAMETER

command Specify the final TSO command and its parameters.

Parameter	Valid	Default
command	1 to 256 character TSO command	None

EXAMPLE

Enter SET CMDFINAL LOGOFF to log off TSO when all MultiTSO sessions have been logged off.

SET DFLTNAME

SET DFLTNAME determines whether MultiTSO will give sessions default names when you open them. If it is on, it will use the last TSO command you entered in a session as the session's name. For instance, if you execute LISTC in session 1, MultiTSO will name session 1 LISTC. If you then execute LISTA in session 1, MultiTSO will change the name to LISTA.

To assign a session a permanent name, use the NAME command.

MultiTSO will usually give the session the same name as the command you specify. However, sometimes an alias for that command is specified in the TSO's ECT (environment control table). Then MultiTSO will use the name in the ECT.

By default, SET DFLTNAME is on.

FORMAT

```
SET DFLTNAME { ON  }
             { OFF }
```

Issue SET DFLTNAME from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET DFLTNAME as a TSO command.

PARAMETER

ON|OFF Activates and deactivates default naming.

Parameter	Valid	Default
ON|OFF	ON or OFF	None

EXAMPLE

To turn off the default naming feature, enter SET DFLTNAME OFF.

SET HEADER

SET HEADER determines whether the MultiTSO menu header line—

```
..#.. Name....  Title.............................Attributes  SwChar:  \
```

—is displayed.

The default for SET HEADER is set with the HEADER start-up option.

FORMAT

```
SET HEADER { ON  }
           { OFF }
```

Issue SET HEADER from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET HEADER as a TSO command.

PARAMETER

ON|OFF Displays or turns off the header line.

Parameter	Valid	Default
ON|OFF	ON or OFF	ON

EXAMPLE

To display the header line, enter SET HEADER ON.

SET IMPTSO

SET IMPTSO ("implied TSO") determines what to do when you enter a non-PIE/TSO command on the MultiTSO menu, with PIEEXEC, or with the switch sequence. It also determines what to do with the PIE/TSO TSO command processors (e.g., ACCESS, PEXEC, PIESTART, etc.) when you enter them from the MultiTSO menu, with PIEEXEC, or with the switch sequence.

If you set SET IMPTSO to ON, MultiTSO will assume that the command is a TSO command. It will open a new session and execute the command as a TSO command in that session. Some users find this a convenient way to execute a command from a session. It avoids the steps required to open and switch to a session.

If you set SET IMPTSO to OFF, MultiTSO will issue a message stating that the command is invalid. Some users prefer this because typing errors don't cause unwanted processing.

The default for SET IMPTSO is set with the IMPTSO start-up option.

FORMAT

```
SET IMPTSO { ON  }
           { OFF }
```

Issue SET IMPTSO from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET IMPTSO as a TSO command.

PARAMETER

ON|OFF Enables or disables the implied TSO feature.

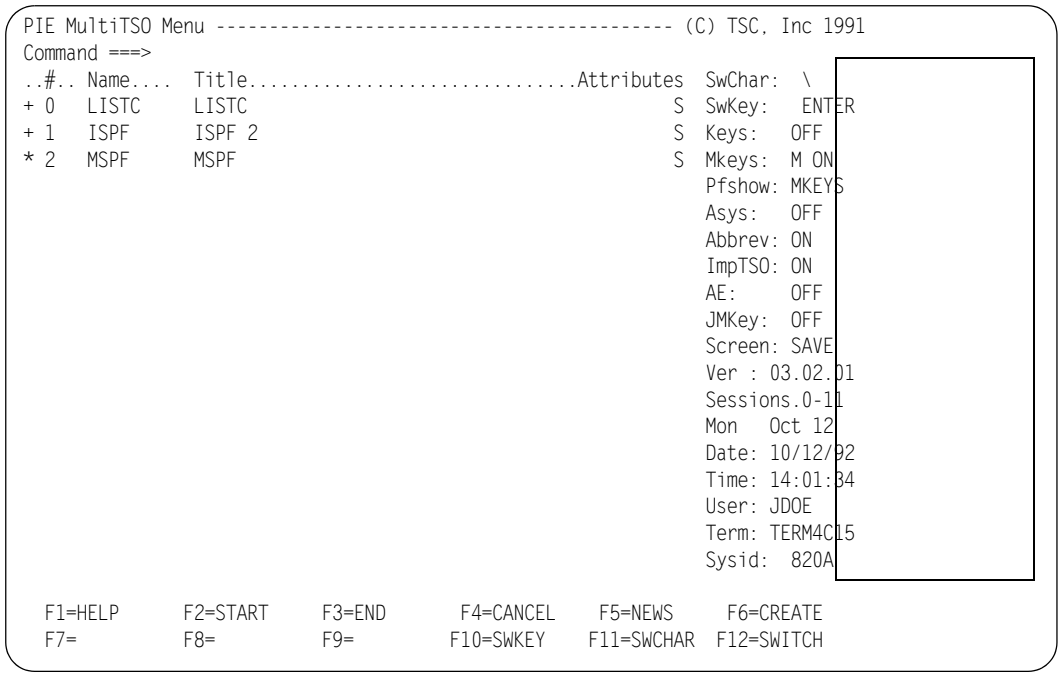
Parameter	Valid	Default
ON|OFF	ON or OFF	None

EXAMPLE

To enable the implied TSO feature, enter SET IMPTSO ON.

SET INFOWINDOW

SET INFOWINDOW controls the display of the information window on your MultiTSO menu. The information window contains information about your MultiTSO environment, such as your switch character and key, whether your PF keys are on or off, whether asynchronous switching is on or off, and so on.



The default for SET INFOWINDOW is set with the INFWIN start-up option.

FORMAT

```
SET INFOWINDOW { ON  }
               { OFF }
```

Issue SET INFOWINDOW from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET INFOWINDOW as a TSO command.

PARAMETER

ON|OFF Displays or removes the information window.

Parameter	Valid	Default
ON|OFF	ON or OFF	None

EXAMPLE

To turn off the information window display, enter SET INFOWINDOW OFF.

SET KEYS

SET KEYS activates MultiTSO function keys in sessions. When session keys are on, the keys in the SKEYS table will be activated. The SKEYS are the keys defined with the SET PFn and SET PA3 commands. When activated, session keys will override the normal key function for the application running in the session.

You can turn session keys on in all sessions, in a group of sessions, or in a single session. You can turn the session keys off in individual sessions so that you can use the PF keys defined in the applications.

If the jump key is on, the jump key will override the corresponding session key.

The default for SET KEYS is set with the KEYS start-up option. Use SET MKEYS to activate function keys on the MultiTSO menu.

FORMAT

```
SET KEYS { ON       } [ ids ]
         { OFF      } [ ALL ]
         { OPPOSITE }
```

Issue SET KEYS from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET KEYS as a TSO command.

PARAMETERS

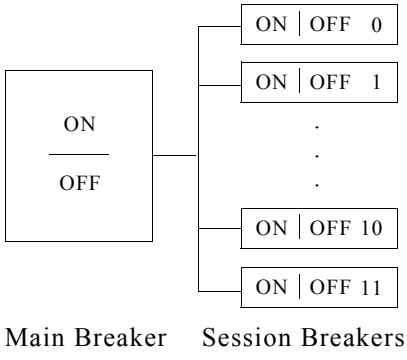
SET KEYS works on a breaker concept, with a main breaker and individual breakers for each session. For PF keys to work in a session, the main breaker and the session breaker must both be on. You can turn the main breaker on or off and leave the session breakers as they are. You can change session breakers without affecting the main breaker.

ON|OFF|OPPOSITE Turns a breaker on or off. You can assign OPPOSITE to a PF key and toggle session keys on and off.

ALL|ids Specifies what breakers to turn on or off.

Use ALL to change the settings for the main breaker and for all sessions (e.g., SET KEYS ON ALL).

If you want to change specific session breakers, specify their session IDs (e.g., SET KEYS ON 1). When you specify an ID, you change only the session breaker—you do not change the main



To change the main breaker only, omit both ids and ALL (e.g., SET KEYS ON).

Parameter	Valid	Default
ON|OFF|OPPOSITE	ON or OFF or OPPOSITE	None
ALL|ids	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None

EXAMPLE

You want to activate session keys in all sessions, you enter SET KEYS ON ALL.

Now you want to use session keys in session 8 but not session 7. You enter SET KEYS OFF 7. Now the session keys won't work in session 7, but they will continue to work in all other sessions.

Now you want to turn session keys off temporarily in all sessions, but you don't want to change your set up. (You want to leave session 7's keys off and all other session keys on.) You enter SET KEYS OFF. To turn them on again, you enter SET KEYS ON.

SET LOCKINPUT

SET LOCKINPUT determines what to do with unsaved data on a terminal screen that has been locked.

When your terminal has been locked automatically, no message is displayed until you press ENTER or a PF key. You might type data on the screen without knowing the terminal is locked. SET LOCKINPUT determines what to do with that data once you enter your password and unlock the terminal. MultiTSO can discard it or store it and return it to the screen.

Since the SET LOCKINPUT command affects AutoLock, an automatic function, and because it must be issued before the time-out to have effect, we recommend you put SET LOCKINPUT in your initial CLIST or REXX EXEC.

FORMAT

```
SET LOCKINPUT  {PROCESS}
               {DISCARD}
```

Issue SET LOCKINPUT from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET LOCKINPUT as a TSO command.

PARAMETER

PROCESS|DISCARD Use PROCESS to store the data and return it to the screen after a lock. Use DISCARD to discard it.

☞ If you specify PROCESS, you have some security exposure. If you walk away from your terminal, someone could type data on your screen. Then when you come back and enter the correct password, their data will be processed. You cannot specify PROCESS if your system administrator specified DISCARD in the LOCKINPUT start-up option.

Parameter	Valid	Default
PROCESS|DISCARD	PROCESS or DISCARD	None

EXAMPLE

You want to save the data. Before your terminal is locked, you issue the following command SET LOCKINPUT PROCESS from the MultiTSO menu.

You don't want others to be able to type data at your terminal when it is locked. You want to discard all unsaved data. Enter @SET LOCKINPUT DISCARD from your initial CLIST or REXX EXEC.

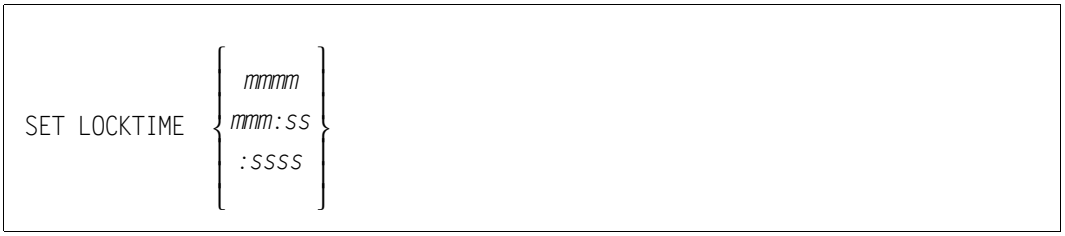
SET LOCKTIME

SET LOCKTIME controls the length of time your terminal must be inactive before it is automatically locked.

Your PIE/TSO administrator may set up AutoLock for everyone, for certain terminals, or for certain users. When a terminal is inactive (no one has pressed a PF key or ENTER) for a specific period of time, it locks.

With SET LOCKTIME, you can make this time period shorter than the default for your terminal or user ID. However you can't make it longer. For instance, if your default time-out interval is 10 minutes, you can change it to 5 minutes but not 15.

FORMAT



Issue SET LOCKTIME from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET LOCKTIME as a TSO command.

PARAMETER

mmmm|mmm:ss|:ssss Specify a time period in one of the three formats. If you omit this parameter, the LOCKTIME reverts to the global default set by the LOCK start-up option.

⚠ WARNING
If you set a LOCKTIME so short that you cannot type in a new LOCKTIME command, you will effectively lock yourself off this terminal.

If you do so inadvertently, enter incorrect passwords until you are logged off. Then log back on with a usable LOCKTIME.

Parameter	Valid	Default
mmmm	0 to 9999	None
mmm:ss	0:0 to 999:99	None
:ssss	:0 to :9999	None

EXAMPLE

To set the time-out to 5 minutes, enter SET LOCKTIME 5.

SET MENU

SET MENU controls the MultiTSO menu display.

You can use SET MENU OFF when you want to make MultiTSO transparent to users. This is useful, for instance, if you want to use only PIE/TSO's AutoLock feature. You can allow users only one session, activate that session automatically, and turn off the MultiTSO menu. AutoLock will work, but users will never know that PIE/TSO is on.

The default for SET MENU is set with the MENU start-up option.

FORMAT

```
SET MENU { ON  }
         { OFF }
```

Issue SET MENU from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET MENU as a TSO command.

PARAMETER

ON|OFF Specify whether you want your MultiTSO menu displayed or not.

Parameter	Valid	Default
ON|OFF	ON or OFF	None

EXAMPLES

To turn off the display of the MultiTSO menu, enter SET MENU OFF.

Now to turn it on again, enter @SET MENU ON from TSO READY.

SET MKEYS

SET MKEYS controls the MultiTSO menu keys.

MultiTSO maintains two tables of function keys. The MKEYS table defines keys that can work in the MultiTSO menu. The SKEYS table defines keys that can work in sessions or the MultiTSO menu or both.

If you like, you can have different function keys in your sessions and in the MultiTSO menu. To do so, set up both the MKEYS and SKEYS tables (using the SET PFn and SET MPFn commands). Activate the SKEYS table in your sessions (with SET KEYS ON). Activate the MKEYS table in the MultiTSO menu (with SET MKEYS MKEYS and SET MKEYS ON).

You can have the same set of function keys in both the MultiTSO menu and your sessions. To do so, you need only set up the SKEYS table. Then activate the SKEYS table in both your sessions (SET KEYS ON) and the MultiTSO menu (SET MKEYS SKEYS and SET MKEYS ON).

The default for SET MKEYS is set with the MKEYS start-up option.

FORMAT

Issue SET MKEYS from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET MKEYS as a TSO command.

PARAMETERS

ON|OFF|OPPOSITE Enables or disables use of the menu keys. To toggle the PF keys on and off, set OPPOSITE to a PF key.

MKEYS|SKEYS Determines which set of PF keys is active on the MultiTSO menu. MKEYS refers to the keys defined with the SET MPFn command. SKEYS refers to the keys defined with the SET PFn command.

Parameter	Valid	Default
ON|OFF|OPPOSITE	ON or OFF or OPPOSITE	None
MKEYS|SKEYS	MKEYS or SKEYS	None

EXAMPLES

To activate the SKEYS in the MultiTSO menu, enter SET MKEYS SKEYS and SET MKEYS ON.

To activate the MKEYS in the MultiTSO menu, enter SET MKEYS MKEYS and SET MKEYS ON.

SET MPA3

SET MPA3 assigns the function of the PA3 key in the MKEYS table (the MultiTSO menu function keys). You can set PA3 to any function, application, or session ID. If you set it to a session ID, pressing PA3 will switch you to that session.

FORMAT

```
SET MPA3 command
```

Issue SET MPA3 from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET MPA3 as a TSO command.

PARAMETER

command Specify an application, function, or session. Code the command as you would to execute it from the MultiTSO menu.

Parameter	Valid	Default
command	1 to 57 characters	None

EXAMPLE

To set the PA3 command to switch to the next active session, enter SET MPA3 SWITCH NEXT ACTIVE from the MultiTSO menu.

SET MPA3L

SET MPA3L assigns a label to the MKEYS table (MultiTSO menu) PA3 key. When the PA keys are displayed at the bottom of the MultiTSO menu, this label will be displayed with PA3.

The default label for a PF key is the first word in its command. For instance, if you enter SET MPA3 CREATE, the label will default to CREATE.

FORMAT

```
SET MPA3L label
```

Issue SET MPA3L from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET MPA3L as a TSO command.

PARAMETER

label Specify the PA3 label.

Parameter	Valid	Default
label	1 to 8 characters	None

EXAMPLE

If PA3 is SWITCH NEXT ACTIVE, label PA3 with SET MPA3L SW NEXT.

SET MPFn

SET MPFn defines the MKEYS table (MultiTSO menu) PF keys. You may assign a command, application, or session to the key.

FORMAT

```
SET MPFn command
```

Issue SET MPFn from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET MPFn as a TSO command.

PARAMETERS

n Specify the PF key to define.

command Specify the application, function, or session. Code the command as you would to execute it from the MultiTSO menu.

Parameter	Valid	Default
n	1 to 24	None
command	1 to 57 characters	None

EXAMPLE

To switch to the MSPF session with PF2, enter SET MPF2 SWITCH MSPF.

SET MPFnL

SET MPFnL assigns labels to MKEYS PF keys (MultiTSO menu PF keys). When the MKEYS PF keys are displayed at the bottom of the MultiTSO menu, these labels will be displayed.

The default label for a PF key is the first word in its command. For instance, if you enter SET MPF2 CREATE to define the PF key, the label will default to CREATE.

FORMAT

```
SET MPFnL label
```

Issue SET MPFnL from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET MPFnL as a TSO command.

PARAMETERS

n Specify the PF key number.

label Specify the label.

Parameter	Valid	Default
n	1 to 24	None
label	1 to 8 characters	None

EXAMPLE

If PF2 starts an ISPF session, you could label it with SET MPF2L NEW ISPF.

SET MTASK

SET MTASK turns MultiTask on or off for a particular session.

You may turn MultiTask on in only one session at a time.

FORMAT

```
SET MTASK { ON  } id
          { OFF }
```

Issue SET MTASK from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET MTASK as a TSO command. You may also execute SET MTASK by placing the cursor in the command column next to the session, typing M, and pressing ENTER.

PARAMETERS

ON|OFF Activates or deactivates MultiTask for the session identified.

id Specify the code or name of the session you want to become your MultiTask background session.

Parameter	Valid	Default
ON|OFF	ON or OFF	None
id	0 to 11, A, B, ?, *, or 1 to 8 character name	None

EXAMPLES

To turn MultiTask on in session 0, enter SET MTASK ON 0.

To turn MultiTask on in session 0, enter @SET MTASK ON 0 from TSO READY.

SET PA3

SET PA3 assigns a function, application, or session to a session PA3 key (SKEYS table).

FORMAT

```
SET PA3 command
```

Issue SET PA3 from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET PA3 as a TSO command.

PARAMETER

command Specify the application, function, or session. Code the command as you would to execute it from the MultiTSO menu.

Parameter	Valid	Default
command	1 to 57 characters	None

EXAMPLE

To assign SWITCH BACK to PA3, enter SET PA3 SWITCH BACK.

SET PA3L

SET PA3L assigns a label to the session PA3 key (SKEYS table). When the MultiTSO menu displays the SKEYS, PA range, this label will be displayed next to PA3.

The default label for a PF key is the first word in its command. For instance, if you enter SET PA3 CREATE, the label will default to CREATE.

FORMAT

```
SET PA3L label
```

Issue SET PA3L from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET PA3L as a TSO command.

PARAMETER

label Specify the PA3 key label.

Parameter	Valid	Default
label	1 to 8 characters	None

EXAMPLE

To set the PA3 label to SW BACK, enter SET PA3L SW BACK.

SET PARSER

SET PARSER determines which MultiTSO parser (release 3.2.1 or 2.5.3) will be in effect. The 3.2.1—and 3.1.0—parser allows you to identify sessions with more than one character. This means you can name sessions, and you can identify them with the numbers 0 through 11. It requires you to use spaces between session IDs—whether names or session numbers—when you have more than one in a command. For example, to cancel sessions 1, 2, and 3, you must enter CANCEL 1 2 3.

The 2.5.3 parser limits session IDs to one character. This means that you can't give a session a name with the NAME command and that you must identify sessions 10 and 11 with the characters A and B. It allows you to identify multiple sessions without using spaces (for example, CANCEL 123).

To check which parser you are now using, see the Sessions field in the MultiTSO menu information window. If you have all sessions available and you are using the old parser, it will show 0-B. If you are using the new parser, it will show 0-11.

FORMAT

```
SET PARSER {OLD}
           {NEW}
```

Issue SET PARSER from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET PARSER as a TSO command.

PARAMETER

OLD|NEW Specify OLD to use the release 2.5.3 parser. Specify NEW to use the 3.2.1 parser.

Parameter	Valid	Default
OLD|NEW	OLD or NEW	None

EXAMPLE

To go back to the 2.5.3 parser, enter SET PARSER OLD.

SET PASSWORD

SET PASSWORD defines your password for ACCESS command variables. When you define your password, Network Access can pass both your user ID and your password to the APPLID you are accessing and sign you on automatically.

You can only pass passwords to APPLIDs using the PIE Logon Director.

FORMAT

```
SET PASSWORD password
```

Issue SET PASSWORD from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET PASSWORD as a TSO command.

PARAMETER

password Specify your password.

Parameter	Valid	Default
password	1 to 8 characters	None

EXAMPLE

First, set the password variable to APPLES: SET PASSWORD APPLES

Next, access CICSTEST with the &ZUSER and &ZPSWD (or &ZSNON) variables: ACCESS CICSTEST &ZUSER &ZPSWD

Network Access will pass the user ID you logged on to TSO with and the APPLES password to CICSTEST automatically.

SET PFn

SET PFn defines SKEYS table PF keys (sessions PF keys). You may assign a command, application, or session to each key.

FORMAT

```
SET PFn command
```

Issue SET PFn from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET PFn as a TSO command.

PARAMETERS

n Specify the PF key to define.

command Specify the application, function, or session. Code the command as you would to execute it from the MultiTSO menu.

Parameter	Valid	Default
n	1 to 24	None
command	1 to 57 characters	None

EXAMPLE

To switch to the session named EDIT, enter SET PF2 EDIT.

SET PFnL

SET PFnL assigns labels to sessions PF keys (SKEYS table). When the SKEYS are displayed at the bottom of the MultiTSO menu, these labels will be displayed.

The default label for a PF key is the first word in its command. For instance, if you enter SET PF2 CREATE to define the PF key, the label will default to CREATE.

FORMAT

```
SET PFnL label
```

Issue SET PFnL from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET PFnL as a TSO command.

PARAMETERS

n Specify the PF key number.

label Specify the label.

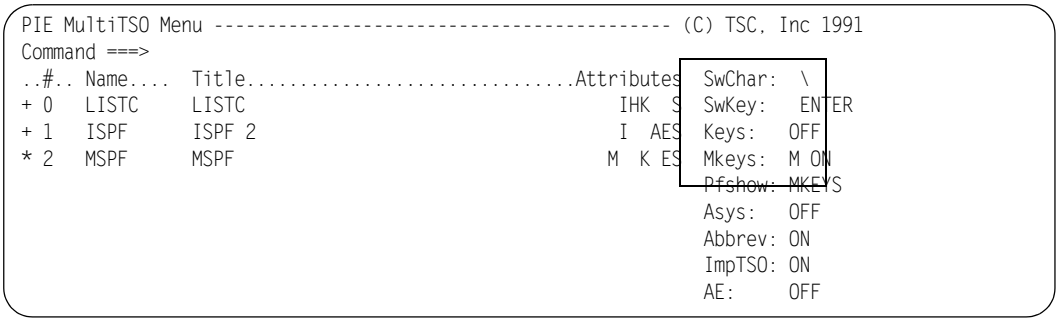
Parameter	Valid	Default
n	1 to 24	None
label	1 to 8 characters	None

EXAMPLE

PF2 switches you to the EDIT session. Label it with SET PF2L EDIT.

SET SESATT

SET SESATT controls the session attribute display on the MultiTSO menu.



The default for SET SESATT is set with the SESATT start-up option.

FORMAT

```
SET SESATT { ON  }
           { OFF }
```

Issue SET SESATT from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET SESATT as a TSO command.

PARAMETER

ON|OFF Activates or deactivates the display.

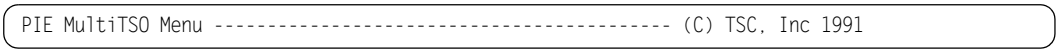
Parameter	Valid	Default
ON|OFF	ON or OFF	None

EXAMPLE

To turn on the session attribute display, enter SET SESATT ON from the MultiTSO menu.

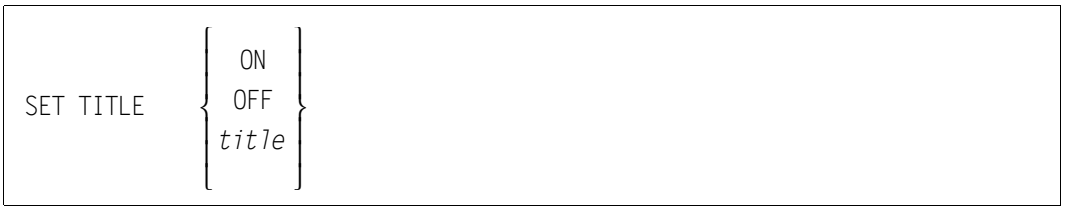
SET TITLE

SET TITLE determines whether the MultiTSO menu title line—



—is displayed and determines its content.

FORMAT



Issue SET TITLE from the MultiTSO menu, with PIEEXEC, or with the switch sequence. You may issue @SET TITLE as a TSO command.

PARAMETER

ON|OFF Activates or deactivates the title line.

title Specify the title line text.

Parameter	Valid	Default
ON|OFF	ON or OFF	None
title	1 to 100 characters	None

EXAMPLES

To turn on the title line, enter SET TITLE ON from the MultiTSO menu.

START and STARTH

START opens a session and switches to it. STARTH opens a hidden session.

FORMAT

```
{ Start  }
{ STARTH }  [id] [command]
```

There are many ways to issue START. You can issue it from the MultiTSO menu or as a TSO command. If you are in the MultiTSO menu, you can issue START by typing S or by pressing PF2 or PF14. START is the default command from the MultiTSO menu. If you type a session ID and press ENTER, MultiTSO will execute START for that ID.

Issue STARTH from the MultiTSO menu or as a TSO command.

PARAMETERS

id Specify the code of the session to open. If you omit it, MultiTSO will open the next session that has not yet been activated. If you specify a session that is already open, MultiTSO will switch you to the session. If you want to switch to a session that is already open, you may specify the session name for id.

command Specify a command and its parameters if you want to execute the command automatically when MultiTSO opens the session. If the session is already active, MultiTSO will ignore this parameter.

☞ If auto-end is on for this session and the application you start with the command parameter fails or you omit the command parameter entirely, auto-end will end your session automatically. You will be returned to the MultiTSO menu without a message. If this occurs, set auto-end off for the session.

Parameter	Valid	Default
id	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	Next available session
command	1 to 256 character TSO command	None

EXAMPLES

To display the time of day on the next unused session, enter START TIME.

If you are in ISPF and you want to open a new session and run the LISTA command in it, enter TSO START LISTA.

START is the default command for the MultiTSO menu. So to open session 6, enter 6 from the MultiTSO menu.

To begin session 4 as a hidden session and run ISPF edit, enter STARTH 4 ISPF 2.

STOP

STOP terminates a session from TSO.

STOP is meant to be used from CLISTs and REXX EXECs. If MultiTSO is not active, it doesn't terminate your TSO user ID. Instead STOP terminates with return code 8. If MultiTSO is active, it will execute the TSO LOGOFF command to terminate the current session.

FORMAT

```
STOP
```

Issue STOP as a TSO command only.

EXAMPLE

To terminate the current session and return to the MultiTSO menu, enter STOP.

SWCHAR

SWCHAR changes the switch character.

Your MultiTSO administrator sets the default switch key for your installation. It can be any non-alphanumeric character on the standard 3270 keyboard. That is, =, !, @, #, and so on. If you do not like your default switch character, you can change it with the SWCHAR command. Your change will remain in effect until you log off TSO or until you change it again.

⚠ WARNING
If you set your switch key to ENTER or PF3, do not set your switch character to =, /, ., &, or *. These combinations can lead to unexpected and unwanted results in certain applications.

FORMAT

```
SwChar character
```

Issue SWCHAR from the MultiTSO menu or as a TSO command. From the MultiTSO menu, you can execute it with PF11 or PF23.

PARAMETER

character Specify a non-alphanumeric character. The distributed default is backslash (\).

Parameter	Valid	Default
character	1 non-alphanumeric character	None

EXAMPLES

To change the switch character to a "{", enter SC {.

SWITCH

SWITCH switches between the current session and the MultiTSO menu. It also switches to a specified session.

FORMAT

```
SWITCH [id]
```

Issue SWITCH from the MultiTSO menu or as a TSO command. If you are on the MultiTSO menu, you can switch to a session by selecting it with the cursor, typing S, and pressing ENTER. (You can also switch to a session from the MultiTSO menu by typing its ID on the command line and pressing ENTER.)

PARAMETER

id Specify the session code or name. If you omit it and you are in the MultiTSO menu, you will be switched to the current session. If you omit it from a session, you will be switched to the MultiTSO menu.

Parameter	Valid	Default
id	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None

EXAMPLES

To get to the MultiTSO menu, enter SWITCH from TSO READY.

To get to session 5, enter SWITCH 5.

To switch to the session named CICS, enter SWITCH CICS.

SWITCH BACK

SWITCH BACK switches between the last two sessions you have accessed. It is especially helpful when you assign it to a PF key.

FORMAT

```
SWITCH BACK
```

Issue SWITCH BACK from the MultiTSO menu or as a TSO command.

EXAMPLE

To switch back to the last session you accessed, enter SWITCH BACK.

SWITCH NEXT

SWITCH NEXT rotates through MultiTSO sessions in numerical order. When SWITCH NEXT gets to the highest session, it switches to the lowest.

If there is only one session, SWITCH NEXT is ignored.

SWITCH NEXT is especially helpful when you assign it to a PF key.

FORMAT

```
SWITCH NEXT  [ ACTIVE   ]
             [ UNHIDDEN ]
```

Issue SWITCH NEXT from the MultiTSO menu or as a TSO command.

PARAMETER

ACTIVE|UNHIDDEN Specify ACTIVE to switch forward to only active sessions. Specify UNHIDDEN to skip hidden sessions. To switch to all sessions, omit both.

Parameter	Valid	Default
ACTIVE|UNHIDDEN	ACTIVE or UNHIDDEN	All sessions

EXAMPLES

To switch forward to the next highest session, enter SWITCH NEXT.

To switch forward but skip hidden sessions, enter SWITCH NEXT UNHIDDEN.

SWKEY

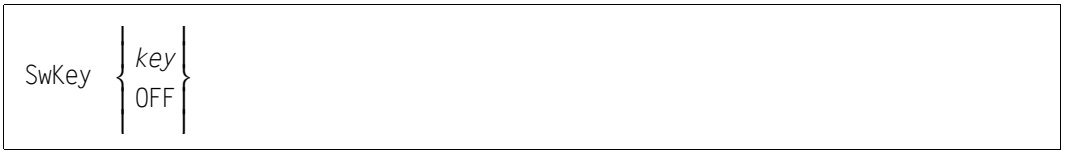
SWKEY sets your switch key.

Your MultiTSO administrator sets the default switch key for your installation. It can be ENTER or any PF key. If you do not like your default switch key, you can change it with the SWKEY command. Your change will remain in effect until you log off TSO or until you change it again.

☞ If you set the switch key to a PF key, it is not doubled into the other set of PF keys. So, for example, PF12 means only PF12, not PF12 and PF24.

⚠ WARNING
If you set your switch character to =, /, ., &, or *, do not set your switch key to ENTER or PF3. These combinations can bring unexpected and unwanted results in certain applications.

FORMAT



Issue SWKEY from the MultiTSO menu or as a TSO command. From the MultiTSO menu, you can issue it from PF10 or PF22.

PARAMETER

key Specify ENTER or any PF key number, in either PF1 or 1 format.

OFF Use OFF to turn the SWKEY off. This can be useful if you want to use the PIE/TSO AutoLock feature only, and you want all other PIE/TSO features to be invisible to users.

Parameter	Valid	Default
key	ENTER or 1 to 24	None
OFF	OFF	None

EXAMPLE

To change your switch key to PF4, enter SK 4.

UNHIDE

UNHIDE unhides sessions so that the jump key can switch to them. Use it if you have opened a hidden session with ALLOCH, CREATEH, or STARTH, or you have hidden a session with the HIDE command.

FORMAT

```
UNhide { ids }
       { ALL }
```

Issue UNHIDE from the MultiTSO menu or as a TSO command.

PARAMETER

ids|ALL Specify the IDs of the sessions you want to unhide. To unhide them all, specify ALL.

Parameter	Valid	Default
ids|ALL	0 to 11, A, B, H, I, V, *, ?, or 1 to 8 character name	None

EXAMPLE

To unhide sessions 3, 4, and 11, enter UN 3 4 11 from the MultiTSO menu.

V

V starts the View session and captures the image of the screen you execute it from. (When you issue the V command from the MultiTSO menu, it only starts View. It doesn't capture the menu.)

The View session is a special session that does not count towards your maximum number of sessions. Its session ID is V. For more information on the View session, see Chapter 4, “Special Sessions,” in the PIE/TSO User Guide.

FORMAT

```
V
```

Issue V from the MultiTSO menu or with the switch sequence.

EXAMPLES

To start View, enter V from the MultiTSO menu.

To capture a screen image, access the screen you want to capture (in any PIE/TSO session) and issue \V [ENTER]. (Substitute your own switch sequence.)

Chapter 3
View Commands

View provides several commands for handling screen images—editing them, saving them, sending them to other users, etc. This chapter is a reference for these commands.

CONTENTS

Using View Commands

How

You can enter View commands:

- by typing the command on the View menu command line
- by pressing PF keys in View

If you enter a command using a PF key and you want to enter any parameters, type the parameters on the command line before you press the PF key.

For instance, to execute EDIT XYZ, you could issue XYZ [PF6] on the View menu command line.

You may type in upper or lower case or a mixture of both. Separate parameters with blanks.

Where

You must be in the View session to use a View command. If you are not in View, the application you are currently in will try to execute the command as its own command.

Some commands only work in browse mode. Each command description indicates whether the command can be issued from browse, edit, or either mode.

View commands affect the screen images you are editing or browsing. If the command allows you to specify a member name, it will affect the member you specify.

View Commands

View commands are listed in alphabetical order. Each command description indicates the PF keys assigned to the command.

DOWN

DOWN scrolls down through the data being displayed. Use it on the View menu or a screen image.

By default DOWN scrolls down the number of data lines on the screen, minus one. This way, the bottom line on the screen before you issued DOWN becomes the top line afterward.

You can temporarily modify the amount scrolled using the HALF or MAXIMUM parameters.

FORMAT

```
DOWN [Half|Maximum]
```

Issue DOWN in browse mode only. It is assigned to PF8 and PF20.

HALF Scrolls only 12 data lines.

MAXIMUM Scrolls to the end of the member or menu.

Parameter	Valid	Default
HALF|MAXIMUM	HALF, H, MAXIMUM, or M	None

EDIT

As you would expect, EDIT puts you in edit mode. It displays a screen image and allows you to modify it and to save it. You can use EDIT to access a newly captured image or an existing member.

FORMAT

```
EDIT [member]
```

Issue EDIT in browse mode only. It is assigned to PF6 and PF18.

member To display a saved screen image, specify a View dataset member name or point to the name with the cursor and press PF6. To display the last screen image you captured, omit the member name.

Parameter	Valid	Default
member	View dataset member name	Home buffer

END

If you issue END from the View menu, it will close the View session.

If you issue END in a browse, it will return you to the View menu. It will not save the screen image.

If you issue END from edit mode or after a RECEIVE, it will prompt you for a member name to save it. If you don't enter a member name, END will return you to the View menu without saving the image.

FORMAT

```
END
```

Issue END in any mode. It is assigned to PF3 and PF15.

HELP

HELP in View is the same as MultiTSO HELP. It opens and switches to the HELP session. If you supply a HELP dataset member name, it will open that member automatically.

FORMAT

```
HELP [member]
```

Issue HELP in any mode. It is assigned to PF1 and PF13.

member Specify a HELP dataset member name to view that member immediately. If you omit the member name, you will see the View HELP member.

Parameter	Valid	Default
member	Any HELP member	VIEW

NEXT

NEXT highlights the next screen image in the list of members on the View menu. If you are already at the bottom of the list, NEXT highlights the screen image at the top of the list.

FORMAT

```
NEXT
```

Issue NEXT in browse mode only. It is assigned to PF11 and PF23.

PREV

PREV highlights the previous screen image in the list of members on the View menu. If you are already at the top of the list, PREV highlights the screen image at the bottom of the list.

FORMAT

```
PREV
```

Issue PREV in browse mode only. It is assigned to PF10 and PF22.

RECALL

RECALL redisplays the last command you entered on the View menu command line. You may then change it and reissue it.

FORMAT

```
RECALL
```

Issue RECALL in browse mode only. It is not assigned to any PF keys.

RECEIVE

RECEIVE displays a screen image that has been sent to you by another user. You may then edit it and save it.

After you receive a screen image, you can modify it and issue RECEIVE again to re-display it with your modifications.

Use RECEIVE after you see the message:

```
A PIE/View screen has been sent by 'userid'
```

Since there can be only one screen image at a time in the inter-user buffer, use RECEIVE promptly to avoid having the screen image overlaid by another screen image sent to you.

If the user who sent the image has a larger screen size than you do, you'll get an error message when you receive it. You can ask the other user to change his or her screen size, or you can view his or her View dataset member in ISPF Browse.

FORMAT

```
RECEIVE
```

Issue RECEIVE in browse mode only. It is assigned to PF9 and PF21.

RETURN

RETURN cancels the current command and returns you to the View menu. If you are already in the View menu, it closes the View session.

If you are in a screen image and you issue RETURN, any modifications you have made will be lost.

FORMAT

```
RETURN
```

Issue RETURN in any mode. It is assigned to PF4 and PF16.

SEND

SEND sends the last screen image you captured to the user you specify. That user may then issue RECEIVE to access it.

SEND places this image in the inter-user buffer. It will overlay any screen image already in this buffer. So it is wise to RECEIVE and save any screen image you have been sent before you issue SEND.

FORMAT

```
SEND userid
```

Issue SEND in browse mode only. It is assigned to PF5 and PF17.

userid Specify the ID of the user you want to send the screen image to. The user must be using View.

Parameter	Valid	Default
userid	TSO user ID	None

SHOW

SHOW is an alias for EDIT. See “EDIT” for more information.

UP

UP scrolls up through the data being displayed. Use it on the View menu or when browsing a screen image.

By default UP scrolls up the number of data lines on the screen, minus one. This way, the bottom line on the screen before you issued UP becomes the top line afterward.

You can temporarily modify the amount scrolled using the HALF or MAXIMUM parameters.

FORMAT

```
UP [Half|Maximum]
```

Issue UP in browse mode only. It is assigned to PF7 and PF19.

HALF Scrolls only 12 data lines.

MAXIMUM Scrolls to the beginning of the member or View menu.

Parameter	Valid	Default
HALF|MAXIMUM	HALF, H, MAXIMUM, or M	None

Index

Symbols

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Executing Commands

PIEEXEC Prefix used to enter PIE/TSO commands from the TSO READY prompt and from applications that accept TSO commands.

```
PIEEXEC command
```

RECALL* Displays the last command entered from the MultiTSO menu.

```
Recall
```

Your MultiTSO Environment

SCREEN Saves data typed on a screen but not yet entered when you switch out of a session. Also displays the MultiTSO menu header line.

```
SCREEN {SAVE|NOSAVE|ON|OFF}
```

LOCK Prevents unauthorized use of your terminal while you are not using it.

```
LOCK msgtext
```

SET LOCKINPUT+ Determines whether data typed on your screen but not entered before the terminal was locked will be passed back to the screen after the terminal is unlocked.

```
SET LOCKINPUT {PROCESS|DISCARD}
```

SET LOCKTIME+ Specifies the length of time your terminal must be inactive before it is AutoLocked.

```
SET LOCKTIME {mmmm|mmm:ss|:ssss}
```

Controlling the MultiTSO Menu

SET INFOWINDOW+ Controls the MultiTSO menu information window.

```
SET INFOWINDOW {ON|OFF}
```

SET SESATT+ Controls the session attribute display.

```
SET SESATT {ON|OFF}
```

SET TITLE+ Controls the title line.

```
SET TITLE {ON|OFF|title}
```

SET HEADER+ Controls the header line.

```
SET HEADER {ON|OFF}
```

RESHOW* Refreshes the MultiTSO menu.

```
RESHOW
```

SET MENU+ Controls the MultiTSO menu display.

```
SET MENU {ON|OFF}
```

Miscellaneous

MSPF and MULTISPF Allow you to run copies of ISPF in multiple sessions at once.

```
{MSPF|MULTISPF} [parms]
```

FENCE* Reserves storage space from applications that take all available storage in your region so you can open more sessions later.

```
FENCE {ON|value}
```

PEXEC Signals PGM to apply the PGN and SWAP and NOSWAP values assigned to the application.

```
PEXEC [CMD|PGM] command [parms]
```

PIECHECK Checks current PIE/TSO status.

```
PIECHECK [code]
```

DISPLAY* Displays environment information.

```
Display [SYStem|id|STOrage]
```

MultiTask

SET MTASK+ Turns MultiTask on and off.

```
SET MTASK {ON|OFF} id
```

Network Access

ACCESS Invokes Network Access.

```
ACCESS [applid] [data|variables]
```

SET AATTN+ Controls the attention key action in Network Access sessions.

```
SET AATTN {SWITCH|ATTN}
```

SET APFn+ Sets the attention key function to a PF key in Network Access sessions.

```
SET APFn [ATTN]
```

SET PASSWORD+ Sets your password for ACCESS variables.

```
SET PASSWORD password
```

Special Sessions

H* Opens the MultiTSO HELP session.

```
H
```

I* Opens the INFO session.

```
I
```

V* Opens View and captures screen images.

```
V
```

* To execute at TSO READY, issue with PIEEXEC or the switch sequence.

+ To execute at TSO READY, prefix with @ or issue with PIEEXEC or the switch sequence.

PIE/TSO

Command Reference Card

Release 3.2.1

UNICOM SYSTEMS, INC.

Opening Sessions

START Opens a session and switches to it.
`Start [id] [command]`

STARTH Starts a hidden session.
`STARTH} [id] [command]`

ALLOC+ Opens a session with the allocated attribute. When you end the session, it remains on the MultiTSO menu.

ALLOCH* Allocates a hidden session.
`{ALLOC|ALLOCH} [code] [command]`

CREATE Opens a session.
`CREATE [code] [command]`

CREATEH Creates a hidden session.
`CREATEH [code] [command]`

SET CMDDEFLT+ Sets a default TSO command to execute when you open a session.
`SET CMDDEFLT command`

SET IMPTSO+ Determines whether MultiTSO opens a new session for non-PIE/TSO commands entered from the MultiTSO menu, with PIEEXEC, or with the switch sequence.
`SET IMPTSO {ON|OFF}`

PIESTART Starts and names the session specified. If the session already exists, PIESTART switches to it.
`PIESTART //name/command`

Ending Sessions and Logging Off

LOGOFF Logs you off a session. If no other sessions are active, it also logs you off TSO.
`LOGOFF`

CANCEL* Abends the application in a session and ends the session.

@CAN Alias for CANCEL.
`CANcel {ids|ALL|PIE}`

CLOSE* Cancels a session but leaves the session on the MultiTSO menu.
`CLOSE {ids|ALL}`

DISC* Disconnects your TSO logon.
`DISC`

KEEP* Assigns the allocated attribute to a session.
`KEEP {ids|ALL}`

FREE* Removes the allocated attribute.
`FREE {ids|ALL}`

SET AE+ Sets the auto-end attribute. The session ends when the application running in it ends.
`SET AE {ON|OFF} {ids|ALL}`

SET CMDFINAL+ Sets a default final TSO command that executes when you log off all sessions.
`SET CMDFINAL command`

RESTART* Cancels and reopens a session.
`REStart id [command]`

LOGON* Cancels all your sessions and logs you off and back on to TSO.
`LOGON [userid] [parms]`

STOP Ends a session from a CLIST or REXX EXEC.
`STOP`

Switching Sessions

SWITCH Switches to a MultiTSO menu or session.
`SWITCH [id]`

SWITCH BACK* Switches between the last two sessions accessed.
`SWITCH BACK`

SWITCH NEXT* Switches forward through sessions in numerical order.
`SWITCH NEXT [ACTIVE|UNHIDDEN]`

SWCHAR Sets the switch character.
`SwChar character`

SWKEY Sets the switch key.
`SwKey {key|OFF}`

JMKEY Sets the jump key.
`JMKey key`

HIDE Hides a session by removing it from the jump key cycle.
`HIde {ids|ALL}`

UNHIDE Unhides a session and returns it to the jump key cycle.
`UNhide {ids|ALL}`

END* Pushes the current session to the bottom of the stack and switches to the last session used.
`END`

ENDBACK* Pushes the current session to the bottom of the stack and switches to the session you accessed before the current session.
`ENDBACK`

SET ASYS+ Enables asynchronous switching.
`SET ASYS {ON|OFF|OPPOSITE} [ids|ALL]I`

Identifying Sessions

NAME* Assigns a name to a session.
`NAME id [name]`

SET DFLTNAME+ Specifies if PIE/TSO can assign names to sessions automatically.
`SET DFLTNAME {ON|OFF}`

NOTE* Modifies a session's comment.
`NOTE id [comments]`

SET ABBREV+ Specifies the use of abbreviations for session names.
`SET ABBREV {ON|OFF}`

SET PARSER+ Authorizes session names.
`SET PARSER {OLD|NEW}`

Controlling Function Keys

PFSHOW* Displays PF key assignments on the MultiTSO menu.
`PFSHOW [ON|OFF|type|range]`

SET MKEYS+ Activates MultiTSO menu PF keys.
`SET MKEYS {ON|OFF|OPPOSITE|MKEYS|SKEYS}`

SET MPFn+ Sets MultiTSO menu PF keys.
`SET MPFn command`

SET MPFnL+ Assigns labels to MultiTSO menu PF keys.
`SET MPFnL label`

SET MPA3+ Sets the PA3 key on the MultiTSO menu.
`SET MPA3 command`

SET MPA3L+ Assigns a label to the MultiTSO menu PA3 key.
`SET MPA3L label`

SET KEYS+ Activates session PF keys.
`SET KEYS {ON|OFF|OPPOSITE} [ids|ALL]`

SET PFn+ Sets session PF keys.
`SET PFn command`

SET PFnL+ Assigns labels to session PF keys.
`SET PFnL label`

SET PA3+ Sets the PA3 key for use in a session.
`SET PA3 command`

SET PA3L+ Assigns a label to the session PA3 key.
`SET PA3L label`

PIE/TSO Administrator Guide

Release 3.2.1

UNICOM
SYSTEMS, INC.

PTSOIG321-01

Printed: October 2002

This manual applies to PIE/TSO release 3.2.1 and to all subsequent releases of the product until otherwise indicated by new editions or updates to this publication.

UNICOM Systems, Inc.
Unicom Plaza - Suite 310
15535 San Fernando Mission Blvd.
Mission Hills, California 91345

Contents

Preface

The PIE/TSO Administrator Guide provides a quick tutorial on PIE/TSO features and information on PIE/TSO maintenance, operations, and advanced commands.

Chapter 1 briefly introduces you to all PIE/TSO components.

Chapter 2 is a quick guide to PIE/TSO.

Chapter 3 introduces you to advanced commands and the commands of particular importance to administrators.

Chapter 4 details PIE/TSO maintenance and operations.

For installation information, see the PIE/TSO Installation and Customization Guide.

This guide is not a complete guide to PIE/TSO features. Instead, it presents topics of special interest to administrators and advanced users. For other user-oriented information, see the PIE/TSO User Guide and the PIE/TSO Command Reference.

Who is this manual for?

This manual is written for PIE/TSO administrators, system programmers, and other advanced PIE/TSO users. We assume that you know how to use TSO and ISPF.

Customer Services

Troubleshooting suggestions

We have found that many problems can be solved by looking into a few common problem areas. We suggest you look into the following.

- Have you made any recent changes to your installation, including changes to MVS, TSO, and other products? Are these changes compatible with PIE/TSO?
- Is the current release of PIE/TSO installed?
- Is your PIE/TSO password current and correctly entered?
- Are there any relevant messages on your PIE/TSO or TSO logs?

Contacting Customer Services

WWW.UNICOMSI.COM/SUPPORT

UNICOM Systems' normal business hours are from 7:00 a.m. to 4:00 p.m. Pacific Time, Monday through Friday. However, emergency customer service is available 24 hours a day, 7 days a week.

When you call Customer Services, please make sure you have the following information available:

- related error messages
- the command or JCL you are inputting
- the dump, if one is generated

CUSTOMER SERVICES PHONE

Our phone number is (818) 838-0606.

If you call after hours our answering service will take the call. You may leave a message to be returned during normal working hours. Or you may leave an urgent message, and a technical support representative will be paged to call you back immediately. When leaving an urgent message, be sure to leave your after-hours phone number.

If you've left an urgent message but haven't received a call back within an hour, please call back. We may be having trouble reaching you.

+ International customers, please contact your distributor for technical assistance.

CUSTOMER SERVICES FAX

Our FAX number is (818) 838-0776.

Chapter 1 Introduction

MultiTSO allows you to operate up to 12 simultaneous TSO sessions on a single terminal with a single TSO user ID. MultiTSO supports everything TSO supports. You can run SDSF, RMFMON, SAS, FOCUS or any other TSO or ISPF command in a MultiTSO session. If you run ISPF in any or all of your sessions, you can use its split-screen capabilities. So it is possible to run twenty-four different commands at once, on the same terminal, with a single TSO ID.

You can switch between sessions at any time. You never have to log off one function to move to another. You simply enter one of several PIE/TSO switching commands. When you return to a session, it will be just as you left it.

What is MultiTask?

With MultiTSO, you can switch out of any session—even a session processing a task—and work in another session. However with MultiTSO alone, processing can only occur in the session you see on the screen. Once you switch out of a session, it stops processing until you return to it.

MultiTask allows TSO to process a task in the background while you work in the foreground. As long as the foreground session is not active—that is, you have not pressed ENTER or a function key within five seconds—TSO will continue to process the task in the background.

So you can view output, type data, write code, consider how to respond to data on the screen, and so on, while the task in the background runs undisturbed. If you press ENTER or a function key, MultiTask immediately returns control to the foreground session, so that you don't have to wait. Then MultiTask waits five seconds, and if you haven't pressed ENTER or a function key again, it returns control back to the background session which continues to process as before.

MultiTask is highly efficient in its switching technology and should not produce any measurable overhead for most installations.

What is Network Access?

Network Access allows you to run VTAM applications in your MultiTSO sessions. You can access any TSO, CICS, IMS, PROFS, etc., system on any CPU in your network.

What is View?

View captures and saves screen images from any VTAM application such as TSO, CICS, IMS, NCCF, PROFS, etc. You can browse or change these images, and send them to other View users.

What is Performance Group Manager?

Performance Group Manager (PGM) allows you to assign performance groups and domains by application ID, rather than by account code or user ID, as with IBM's SRM. If you have special commands that you want to have higher or lower priority during a user's logon, you can assign special performance group and SWAP/NOSWAP values to the application IDs. (See the *PIE/TSO Installation and Customization Guide* for details on setting up an application for PGM processing.)

PGM automatically assigns performance group numbers to applications when they are executed with the PIE/TSO PEXEC command or as aliases to PEXEC. When users switch between applications, PGM switches the performance group.

Chapter 2 Using PIE/TSO

In this chapter, you'll learn the basic elements of PIE/TSO. You'll learn how to log on, how to open, control, and end your sessions, and how to switch back and forth between sessions. You'll learn to use MultiTSO, MultiTask, View, and Network Access. Our examples assume that you have not changed the distributed defaults.

This tutorial progresses at a faster pace than the *User Guide* tutorial and contains information on special circumstances and advanced features. However it does not include all the information included in the *User Guide* tutorial. For a complete guide to PIE/TSO functions, see the *User Guide* tutorial as well.

Accessing PIE/TSO

To access PIE/TSO, log on to TSO using your standard logon method.

1. Log on to TSO.
2. You'll probably see the MultiTSO menu, which looks like the screen below. If not, enter `SWITCH` from TSO READY. This should switch you to the MultiTSO menu. If it doesn't, make sure your logon procedure invokes PIE/TSO. See the *PIE/TSO Installation and Customization Guide* for details.

```
PIE MultiTSO Menu ------------------------------------------(C) TSC, Inc 1991
 Command ===>
 ..#.. Name....  Title..............................Attributes  SwChar:  \
                                                                SwKey:   ENTER
                                                                Keys:   ON
                                                                Mkeys:  S ON
                                                                Pfshow: MKEYS
                                                                Asys:   ON
                                                                Abbrev: ON
                                                                ImpTSO: ON
                                                                AE:     ON
                                                                JMKey:  OFF
                                                                Screen: SAVE
                                                                Ver : 03.02.01
                                                                Sessions 0-11
                                                                Thu   Oct  8
                                                                Date: 10/08/92
                                                                Time: 17:23:39
                                                                User: JDOE
                                                                Term: TERM4C15
                                                                Sysid:  820A
```

The MultiTSO Menu

Most PIE/TSO commands are executed from the MultiTSO menu. While you may execute MultiTSO commands from a session, the commands are passed to the MultiTSO menu before executing. (PIE/TSO programs such as ACCESS or PIESTART are processed from TSO ready, like other TSO applications.)

The MultiTSO menu is divided into three sections: the header area, the information window, and the session information area. For more information on these areas, see Chapter 2 of the *User Guide*.

1. Before you proceed, note your SwChar (switch character) and your SwKey (switch key) in the information window on the right of the menu. You'll be using them to switch between sessions and enter commands. In our examples, we assume they are \ (backslash) and ENTER.

Opening Sessions

PIE/TSO provides several commands to open sessions:

CREATE	Opens a session, but does not switch you to it.
CREATEH	Opens a hidden session, but does not switch you to it.
START	Opens a session and switches to it.
STARTH	Opens a hidden session and switches to it.
ALLOC	Opens a session with the allocated attribute. When you end the session, it will remain on the MultiTSO menu. The ALLOC command does not switch you to the session.
ALLOCH	Opens a hidden session with the allocated attribute. When you end the session it will remain on the MultiTSO menu. The ALLOCH command does not switch you to the sessio

These session opening commands have the following format:

```
commandname [id] [command]
```

If you specify an *id*, MultiTSO will open the session you specify. If you omit it, MultiTSO will open the next available session.

A session opens at the TSO READY prompt. From TSO READY you can execute any TSO command. You can open a session and execute a TSO command at the same time by issuing the TSO command with the command parameter. The TSO command will be executed from the TSO READY prompt the first time you switch to the session.

1. Open a session with the START command. Enter `START` on the MultiTSO menu command line.

   Since you left out the *id* and *command*, MultiTSO will open the next available session and leave the session at the TSO READY prompt.

   If you switched right back to the MultiTSO menu, auto-end is on and is ending your session as soon as it hits TSO READY. For the purposes of this tutorial, turn auto-end off. Enter `SET AE OFF` from the MultiTSO menu command line. Now enter `START` again.

2. Now include a session ID to open session 3. Use the ALLOC command. Enter `PIEEXEC ALLOC 3`.

   The PIEEXEC command allows you to execute MultiTSO commands from TSO READY. There are some PIE/TSO commands that can be entered directly from TSO (such as CREATE). When you execute these commands with PIEEXEC, MultiTSO will open a new session and execute them from that session. It is usually best to execute these commands without PIEEXEC. For each command's requirements, see the PIE/TSO *Command Reference*.

3. Now switch back to the MultiTSO menu. Issue `\ [ENTER]`.

4. Open another session using CREATE. Enter `CREATE MSPF 2`.

   You included a TSO command: `MSPF 2`. Because you used the CREATE command, you did not switch to the session automatically. When you do switch to it, TSO will execute MSPF 2. `MSPF` is a special MultiTSO command that sets up `ISPF` files and then calls ISPF. MSPF allows you to run ISPF in more than one session at a time. You must use MSPF to run ISPF if you already have a session running ISPF. You can also use MSPF the first time you execute ISPF.

5. Switch to the session you just created to see the results of your last command. Issue `\1 [ENTER]`.

```
PIE MultiTSO Menu ---------------------------------------- (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title..............................Attributes  SwChar:  \
+ 0   PIEEXEC                                          S  SwKey:   ENTER
  1   MSPF      MSPF                                    S  Keys:   OFF
                                                           Mkeys:  M ON
* 3   ISPF      ISPF 2                                A S  Pfshow: MKEYS
                                                           Asys:   OFF
                                                           Abbrev: ON
                                                           ImpTSO: ON
                                                           AE:     OFF
                                                           JMKey:  OFF
                                                           Screen: SAVE
                                                           Ver : 03.02.01
                                                           Sessions.0-11
                                                           Mon   Oct 12
                                                           Date: 10/12/92
                                                           Time: 14:01:34
                                                           User: JDOE
                                                           Term: TERM4C15
                                                           Sysid:  820A

  F1=HELP      F2=START     F3=END       F4=CANCEL    F5=NEWS     F6=CREATE
  F7=          F8=          F9=          F10=SWKEY    F11=SWCHAR  F12=SWITCH
```

Switching Between Sessions

Because MultiTSO's primary function is to make it easy for you to get from one session to another, it provides several ways to switch between sessions. We'll discuss and practice using the switch sequence and switching from the MultiTSO menu.

Other switching commands are:

SWITCH	Switches to the MultiTSO menu or to the specified session.
SWITCH BACK	Switches between the last two sessions accessed.
SWITCH NEXT	Switches forward through sessions in numerical order.
SWCHAR	Sets the switch character.
SWKEY	Sets the switch key.
JMKEY	Sets the jump key.
HIDE	Hides a session, removing it from the jump key cycle.
UNHIDE	Unhides a session, returning it to the jump key cycle.
END	Pushes the current session to the bottom of the stack and switches to the last session accessed.
PUSH	Alias for END.
ENDBACK	Pushes the current session to the bottom of the stack and switches to the session you accessed before the current session.
PUSHJUMP	Alias for ENDBACK.
SET ASYS	Enables asynchronous switching.
SET ABBREV	Determines whether you can use abbreviations for session names.

⚠ WARNING

If SCREEN (displayed in your information window) is set to NOSAVE and you switch out of a session, you will lose any data that you have typed but haven't entered (by pressing ENTER or a PF key).

If you are running SCREEN NOSAVE, always press ENTER or a PF key before you switch out of the session. If SCREEN is set to SAVE, MultiTSO will effectively “press ENTER” before it switches.

For the purposes of this tutorial, it does not matter how SCREEN is set.

Using the Switch Sequence

The switch sequence is the most versatile switching tool in PIE/TSO. You can use it to switch to any session. You can even execute a command with it. You already used the switch sequence when you opened your sessions in the last section.

Requirements

To perform the switch sequence, you type the switch character, then press the switch key.

You must type the switch character in the first position of an input field on your screen.

+ If there is any data typed in this field but not yet entered (with ENTER or a PF key), it will be lost during the switch. This is true even when SCREEN is set to SAVE. So it is always a good idea to press ENTER or a function key before you use the switch sequence.

MultiTSO will ignore the switch character until you press the switch key. It will ignore the switch key if you don't type the switch character correctly (e.g., in the first position of an input field) first. That means if you press the switch key without first entering the switch character correctly, the program executing in that session will execute its normal processing for that key. So it is best to avoid having critical application commands assigned to the same key as your switch key.

You can change your switch character and key. Your switch character can be any non-alphanumeric character on the 3270 keyboard. Your switch key can be ENTER or any PF key. To change your switch character, issue the SWCHAR command. To change your switch key, issue the SWKEY command.

Some applications will only allow you to input numerics on the screen. To use the switch sequence in this type of application, be sure to change your switch character to + or - before you enter the application. (These are the only characters that are valid both on these screens and as switch characters.)

Options

You can use the switch sequence to switch from any session to the MultiTSO menu or to any other session. You can also use it to switch from the MultiTSO menu to any session.

If you type the switch character by itself and press the switch key, you will switch back and forth between the MultiTSO menu and the current session. (The current session is the last active session. It is marked with an asterisk on the MultiTSO menu.) So to get to the MultiTSO menu from the current session, you would issue your switch sequence without any parameters: `\ [ENTER]`.

If you type the session ID after the switch character, you will switch to that session. For example, to switch to session 2, you would issue `\2 [ENTER]`. You can identify a session by code or by name. So if session 1 is named ISPF, you can switch to it by issuing `\ISPF [ENTER]`.

You can also enter a PIE/TSO or TSO command after the switch character. If you enter a PIE/TSO command, MultiTSO will execute the command, but it won't switch you. For example, to hide all sessions from the current session, you could issue: `\HIDE ALL [ENTER]`.

If you enter a TSO command, MultiTSO will open a new session and execute the command in it. For example, to open a new session and execute SDSF in that session, issue `\SDSF [ENTER]`. (If you execute a command that is identical to a session name, MultiTSO will switch you to that session. It will not execute the command again.)

Executing TSO commands from the switch sequence requires that IMPTSO be set on. Set it on with the SET IMPTSO command. See the PIE/TSO *Command Reference* for details.

Practice

1. Issue `\ [ENTER]`. (Substitute your own switch character and switch key.) You switch back to the MultiTSO menu.
2. From the MultiTSO menu, issue `\ [ENTER]`. You'll switch to your ISPF 2 session, the current session.
3. Issue `\0 [ENTER]`. You'll switch to session 0.

   Notice that the TSO screen has been cleared. Because of TSO limitations, any line output or previous commands that were on the terminal screen are lost when you switch from line mode commands. To avoid losing data from line mode commands, run them in Session Manager.
4. Issue `\ [ENTER]`. You'll switch to the MultiTSO menu. Notice that session 0 is now the current session.
5. Now try executing commands with the switch sequence. From your ISPF session, issue `\LISTA [ENTER]`.

   This will open session 2 and issue the LISTA command in it.
6. Issue `\0 [ENTER]`.

   This switches you back to another TSO READY session. Both sessions are line mode. So MultiTSO doesn't clear the screen when you switch. It just adds the lines from the new session to the screen.
7. Now issue `\ALLOC [ENTER]`. This will create a new session.
8. Switch back to the MultiTSO menu to see it: `\ [ENTER]`.
9. Continue practicing the switch sequence until you can use it without having to think about it. When you are done, switch to the MultiTSO menu.

Switching from the MultiTSO Menu

The easiest way to switch from the MultiTSO menu to a session is to type the session's name or code on the MultiTSO menu command line and press ENTER.

1. Try it now. Type `2` on the MultiTSO command line and press ENTER. You switch to session 2.
2. Use your switch sequence to switch back to the MultiTSO menu. Then enter `0` on the command line. You switch to session 0. Simple.

   You are actually using the START command to switch. You may remember that START both opens a session and switches to it. If you request a session that is already open, START will simply switch you to that session. It won't open a new session.

   START is the default command on the MultiTSO menu. So if you type parameters on the MultiTSO menu command line press ENTER, MultiTSO will execute them as parameters to the START command.

   So when you enter a session code or name from the MultiTSO menu command line, you'll switch to that session.
3. Switch back to the MultiTSO menu and enter `MSPF` on the command line.

   You switch to session 1, which is named MSPF.

Identifying Sessions

Sessions are identified with both codes and names.

Session codes are one alpha-numeric character (except for the codes 10 and 11, which are two). They are shown in the # column of the MultiTSO menu. Session numbers are 0 through 11. The alphabetic codes are:

- A–Session 10
- B–Session 11
- H–Help session
- I–INFO session
- V–View session
- X–MultiTSO menu

The session name is displayed in the Name column on the MultiTSO menu. If default naming is on (using the SET DFLTNAME command), MultiTSO will automatically give each session a name. That name will be the first word of the last command you entered in the session. For instance, if you executed MSPF 3.2 in a session, that session's name would be MSPF. Each time you execute a new command, the session gets a new name.

1. You can assign a permanent name to a session with the NAME command. From the MultiTSO menu command line, enter `NAME 1 EDIT`.
2. If Abbrev is on, then you can abbreviate session names in commands and from the MultiTSO menu. If it isn't on already, turn it on now. Enter `SET ABBREV ON`.
3. Now switch to your EDIT session by typing `E` on the command line.

Anytime you need to identify a session, you can use either its name or its code. (Throughout the PIE/TSO manuals, the term ID refers to both session names and codes.) When you request a session, MultiTSO goes through a matching sequence.

- First it tries to match your string with a command. (For example, if you type HIDE, MultiTSO will execute the HIDE command.)
- Next it tries to match a session code.
- Next it tries to find a session with the exact name you typed.
- Last it tries to find a session that begins with the characters you typed.

So when you name a session or abbreviate a session name, you have to make sure that it will be the first match in this sequence. For example, don't abbreviate a session name to A, B, H, I, V, or X. You'll get the session the code refers to. Whenever in doubt, use the whole session name or its code.

In addition to session names and codes, there are two variables for identifying sessions. The * (asterisk) identifies the current session. The ? (question mark) identifies the last session you opened.

Exiting Sessions and Logging Off

There are several ways to terminate your sessions in MultiTSO. The following PIE/TSO commands affect exiting sessions and logging off.

LOGOFF	Logs you off a session. If no other sessions are active, it also logs you off TSO.
CANCEL	Abends the application in a session and ends the session.
CLOSE	Cancels a session but leaves the session on the MultiTSO menu.
DISC	Disconnects your TSO logon.
KEEP	Assigns the allocated attribute to a session. When you end the session, it will remain on the MultiTSO menu.
FREE	Removes the allocated attribute. When you end the session, it will be removed from the MultiTSO menu.
SET AE	Sets the auto-end attribute. When the application running in an auto-end session terminates, the session also terminates.
SET CMDFINAL	Sets a default TSO command to execute when you log off all sessions.
RESTART	Cancels a session and reopens it automatically.
LOGON	Cancels all your sessions and logs you back on to TSO.
STOP	Ends a session from a CLIST or REXX EXEC.

This section describes LOGOFF and CANCEL, which are variations of the TSO LOGOFF and CANCEL commands.

1. To end a session, first exit the application in the session. Then enter the LOGOFF command at TSO READY. Try this now. Session 0 is already at TSO READY. So go to session 0 and enter `LOGOFF`.

   You will return to the MultiTSO menu. Notice that the line for session 0 is gone from the session information area.

   You have to log off all sessions before you can log off TSO and PIE/TSO. If you want, you can abnormally terminate all your sessions and log off TSO with one command: `CANCEL ALL`.

⚠ WARNING

CANCEL abnormally terminates the application running in every canceled session. If you leave any unsaved data in any session, it will not be saved.

It is always best to switch to any session that may have unsaved changes or critical programs executing. Then you can save your changes and terminate programs normally in these sessions. When you are finished with critical sessions, you can safely use CANCEL ALL to abend all remaining sessions.

2. We don't have any critical sessions active in our tutorial—the CANCEL command is safe for our use. So to terminate all sessions, enter `CANCEL ALL` from the MultiTSO menu command line.

   All your sessions have been canceled, and you have been logged off TSO.

3. Log on again to continue with the tutorial.

Asynchronous Switching

Asynchronous switching allows you to switch out of a session while the application running in the session is still processing a task. When you switch out, you will go to the MultiTSO menu. From there you can perform any of your usual tasks. For instance, you can switch to any of your existing sessions or open new sessions and run any TSO application.

The session you switch out of is suspended until you return to it. Once you return, the application will continue to process its task as if it had never been interrupted.

You can use asynchronous switching to:

- suspend a long running task to answer a question
- set up a dump dataset for a task that failed so you don’t have to recreate the problem
- cancel a task that has gone into a loop

There are several circumstances under which asynchronous switching will not work. It is disabled:

- at the VTAM page prompt (***)
- by certain enqueues
- by a system must complete condition
- by a step must complete condition

Most of the time you will not be able to tell that you are in a system or step must complete condition or that the enqueue in effect prohibits switching. So if asynchronous switching doesn't work, don't assume that it is not working properly. Simply wait a moment or two and try it again.

1. Before you can use asynchronous switching, you must turn it on.

   To turn it on in all sessions, enter `SET ASYS ON ALL` from the MultiTSO menu.

   To set asynchronous switching on in particular sessions, enter `SET ASYS ON` *ids*. Identify the sessions with the *ids* parameter. Then enter `SET ASYS ON`.
2. Switch to a session and start a long task, such as a long query of a database.
3. If your application is initializing, wait for it to finish. (If you switch out while an application is initializing, it will probably abend when you switch back.)

   While your long running task is processing, switch out of the session:

- From an SNA terminal—press ATTN
- From a non-SNA terminal—press RESET and then PA1

  + When asynchronous switching is on, it overrides the normal use of your ATTN key (or RESET/PA1 keys) for the application. To attention out of your application, press ATTN twice (or RESET, PA1, PA1).

4. You have switched to the MultiTSO menu. Switch back to your suspended session, and it will resume processing.
5. Turn asynchronous switching off. Enter `SET ASYS OFF`.

Using MultiTask

MultiTask allows you to process a TSO application in a background session (a session that you don't see), while you work in a foreground session (a session that is displayed on your terminal screen).

1. You must turn MultiTask on the session you want to be your background session. Enter `SET MTASK ON` id from the MultiTSO menu. Or place the cursor in the command column next to the session and enter `M`. (The session you used to query your database would be a good choice here.)

   MultiTask will be activated in the session you specify. Once you activate it, MultiTask will remain on in the session until you activate it in another session or you turn it off. To turn it off, enter `SET MTASK OFF`.

   + You must be authorized to use MultiTask. See the PIE/TSO *Installation and Customization Guide* for details.
2. Now turn asynchronous switching on in your background session. Asynchronous switching must be on so that you can switch out of the session while it is running a task.

   Simply set asynchronous switching on for all sessions. Enter `SET ASYS ON ALL`.
3. Switch to your background session and start a long running task, such as a long query of a database.
4. Now use asynchronous switching to switch out of the session while the task is running. (Remember not to switch out while an application is initializing.)

- From an SNA terminal—press ATTN.
- From a non-SNA terminal—press RESET and then PA1.

   You are in the MultiTSO menu. You can open a new session and work in it, work in an existing session, anything. As you work in your foreground session, your background session will continue to process.

5. After a while (30 seconds, a minute), switch back to your background session to see what is going on. You will see the latest screen updates. Switch back out to continue working in other sessions.

   With MultiTask, you can switch in and out of background tasks as often as you want. However don't do so too quickly. You can cause the task to fail.

MultiTask gives control to the foreground session whenever you press ENTER or a PF key. It waits five seconds before it reactivates the background session. (It waits to minimize overhead.) So, the more you press ENTER or a PF key, the longer it will take to process your background session.

When processing is complete in the background session, MultiTask will send the following message to your foreground session:

```
PCS300I MULTITASK SESSION COMPLETE
```

If you turn MultiTask off, the current background session stops processing. However, it still exists, and it will resume processing as soon as you switch to it.

Using Network Access

Network Access allows you to access VTAM applications in your MultiTSO sessions. You invoke Network Access with the ACCESS command.

ACCESS is a TSO command. Issue it:

- from the MultiTSO menu (if the PIE/TSO command SET IMPTSO is set ON)
- from TSO READY
- in a CLIST or REXX EXEC in a session opened with CREATE, ALLOC, etc.

1. Open a Network Access session now. Enter `START ACCESS` from the MultiTSO menu.

   MultiTSO takes you to the Network Access menu, similar to the example below. Of course, your menu will be different. It will list the applications you have available at your installation. In fact, different users may have different Network Access menus.

```
PIE/TSO ---------------------- Network Access --------------- (C) TSC, Inc 1991
Cmd ==>
PF1/13 = Help   PF3/15 = END   PF7/19 = up   PF8/20 = down       Ver: 03.02.01
SELECT AN APPLICATION TO ACCESS
--#---System---- Description -------------------- Initial Text ----------------
  1   TSO        TIME SHARING OPTION
  12  JPODB      PRODUCTION DATABASE
  13  CICSA      PERSONNEL MANAGEMENT SYSTEM
  14  PROFS      ELECTRONIC MAIL
  15  IIN        IBM INFORMATION NETWORK

****************************** BOTTOM OF DATA ******************************
```

Starting an Application

You can start an application by selecting it from the menu.

1. Select the application with the cursor in the command column on the left. Type the letter `S`. Or you can select it by typing its menu code on the command line.

   In a moment, when you press ENTER, Network Access will establish a session with the APPLID you have selected. It will pass that application the information in the initial text area. For instance, if the initial text area for that application contains your user ID, Network Access will pass your user ID to the application automatically. Your initial text area may contain variables. For information on ACCESS variables, see “ACCESS” in the PIE/TSO *Command Reference*.

2. You may change the information in the initial text area, if you want. TAB over to the initial text area and type your user ID and logon procedure. (You can't supply passwords in the initial text area unless you are using the PIE Logon Director.)

   Your menu line will look something like the following:

```
 1   CICS       CICS TEST             george proc(gfd037)
```

3. Now press ENTER.

   Network Access will execute the application. You can enter your password and use the application.

Switching

To switch from Network Access sessions, you can use the switch sequence. The switch sequence works exactly the same in Network Access sessions as it does in standard MultiTSO sessions.

However in some cases you may use Network Access to log on to TSO on another system or with another user ID. If your second TSO logon also accesses MultiTSO, you won't be able to use the same switch sequence to move around inside the second MultiTSO session. To get around this, change either the switch key or switch character in the second MultiTSO session and use the new switch sequence to move around inside the second MultiTSO session. Use the original switch sequence to return to your original MultiTSO session.

Starting a Session and Application in One Step

For quicker access, you can bypass the Network Access menu and start Network Access and the VTAM application in one step. To do so, you will issue the ACCESS command with parameters. The parameters are:

```
ACCESS [applid]  [ data      ]
                 [ variables ]
```

Use *applid* to specify which VTAM application you want to use (for instance, `ACCESS CICSTST1`). Network Access will bypass the Network Access menu and connect you directly to the application you specify. You will see your normal initial screen for that application.

Use *data* to pass initial text to the VTAM application. For instance, to pass your user ID and logon procedure to CICSTST1 enter:

```
ACCESS CICSTST1 GEORGE PROC(GFD037)
```

The *variables* parameter is for executing the ACCESS command in CLISTs and REXX EXECs. For information on these variables, see the PIE/TSO *Command Reference*.

Start a Network Access session and a VTAM application in one step. From a new session at TSO READY, enter `ACCESS` *applid userid*. When you enter your password, you'll be ready to use the application.

Ending Network Access Sessions

To end a Network Access session, you must first log off your VTAM application.

If you started the APPLID from the Network Access menu, you will be returned to the Network Access menu. If you want, you can start a new VTAM application in the same session. To exit Network Access, you will use the END command (PF3).

1. Exit your first Network Access session now. Log off the application.
2. You are returned to the Network Access menu. Press PF3 to exit Network Access.
3. If you are still in a MultiTSO session, log off the session to exit it.

   Your installation may allow you to use the CANCEL command to end Network Access sessions. However this method is not recommended. Some VTAM applications, including CICS and IMS, do not always sign you off when you cancel them. So it is better to sign off and end your Network Access session than to cancel it.

Long Running Tasks

Network Access sessions can process in the background while you work in other sessions. To exit out of a Network Access session running a task,

- If you are on an SNA terminal—press ATTN.
- If you are on a non-SNA terminal—press the RESET key and use the switch sequence.

You will switch to the MultiTSO menu. Your Network Access session will continue to process its task just as if your terminal displayed it. When you return, your screen will contain the latest updates.

(If you are at an SNA terminal and you want to use the ATTN key in your application, press ATTN twice. If you are at a non-SNA terminal, you must set your ATTN key function using the SET AATN or SET APFn command.)

MultiTSO Special Sessions

MultiTSO provides special sessions that don't count toward your twelve maximum sessions. The special sessions are:

- HELP–provides help on all PIE/TSO commands. Access it with the H command.
- INFO–provides general programming information, such as hex conversion tables. Access it with the I command.
- View–allows you to capture and manipulate screen images

All these sessions run in full-screen format, using standard PF key settings. The only session that requires special explanation is the View session.

Using View

View captures and saves screen images. You can compare these images, share them, or use them in training and documentation. View runs in its own special session. Its session ID is V.

1. Before you can use View, you must have a View dataset allocated. Your installation may have set up a global dataset. You can use this dataset or create and allocate your own.

   If you want your own View dataset, create a partitioned dataset with a logical record length of 80. It can have any blocksize.

   Then allocate it with the ddname PIEVIEW. (This ddname can be changed with the VIEWDD start-up option.) Enter the following command from TSO READY:

   `ALLOC FI(PIEVIEW) DA(`*dataset-name*`) SHR`

2. To start View, enter the V command from the MultiTSO menu: `V`
3. To capture a screen image, go to the session containing the screen you want to capture and issue the V command using the switch sequence.

   Go to a session and issue `\V [ENTER]` from any input field. View will capture the image and send you to the View menu.
4. To display the screen image, place the cursor on the View menu command line, and press PF6

for EDIT.

5. To exit the image, press PF3 for END.

   You may save your image at this time, if you want. To do so, enter a member name for the image at the prompt. The name can be up to 8 characters long.

6. To send an image to another user, first capture it. Then type the user's ID on the View menu command line and press PF5 for SEND. (You can send one to yourself for practice.)
7. To receive an image, go to the View menu and press PF9 for RECEIVE.
8. To switch between View and other PIE/TSO sessions, use the switch sequence.
9. To print a screen image, switch to a TSO session. Use your standard printing procedures and specify your View dataset and the appropriate member name.
10. To delete a screen image, delete its member name using your ISPF facilities.
11. To close your View session from within View, issue END or RETURN from the View menu. From the MultiTSO menu, enter `CAN V`.

Chapter 3 Commands

This chapter describes special advanced commands and outlines some commands that are especially useful in CLISTs, EXECs, and for problem solving.

There are many useful commands that aren't mentioned in this chapter. For information on these commands and for information on typographical conventions and issuing commands, see the *PIE/TSO Command Reference*.

Special Commands

There are several privileged commands for system programmers only. UNICOM Systems' Customer Services may ask you to execute one of these commands to help answer a question or diagnose a problem.

To use these commands, the user ID affected must be defined in the PCSUSERS table with CLASS=015. If the user ID does not have the proper authorization, you will receive an INVALID COMMAND message.

1. To get authorization, specify `CLASS=015` in the PIE/TSO PCSUSERS table for the user ID involved. (015 indicates classes 0, 1, and 5. 0 and 1 are always required.) Reassemble the table and re-log on to TSO.

   See the *PIE/TSO Installation and Customization Guide* for more information.

LOCATE

The LOCATE command displays the contents of certain MultiTSO and MVS control block address spaces.

FORMAT

```
LOCATE id
```

Issue locate from the MultiTSO menu or with PIEEXEC or the switch sequence.

PARAMETER

id Specify the code of the session you want to display.

Parameter	Valid	Default
id	0 to 11 or 1 to 8 character name	None

EXAMPLE

To display the control block addresses for session 1, enter `LOCATE 1`

TRACE CARO and TRACE ACCESS

TRACE CARO and TRACE ACCESS are debugging aids for MultiTSO and Network Access. They provide specific trace records for UNICOM Systems' development staff regarding MultiTSO and Network Access. They are not for general debugging. Do not enable them unless UNICOM Systems' Customer Services requests you to do so.

To use TRACE CARO and TRACE ACCESS, perform the following.

1. From the MultiTSO menu command line, enter `TRACE CARO ON` and `TRACE ACCESS ON`. There will be no message or response to these commands.
2. Run the GTF VTAM Buffer Trace.

   CARO and ACCESS records will appear along with the standard records for this trace.
3. Turn TRACE CARO and TRACE ACCESS off. Enter `TRACE CARO OFF` and `TRACE ACCESS OFF`.

Other Useful Commands

This section describes normal MultiTSO commands and features that are especially useful to advanced users and PIE/TSO administrators. See the *PIE/TSO Command Reference* for details on any of these commands.

PIEEXEC

The PIEEXEC command allows you to issue PIE/TSO commands from TSO READY or from applications as TSO commands. For instance, you can execute the PIE/TSO ALLOC command from TSO READY as `PIEEXEC ALLOC` and it would execute just as though you had entered it directly from the MultiTSO menu.

You can also use PIEEXEC to avoid assigning aliases to PIE/TSO commands that have the same name as TSO commands. When you execute the command using PIEEXEC, PIE/TSO will execute its own command. For instance, if you want to execute PIE/TSO CANCEL instead of TSO CANCEL, you can enter `PIEEXEC CANCEL ALL`. This will execute PIE/TSO's CANCEL ALL command, just as if you had entered CANCEL ALL from the MultiTSO menu.

PIEEXEC is not necessary for the following commands:

@ALLOC	LOCK	STARTH
@CAN	MSPF and MULTISPF*	STOP*
All @SET commands	NOTE	SWCHAR
ACCESS*	PEXEC*	SWITCH
CREATE	PIECHECK*	SWKEY
CREATEH	PIESTART*	UNHIDE
HIDE	SCREEN	
JMKEY	START	

The commands marked with an asterisk (*) are TSO command processors. Normally you would execute these commands directly from TSO READY or from an application as TSO commands. The rest of the commands are aliases to PIEEXEC—so you don't need to use PIEEXEC when you execute them.

If you execute any of these commands with the PIEEXEC command, MultiTSO will open a new session and execute the command from that session. (This is controlled by the SET IMPTSO command.)

DISPLAY

DISPLAY provides information about your sessions, the CPU, and storage utilization. If Network Access is active, it will display the name of the virtual terminal in use.

The format of the DISPLAY command is:

```
Display [SYStem|id|STOrage]
```

1. To display information about your system, enter `D`. (SYSTEM is the default.)
2. To display information about session 3, enter `D 3`.
3. To display information about storage use, enter `D STO`.

FENCE

Some commands, such as APL2, may take the entire region for work space when they initialize. When that happens, there is no storage left to open new sessions. The FENCE command reserves a predetermined amount of storage for such cases.

This amount is set with the FENCE start-up option. The distributed default is 256K. You may change this default with the FENCE command.

1. Change the amount of storage reserved to 1M. Enter `FENCE 1M` from the MultiTSO menu.

1. Before you start the storage intensive application, enter `FENCE ON`. MultiTSO will reserve storage space for new sessions.
2. Open a session and start the application.

   When you switch out of the session the reserved storage will be freed. You will have memory available, and you may open a new session.

PEXEC

PEXEC is an interface to the PIE/TSO Performance Group Manager (PGM). PGM contains a table that assigns performance group numbers and SWAP/NOSWAP status to applications. These assignments were initialized when PIE/TSO was installed. For more information on them, see the PIE/TSO *Installation and Customization Guide*.

PGM does not change the assignment for an application unless you tell it to. You can tell PGM to change the assignment in either of two ways:

- by executing the application with the PEXEC command
- by defining the application as an alias to PEXEC

If the application is an alias, PGM will change the performance group assignments automatically when the application is requested. If it is not an alias, you must execute it as a parameter to the PEXEC command.

EXAMPLE

You want to execute ACCESS with the CICS170 APPLID. You want to use the values in the PGM table. Enter `PEXEC CMD ACCESS CICS170`.

ACCESS will start, and the performance group number and swap values will be set as specified in the PGM table. When you switch out of the ACCESS session, your performance group number and swap values will be reset to their original values. The performance group number and swap values are maintained by session and are set and reset as you switch from session to session.

SET PARSER

SET PARSER designates which MultiTSO parser (release 3.1.0+ or 2.5.3) will be enabled.

The 3.1.0 parser allows you to identify sessions with more than one character. This means sessions can have names (like SAS, ISPF, EDIT, etc.), and you can identify them by the numbers 0 through 11. The 3.1.0 parser requires you to use spaces between session IDs—whether names or session codes—when you have more than one in a command. For example, to cancel sessions 1, 2, and 3, you must enter `CANCEL 1 2 3`.

The 2.5.3 parser limits session IDs to one character. This means you can't name a session and that you must identify sessions with the characters 0 through 9, A, and B. It allows you to identify multiple sessions without using spaces.

EXAMPLE

The 3.1.0 parser is the default. To go back to the 2.5.3 parser, enter `SET PARSER OLD` from the MultiTSO menu.

Controlling Function Keys

You can define PF and PA keys as shortcuts to perform any PIE/TSO command. You can activate these keys in the MultiTSO menu or in any MultiTSO session. When you are in a session, the MultiTSO function key overrides the application's function key. You can use MultiTSO function keys to:

- change menu attributes
- perform switching functions
- switch to a specific session

If you like, you can have two sets of function keys: MKEYS and SKEYS. The MKEYS operate only in the MultiTSO menu. The SKEYS can operate in both sessions and the MultiTSO menu.

USING BOTH SETS OF PF KEYS

If you want to have one set of keys in the MultiTSO menu and another set of keys in your sessions, you can set up both sets of function keys, and activate the MKEYS on the MultiTSO menu and the SKEYS in your sessions.

1. To do so, first set up your MKEYS. Assign PF key functions with the SET MPFn command. For instance, to set PF2 to switch to your EDIT session, enter `SET MPF2 SWITCH EDIT`.

   + Session variables can be useful in PF keys. To identify the current session, use an asterisk

(e.g., SET MPF2 CLOSE *). To identify the last session you opened, use a question mark (e.g., SET MPF2 SWITCH ?).

2. To set the PA3 key, use the SET MPA3 command. For example: SET MPA3 RECALL
3. Now activate the MKEYS on the MultiTSO menu. Enter SET MKEYS MKEYS and SET MKEYS ON.
4. Now set your up your SKEYS. Assign PF key functions with the SET PFn command. SET PFn is just like the SET MPFn command. So, to set PF2 to switch back to MultiTSO menu, enter SET PF2 SWITCH.
5. Set the PA3 key with SET PA3. For example: SET PA3 SWITCH BACK. (SWITCH BACK switches you between the last two sessions you accessed.)
6. Now enter SET KEYS ON ALL to activate the SKEYS in your sessions.

USING THE SAME SET OF PF KEYS

If you want to use the same set of keys in both the MultiTSO menu and your sessions, set up only the SKEYS and activate the SKEYS in both your sessions and the MultiTSO menu.

1. To start, set up your SKEYS. Assign PF key functions with the SET PFn command. For instance, to set PF2 to switch between your current session and the MultiTSO menu, enter SET PF2 SWITCH.
2. If you like, set the PA3 key using the SET PA3 command. For example, to set PA3 to SWITCH BACK, enter SET PA3 SWITCH BACK.
3. Activate the SKEYS on the MultiTSO menu. Enter SET MKEYS SKEYS and SET MKEYS ON.
4. Activate the SKEYS in your sessions. Enter SET KEYS ON ALL.

DISPLAYING PF KEYS

Use the PFSHOW command to display PF keys on your MultiTSO menu.

1. Enter PFSHOW ON to turn the PF key display on.
2. Enter PFSHOW PF1 to display PF keys 1 through 12. Enter PFSHOW PF13 to display PF keys 13 through 24. Enter PFSHOW PA1 to display your PA keys
3. Enter PFSHOW MKEYS to display the MKEYS. Enter PFSHOW SKEYS to display the SKEYS.
4. You can change the label on any PF key. Use the SET MPFnL command to change MKEYS PF key labels. Use the SET PFnL command to change SKEYS PF key labels. Use the PA3L and MPA3L commands to change PA3 key labels. For example, to change the MKEYS PF2 label to EDIT, enter SET MPF2L EDIT.

Controlling the MultiTSO Menu Display

You can customize the display of the MultiTSO menu to suit your needs. To customize the MultiTSO menu, use the following commands.

SET INFOWINDOW	Controls the MultiTSO menu information window.
SET SESATT	Controls the session attribute display.
SET TITLE	Controls the title line.
SET HEADER	Controls the header line.
RESHOW	Refreshes the MultiTSO menu.

SET MENU	Enables the MultiTSO menu display.

Building Multiple Sessions Automatically

If you execute an initial CLIST or REXX EXEC at logon, you can open sessions and start applications automatically, name your sessions and create comments for them, set up your PF keys, change your switch sequence, and so on. That way, your PIE/TSO environment is ready every time you log on. The MultiTSO menu will display your opened sessions, comments, etc. To use an application, all you have to do is enter the session code on the command line.

```
PIE MultiTSO Menu ------------------------------------------ (C) TSC, Inc 1991
Command ===>
..#.. Name....  Title.............................Attributes  SwChar:  \
  1   ISPF      ISPF                                  HKA S  SwKey:  ENTER
  2   EDIT      ISPF 2 (EDIT)                          KA S  Keys:   ON
  3   DSLIST    ISPF 3.4 (DSLIST)                      KA S  Mkeys:  M ON
  4   CICSTEST  Test CICS                              KA S  Pfshow: MKEYS
  5   INFO      Customer Service Database              KA S  Asys:   OFF
  6   NAMENU    Network Access Menu                    KA S  Abbrev: ON
                                                             ImpTSO: ON
                                                             AE:     OFF
                                                             JMKey:  PF21
                                                             Screen: SAVE
                                                             Ver : 03.02.01
                                                             Sessions.0-11
                                                             Mon   Oct 12
                                                             Date: 10/12/92
                                                             Time: 14:01:34
                                                             User: JDOE
                                                             Term: TERM4C15
                                                             Sysid:  820A

  F1=HELP      F2=START     F3=CAN *    F4=ISPF      F5=NEWS     F6=CREATE
  F7=DISC      F8=          F9=        F10=SWKEY    F11=SWCHAR  F12=SWITCH
```

The Basics

You can issue all PIE/TSO commands from CLISTs and REXX EXECs. We will show you a few basic commands in this section. There are many more commands available. To get a feel for the possibilities, see the PIE/TSO *Command Reference*.

You'll be better able to create initial CLISTs and EXECs if you understand how the commands are executed. Your logon procedure starts a session and executes the CLIST/EXEC in that session. Your commands execute at TSO READY; so you must code every command in TSO format. Do not code abbreviations—code the full command name. Remember that some commands, such as ALLOC, must be preceded with PIEEXEC to execute at TSO READY. Other commands, such as CREATE, execute at TSO READY by default. For example, to create one session and allocate another, you would code:

```
CREATE
PIEEXEC ALLOC
```

(Refer to the PIE/TSO *Command Reference* for specific command requirements.)

Setting Up an Initial CLIST/EXEC

To execute the initial CLIST or REXX EXEC:

1. Include a PARM statement in the logon procedure as described in "Step 2: PCSEFT01 EXEC Parameters," on page 20, of the *PIE/TSO Installation and Customization Guide*.
2. In the PARM statement, include the multitso-command parameter and specify `START`. Also include the tso-command parameter and specify the name of the initial CLIST or EXEC. For example, if your CLIST/EXEC is named INIT, your PARM statement should be:
   `PARM='START %INIT'`
3. If you want your users to be able to modify their own CLISTs/EXECs, tell them what to name the CLIST/EXEC in their own CLIST library. Instructions for creating an initial CLIST/EXEC are also included in the PIE/TSO *User Guide*.

Hints For Creating an Effective CLIST/EXEC

1. Your main objective in an initial CLIST/EXEC is probably to open sessions automatically. There are several session opening commands, but ALLOC is the best one for CLISTs/EXECs. That is because ALLOC opens sessions without switching to them. (Once you switch out of the session executing the CLIST/EXEC, it stops executing until you return to it.) Also, sessions opened with ALLOC remain on the MultiTSO menu even after you end them. So your MultiTSO menu doesn't change when you end a session.
2. When you open a session, you can start a TSO command in it, too. In the menu on the previous page, we started an application in every session we opened, except session 0, which is native TSO. We started ISPF in session 1 (`PIEEXEC ALLOC 1 ISPF`), ISPF 2 in session 2 (`PIEEXEC ALLOC 2 MSPF 2`), and so forth. The TSO command will execute when you switch to the session the first time.
3. Never execute ISPF in a CLIST or EXEC. Always execute MSPF. This ensures that MultiTSO can execute ISPF successfully no matter which session you switch to first.
4. To create the session menu effect, you can give each session a name and a comment. Use the NAME and NOTE commands to do this.
5. To change your switch key, switch character, and jump key, use the SWKEY, SWCHAR, and JMKEY commands, respectively.
6. You may want to change your PF keys as well. See " Controlling Function Keys" on page 22, for these commands.
7. When you are finished opening sessions and defining the environment, you may want to switch back to the MultiTSO menu. That way you are presented with the MultiTSO menu when you log on, instead of the session in which the CLIST/EXEC executed. Use the SWITCH command to switch to the MultiTSO menu, or cancel the session your CLIST/EXEC is running in with `PIEEXEC CANCEL *`.

8. At times you will want to interrupt your CLIST/EXEC and execute something in another session. To do so, use any switching command (such as SWITCH or START) to switch to the other session. When you switch back to the session executing the CLIST/EXEC, it will resume processing.
9. You can set up several ISPF sessions with different options. Pass parameters for these options just as you would pass them to ISPSTART. For example, to allocate a session that will present panel ZSDSFOP2, code

   ```
   ALLOC MSPF PANEL(ZSDSFOP2) NEWAPPL(ISF)
   ```

   To allocate a session that will present the RMFWDM command, code

   ```
   ALLOC MSPF CMD(RMFWDM)
   ```

   To allocate a session that will present the program ICEPMD01, code

   ```
   ALLOC MSPF PGM(ICEPMD01) NEWAPPL(ICE) NOCHECK
   ```

10. You may want to simplify access to tools you don't use very often. In this case, you want pre-allocation to be automated, but you don't want to waste time and resources on these tools every time you log on. To accomplish this, create a separate CLIST/EXEC for the tool. Then when you allocate the session for it, execute that CLIST or EXEC. The CLIST/EXEC will only execute if you switch to the session.

    For example, to set up INFOSYS with two databases, create a CLIST named INFO containing the following statements.

    ```
    ALLOC DDN(VSAM) DSN('INFO.SUPPORT') SHR
    ALLOC DDN(VSAM2) DSN('INFO.VSAM') SHR
    PEXEC PGM BLGOZD   '&PARM.'
    FREE DDN(VSAM VSAM2)
    ```

    This will allocate your databases, start INFOSYS, and, when you exit the session, free the datasets. Then in your initial CLIST, allocate the INFOSYS session as follows:

    ```
    PIEEXEC ALLOC  %INFO
    PIEEXEC NAME ? INFO
    PIEEXEC NOTE ? Customer Service Database
    ```

    (The ? is a variable for the last session opened.)
11. If you want to fence storage from an application automatically, you must execute the application from a separate CLIST/EXEC and code FENCE ON in that CLIST/EXEC before you start the application. (If you code FENCE ON in your initial CLIST/EXEC, you can't ensure that it will apply to the correct application. It will apply only to the application running in the first session you switch to.) For example, you could start APL2 with the following CLIST:

    ```
    PIEEXEC FENCE ON
    APL2
    ```

12. To automate View, pre-allocate your View dataset, then allocate your View session, as follows:

    ```
    ALLOC FI(PIEVIEW) DA(dataset-name) SHR
    PIEEXEC V
    ```

13. PIESTART is a special command that is useful in CLISTs/EXECs or can be used in place of them.

    When you execute PIESTART, you indicate the session by name and you may include a

command to execute in the session. If the session does not yet exist, PIESTART will execute START and NAME, to open and switch to the next available session and to give it the name you indicated. If you include a command, PIESTART will execute that command in the session.

If the session already exists, PIESTART will switch you to that session. It will ignore any command you include to execute in the session.

PIESTART format is:

```
PIESTART //name/command
```

14. PIECHECK determines whether PIE/TSO is active. It passes a return code to notify you of PIE/TSO status. The return codes are listed in the PIE/TSO *Command Reference*. With PIECHECK your initial CLIST/EXEC can perform special processing if MultiTSO is not active.

Sample CLIST

The following is a sample initial CLIST. It activates the jump key and sets it to PF21. It sets PF keys for the MultiTSO menu and for sessions and turns them on. SET AE ON ALL prevents you from seeing TSO READY when you end an application.

The CLIST allocates sessions 1 through 6. In this case, your logon procedure starts session 0. The CLIST also names these sessions and creates menu comments for them. When all is finished, it cancels the current session and switches you back to the MultiTSO menu.

```
JMKEY PF21
@SET MKEYS ON
@SET MPF3  CAN *
@SET MPF7  DISC
@SET KEYS  ON ALL
@SET PF4   SWITCH ISPF
@SET PF11  LOCK
@SET AE ON ALL
PIEEXEC ALLOCH 1 MSPF 2
PIEEXEC NAME   1 EDIT
PIEEXEC NOTE   1 ISPF 2 (EDIT)
PIEEXEC ALLOC  2 MSPF CMD(RMFWDM)
PIEEXEC NAME   2 RMF3
PIEEXEC NOTE   2 RMF Monitor III
PIEEXEC ALLOC  3 MSPF 3.4
PIEEXEC NAME   3 DSLIST
PIEEXEC NOTE   3 ISPF 3.4 (DSLIST)
PIEEXEC ALLOC  4 ACCESS CICSTEST &&ZSNON
PIEEXEC NAME   4 CICSTEST
PIEEXEC NOTE   4 Test CICS
PIEEXEC ALLOC  5 %INFOMVS
PIEEXEC NAME   5 INFO
PIEEXEC NOTE   5 Customer Service Database
PIEEXEC ALLOC  6 ACCESS
PIEEXEC NAME   6 NAMENU
PIEEXEC NOTE   6 Network Access Menu
PIEEXEC CANCEL *
```

Sample REXX EXEC

The following sample is found in member PIEINIT3 of the PIE/TSO CLIST dataset.

```
/* REXX */
"PIEEXEC"
/*************************************************************/
/* Change the MultiTSO Title                                 */
/*************************************************************/
"PIEEXEC SET TITLE TSO Application Menu"
/*************************************************************/
/* Control MultiTSO Menu display                             */
/*************************************************************/
"PIEEXEC PFSHOW OFF"
"PIEEXEC SET NOTEPAD OFF"
/*************************************************************/
/* Assign MultiTSO PF Keys                                   */
/*************************************************************/
"PIEEXEC SET PF2 SPF"
"PIEEXEC SET PF3 DIDOCS"
"PIEEXEC SET PF4 OM"
"PIEEXEC SET PF5 TSCDB"
"PIEEXEC SET PF10 ENDBACK"
"PIEEXEC SET PF11 SET ASYS OPPOSITE"
"PIEEXEC SET PF12 SET KEYS OPPOSITE"
"PIEEXEC SET PF13 SWITCH BACK"
/*************************************************************/
/* Make the MultiTSO Menu Keys the same as the session keys */
/*************************************************************/
"PIEEXEC SET MKEYS SKEYS
/*************************************************************/
/* Assign Specific labels to some PF keys                    */
/*************************************************************/
"PIEEXEC SET PF11L ASYS OPP"
"PIEEXEC SET PF12L KEYS OPP"
"PIEEXEC SET PF13L BACK"
/*************************************************************/
/* Assign Switch Key and Character                           */
/*************************************************************/
"PIEEXEC SWKEY ENTER"
"PIEEXEC SWCHAR {"
/*************************************************************/
/* Turn Jump Key OFF                                         */
/*************************************************************/
"PIEEXEC JMKEY OFF"
/*************************************************************/
/* Allocate and assign names to sessions                     */
/*************************************************************/
"PIEEXEC ALLOCH DIDOCS"
"PIEEXEC NAME 1 DIDOCS"
"PIEEXEC ALLOCH %MYSPF PGM(OMNI#ISP)"
```

```
"PIEEXEC NAME 2 OMNI"
"PIEEXEC ALLOC %MYSPF"
"PIEEXEC NAME 3 SPF"
"PIEEXEC ALLOCH %MYIPCS"
"PIEEXEC NAME 4 IPCS"
"PIEEXEC ALLOCH %MYSPF RMF3"
"PIEEXEC NAME 5 RMF3"
"PIEEXEC ALLOC ACCESS TSCDB &ZSNON"
"PIEEXEC NAME 6 TSCDB"
"PIEEXEC ALLOCH RMFMON"
"PIEEXEC NAME 7 RMF"
/*************************************************************/
/* Assign notes                                              */
/*************************************************************/
"PIEEXEC NOTE TSCDB ELECTRONIC MAIL SYSTEM"
"PIEEXEC NOTE SPF ISPF/CATALYST"
"PIEEXEC NOTE DIDOCS MVS SYSTEM CONSOLE"
"PIEEXEC NOTE OMNI OMNIJES CONTROLLER"
"PIEEXEC NOTE RMF3 RMF MONITOR III"
"PIEEXEC NOTE RMF RMFMON"
"PIEEXEC NOTE IPCS DUMP AND TRACE ANALYSIS"
/*************************************************************/
/* Control Asynchronous Switching                            */
/*************************************************************/
"PIEEXEC SET ASYS ON ALL"
/*************************************************************/
/* Control Autoend                                           */
/*************************************************************/
"PIEEXEC SET AE ON ALL"
```

Chapter 4 Maintenance and Operation

This chapter describes all PIE/TSO maintenance and operation procedures.

Starting and Stopping PIE/TSO

Use PCSINIT to start and stop PIE/TSO. We recommend that you start and stop PIE/TSO with a COMMNDxx member. (Adding PCSINIT to a COMMNDxx member is part of the installation process.) However you may also start and stop PIE/TSO by issuing the ACT command from the operator console.

FORMAT

```
                  ┌ START ┐
                  │ STOP  │
S PCSINIT,ACT     │= PURGE│
                  │ NEWPARM
                  └ REPLACE┘
```

START	Activates MultiTSO if it was disabled with ACT=STOP. START will re-enable PIE/TSO using the start-up options currently in effect. (If you have made changes to your SESSIONS member and want to put them into effect, use NEWPARM.) START does not reload the PIE/TSO global modules and control blocks into the CSA/ECSA. START is the ACT default.
STOP	Deactivates PIE/TSO without removing its SVC front end. All users logged on to PIE/TSO will be able to continue using their active session and log off TSO. However users will not be able to switch or to create new sessions. Users who get a MultiTSO menu may wait for PIE/TSO to resume or issue the CANCEL ALL command.
NEWPARM	Updates PIE/TSO start-up options from the SESSIONS member of the PARMLIB dataset. It does not reload the PIE/TSO global modules and control blocks into the CSA/ECSA. Use it to put new start-up options into effect. To realize the changes, users must log off and back on.
PURGE	Deactivates PIE/TSO and removes its SVC front end. Do not use it unless the SVC front end is malfunctioning. The effect on users is the same as for STOP.
REPLACE	Performs ACT=PURGE then ACT=START. Use it to reload PIE/TSO global modules into CSA/ECSA when you have applied PTFs to an active PIE/TSO system. (To realize changes made to the SESSIONS member of the PARMLIB dataset, users must log off and back on.) Each time you use ACT=REPLACE you deplete ECSA by 24K plus 256

bytes per logged on user and CSA by 32K. This storage can be reclaimed only with an IPL.

Effects of Stopping PIE/TSO

Any users who log on to TSO while PIE/TSO is stopped will execute under normal TSO, without PIE/TSO. These users must log on to TSO again to use PIE/TSO after it is restarted.

Users already logged on to PIE/TSO will remain in the session they were using. All other sessions remain in a wait state. When PIE/TSO is restarted, any waiting sessions will be restored, and all PIE/TSO functions will become available automatically. Users may remain in the session on their terminal until PIE/TSO is restarted.

If users want to access waiting sessions, they must log off the session they are currently using. When they log off, they will access the last session used, until all sessions are logged off or until they access the MultiTSO menu. If they reach the MultiTSO menu, the only valid command until PIE/TSO restarts is CANCEL ALL.

Users who cancel out of PIE/TSO and log on to stand-alone TSO must log off and log back on to TSO to access PIE/TSO when it restarts.

Starting and Stopping Network Access

Use PNAIPL to start and stop Network Access independently of PIE/TSO. Before starting PNAIPL, run PCSINIT.

Use the following command to start and stop Network Access from the operator console.

FORMAT

```
S PNAIPL,ACT [= START ]
             [  STOP  ]
             [ REPLACE]
             [  TEST  ]
```

START	Activates Network Access if it was disabled with ACT=STOP. START is the ACT default.
STOP	Quiesces Network Access. Current Network Access users will be unaffected. However no new Network Access sessions can be created until you restart Network Access. Users do not have to log off and on again.
REPLACE	Performs ACT=PURGE then ACT=START. Use it to reload Network Access global modules into CSA/ECSA when you have applied PTFs to an active Network Access system.
TEST	Checks the PARMLIB dataset for errors. It issues a message indicating whether Network Access is active, passing return code 0 if Network Access is active or return code 4 if it is not. Use ACT=TEST to verify that changes to the PARMLIB dataset are correct. You may also use it in job streams to determine whether Network Access is active.

Terminating Current Network Access Users

To stop Network Access and to terminate all existing Network Access sessions, use the VTAM VARY command: `V NET,INACT,ID=pie`. Substitute the VTAM major node for all Network Access virtual terminal applications for pie. For example: `V NET,INACT,ID=APAPIE01`

You may use the I (immediate) or F (force) options on the VARY command, if you like.

Recording Resource Savings

You can produce reports that analyze the resources you save by using PIE/TSO.

Hint: The reports use session names as a data key. Have your users assign session names that reflect the applications being run in them.

⚠ WARNING

Use SMF recording only on MVS/ESA version 3 and above. It will cause abends on MVS/XA.

Step 1: Record SMF Data

1. Include the SMFRTY start-up option in your PARMLIB(SESSIONS) member. To record data, specify an SMF record type (128-255) that is not being used by any other application.
   Example: `SMFRTY=227`

   To turn off recording, specify `SMFRTY=NONE`.
2. Update your SMFPARMxx member. Include the record type you specified in the SMFRTY start-up option in the TSO subsystem statement.
   Example: `SUBSYS(TSO,TYPE(227))`
3. To activate the new member issue `SET SMF=xx` from the console. Substitute the last two characters of the SMFPARMxx for *xx*.
4. Run PCSINIT with ACT=NEWPARM to put your SMFRTY start-up option into effect.
   Example: `S PCSINIT,ACT=NEWPARM`

   PIE/TSO will begin recording data for all users who log on after this point. When you have gathered enough data, run the analysis program.

Step 2: Run the Analysis Program

Member SASRPTS of the SAMPLIB dataset analyzes and reports PIE/TSO resource savings. It analyzes the SMF records PIE/TSO generates. The reports it creates are described in the following section.

SASRPTS is written in the base SAS language and requires SAS version 5.18 or later.

1. To run SASRPTS, edit the JCL contained in member SMFRPTS of the CNTL dataset.
2. Update the following DD statements and run the JCL.

   `DUMPIN` Specify the SMF dataset containing the PIE/TSO measurement records. In the first step, this JCL will dump this dataset using the IBM utility IFASMFDP. In the second step, SAS will use this dataset to make its reports.

   `DUMPOUT` Specify the dataset to dump the SMF records into.

   `RPT1` Specify the SYSOUT dataset to contain the Input Diagnostic report. This report can be very large. You may wish to dummy this DD statement.

   `RPT2` Specify the SYSOUT dataset to contain the User ID summary report.

   `RPT3` Specify the SYSOUT dataset to contain the Application summary report.

   `PARMS` Code as `//PARMS DD *` and follow it with three parameter statements:

   ```
   FROM fromdate fromtime
   TO todate totime
   SMFTYPE nnn
   ```

   fromdate and *todate* specify the date range to be processed, in MM:DD:YY format.

   fromtime and *totime* specify time cutoffs in HH:MM:SS format. The report will include data beginning with fromdate, fromtime and ending with todate, totime.

   nnn specify the SMF record type you assigned to the PIE/TSO measurement records with the SMFRTY start-up option.

Example

This example will process SMF records of type 227 written between 10:00 AM, January 14, 1993, and 6:00 PM, January 20, 1993.

```
//SMFREP  JOB  (PIE),'PRINT SAS RPTS',CLASS=A,NOTIFY=&SYSUID,
//             MSGCLASS=X,REGION=5M
//DUMP     EXEC PGM=IFASMFDP
//SYSPRINT DD   SYSOUT=*
//DUMPIN   DD   DISP=SHR,DSN=SYS1.MAN1
//DUMPOUT  DD   DISP=(,CATLG,DELETE),SPACE=(TRK,(100)),
//  DCB=(RECFM=VBS,BLKSIZE=23476,LRECL=32760,DSORG=PS),
//  UNIT=SYSALLDA,DSN=PIE.SMFDATA
//SYSIN    DD   *
 INDD(DUMPIN,OPTIONS(DUMP))
 OUTDD(DUMPOUT,TYPE(000:255))
/*
//*
//REPORTS  EXEC  SAS
//SYSIN    DD    DISP=SHR,DSN=PIE.PTS310.SAMPLIB(SASRPTS)
//SMFDATA  DD    DISP=SHR,DSN=*.DUMP.DUMPOUT
//RPT1     DD    SYSOUT=*
//RPT2     DD    SYSOUT=*
//RPT3     DD    SYSOUT=*
//PARMS    DD    *
 FROM 01/14/93 10:00:00
 TO 01/20/93 18:00:00
 SMFTYPE 227
/*
```

PIE/TSO Resource Savings Reports

The SMF records generated by PIE/TSO are used to generate three reports. They are the Input Diagnostic, the User Summary, and the Application Summary reports. The User Summary and Application Summary reports are each printed three times, in three different sort sequences.

Input Diagnostic

The Input Diagnostic report essentially displays the raw data from each SMF record. At the top of each page it shows the TSO user ID, the date and time the SMF record was created, and the step completion code associated with the TSO session.

Immediately to the right of these numbers, the report lists the resources used for initial logon. At present, these values are not being captured and will be zero.

Below this summary data, the report displays the resource use and savings data for each PIE/TSO session. For each session, the following information is displayed:

- The session name. For the report to be most meaningful, session names should be chosen to reflect the application being run in that session.
- The resources actually consumed to start up the session. These values consist of wall clock time and service units. Service units are categorized as CPU, SRB, I/O, and storage.
- The number of times the user switched into the session. Each time the user switches to the session represents an application start-up that was avoided. In other words, by permitting the user to switch out of and then back into an application, PIE/TSO eliminates the need to shut the application down and start it up again.
- The initialization resource expenditures avoided. These numbers represent the resources that were not expended, i.e., saved through the use of PIE/TSO. They are reported as wall clock time and service units, just like the initialization resources. They are calculated as the initialization resources multiplied by the number of switches.

User Summary

The User Summary report summarizes the resources saved by PIE/TSO for each TSO user. Each line of the report corresponds to one TSO user. At the top of each page the report lists the date/time range included in the report.

Each detail line includes the following values.

- The TSO user ID.
- The number of actual application starts (initializations). This number corresponds to the number of sessions shown on the Input Diagnostic.
- The number of application starts avoided. This is the total number of switches reported for all sessions for this user.
- The total resources saved (not expended). These are reported as wall clock time and service units, categorized as described above for the Input Diagnostic.

The data in this report is summarized using the TSO ID as the key. If a user logs on more than once during the period measured, data for all of the logons will be summarized into a single line on the report.

Application Summary

This report summarizes the resources saved by PIE/TSO for each application. Each line of the report corresponds to one application (session name). At the top of each page are displayed the date/time range included in the report.

Each detail line includes the following values.

- The application (session) name.
- The number of actual application starts (initializations). This number corresponds to the number of sessions shown on the Input Diagnostic.
- The number of application starts avoided. This is the total number of switches reported for all sessions for this user.
- The total resources saved (not expended). These are reported as wall clock time and service units, categorized as described above for the Input Diagnostic.

The data in this report is summarized using the application (session) name as the key. So if different users use the same session name, the data for that session name will be summarized into a single line on the report. In order to ensure that the reported data is as meaningful as possible, you should make sure that a given session name always corresponds to the same application regardless of which user ID it is used by. For example, sessions named SDSF will be combined into a single detail line in this report. Therefore, make sure that any session named SDSF is really used to run the SDSF program in order to make sure that the numbers reported for SDSF really correspond to that application.

Adding New Users to PIE/TSO

In order to use PIE/TSO, each user's logon procedure must invoke the MultiTSO address space control program, PCSEFT01. For logon procedure requirements, see "Modify Your Logon Procedure" in the *PIE/TSO Installation and Customization Guide*. See "PCSEFT01 EXEC Parameters" in the *PIE/TSO Installation and Customization Guide* for logon procedure options.

Maintaining the PCSDFLD@ Module

At times you may need to reassemble PCSDFLD@. You need to do so

- after you have applied MVS, TSO, DFP, or VTAM maintenance
- if you want to redefine a parameter, such as security support
- when module PCSEFT01 issues the following message at logon:

  ```
  PIE/TSO MULTITSO AND MVS LEVELS ARE INCOMPATIBLE -
  INFORM SYSTEM PROGRAMMERS
  ```

  If this message is issued, users will still be allowed to log on, but it will appear as if PIE/TSO has not been activated.

To update PCSDFLD@, perform the following procedure.

1. Edit member PCSDFLD@ of the CNTL dataset.
2. Specify your security system in the OPTIONS parm. Specify ACF2, RACF, RACR (for RACROUTE), TOPS (for TOP SECRET), or NONE.
3. Run the job and check its output.

   PCSDFLD@ provides support for all versions of TSO through TSO/E 2.4.1. Depending on the version available at your installation, you may get assembler MNOTE statements when you reassemble. If the assembly generates a return code of 4 or less, these MNOTEs are normal. If your return code is higher, save the assembly listing, and call UNICOM Systems' Customer Services.

   If you use the PIE/TSO load library from the distribution tape as the APF authorized library, the COPYAPF step will end with a condition code of 8 because IEBCOPY will think that a "member select" is being attempted during a compress. If this occurs the IEBCOPY will fail with a parameter validation error before doing the compress or the copy. In this case, 8 is an acceptable return code for the COPYAPF job step. PCSDFLD@ must reside in an APF authorized library or implicitly authorized library, such as link list.

Maintaining the PCSVPSWD Module

The LOCK command needs to communicate with your security system to validate user passwords. It does so through the PCSVPSWD module. PCSVPSWD is set up initially by the installation job. However you must modify and reassemble it if you change your security system at a later date.

1. Edit member PCSVPSWD of the CNTL dataset.
2. Specify your security system in the OPTIONS parm. Specify ACF2, RACF, RACR (for RACROUTE), TOPS (for TOP SECRET), or NONE.

   If you code any other value, PCSVPSWD will generate condition code 16. If you code NONE, it will generate condition code 12. In either case, no new load module will be created, and the default module for UADS support will be used for the LOCK command.

Troubleshooting

Review the *PIE/TSO Release Notes* shipped with your tape. Following its instructions will prevent many common errors.

Periodically, UNICOM Sytems' makes available a PTF tape for PIE/TSO. We will send you a notice each time the tape becomes available. We recommend that you request this tape and apply all PTF fixes regularly to prevent problems from occurring.

As we discover IBM PTFs or APARs that affect PIE/TSO, we add them to the HELP dataset. These HELP entries are named ZEWSnnnn. If there is a corresponding entry in the IBM INFO database, we will refer to it. (We cannot distribute actual entries extracted from INFO, since they are copyrighted.) Please read these HELP entries to see if any of the reported problems could happen at your site.

System 878 Abends

Some TSO commands use the MVS GETMAIN macro to acquire a work area. Occasionally this may cause a TSO address space to run out of storage. Because you are using multiple sessions and thus using more storage, you may get 878 or 80A abends. If this occurs, increase the region's virtual storage. We recommend a region size of 5-7MB.

To determine your region size, the total amount of virtual storage left, and the largest contiguous amount of virtual storage, issue the MultiTSO command `D STO`.

Page Dataset Requirements

PIE/TSO users tend to use more memory in their address spaces, because they have more applications loaded. This often changes the requirement for the size of local page datasets.

If your local page datasets are nearly full with a normal non-PIE/TSO work load, increase them to compensate for the increased amount of allocated storage made possible by PIE/TSO. The rule of thumb is to multiply the expected average number of logged on PIE/TSO users by the expected number of active sessions by the current working set size of an average TSO user. This gives the total PIE/TSO related storage requirement in bytes.

PIE/TSO can cause an increase in page stealing on systems with existing paging problems if more than 40 PIE/TSO users are active.

Screen Doesn't Restore After a Message

If the screen for an application doesn't restore when you press ENTER after a message or after you disconnect/reconnect, the application may not support the TSO RESHOW function. RESHOW support is required for MultiTSO compatibility.

If the application is user-written, see the IBM *TSO Guide to Writing a TMP or a CP*, "Screen Content Restoration," for information on supporting RESHOW.

If the application is vendor written, please call UNICOM Sytems' Customer Services.

Varying Network Access Terminals Inactive

If a user reports problems with a particular Network Access session, you can use the VTAM VARY command to vary it inactive and active. This will break the VTAM session between the virtual terminal and the application. For example, enter:

```
V NET,INACT,ID=PIE10001,I
V NET,ACT,ID=PIE10001
```

To determine the virtual terminal name, have the user perform the following procedure.

1. Switch to the MultiTSO menu. If the keyboard is locked, press ATTN twice or press RESET and then enter the switch sequence.
2. From the MultiTSO menu, execute the *DISPLAY* id command, where *id* is the Network Access session ID.

   The response will show the virtual terminal name and the application name.

   ```
   ACCESS SESSION 6 - PIEZ0001 TO JPODB  FROM 13:28
   ```

Index

Symbols

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

PIE/TSO
Messages and Codes

Release 3.2.1

UNICOM
SYSTEMS, INC.

PTSOIG321-01

Printed: October 2002

This manual applies to PIE/TSO release 3.2.1 and to all subsequent releases of the product until otherwise indicated by new editions or updates to this publication.

UNICOM Systems, Inc.
Unicom Plaza - Suite 310
15535 San Fernando Mission Blvd.
Mission Hills, California 91345

Contents

Preface

This manual contains all messages produced by PIE/TSO. It provides the message text along with an explanation of its meaning and information on corrective action. It also explains MNOTEs generated by errors in assemblies and user abend codes from PIE/TSO modules.

Messages are arranged first by the issuing component. Then they are sorted by message number. Unnumbered messages are listed in alphabetical order.

Who is this manual for?

This manual is written for PIE/TSO administrators, system programmers, and operators.

Customer Services

Troubleshooting suggestions

We have found that many problems can be solved by looking into a few common problem areas. We suggest you look into the following.

- Have you made any recent changes to your installation, including changes to MVS, TSO, and other products? Are these changes compatible with PIE/TSO?
- Is the current release of PIE/TSO installed?
- Is your PIE/TSO password current and correctly entered?
- Are there any relevant messages on your PIE/TSO or TSO logs?

Contacting Customer Services

WWW.UNICOMSI.COM/SUPPORT

UNICOM Systems' normal business hours are from 7:00 a.m. to 4:00 p.m. Pacific Time, Monday through Friday. However, emergency customer service is available 24 hours a day, 7 days a week.

When you call Customer Services, please make sure you have the following information available:

- related error messages
- the command or JCL you are inputting
- the dump, if one is generated

CUSTOMER SERVICES PHONE

Our phone number is (818) 838-0606.

If you call after hours our answering service will take the call. You may leave a message to be returned during normal working hours. Or you may leave an urgent message, and a technical support representative will be paged to call you back immediately. When leaving an urgent message, be sure to leave your after-hours phone number.

If you've left an urgent message but haven't received a call back within an hour, please call back. We may be having trouble reaching you.

☞ International customers, please contact your distributor for technical assistance.

CUSTOMER SERVICES FAX

Our FAX number is (818) 838-0776.

Chapter 1
PCSINIT Messages

Numbered Messages

PCS000I
Explanation: PCSINIT was executed without a SYSPRINT DD statement. Message PCS013E will also be issued, as well as the headings, PCS010I, PCS011I and PCS12I. PCSINIT will list all parameter dataset statements on the console using this message identifier.

System Action: None.

Operator Response: None.

Programmer Response: None.

PCS010I PIE/TSO RELEASE rr.vv.mm PAGE nn
PCS011I ON CPU cpuid model
PCS012I AT hh:mm:ss ON yy.ddd

Explanation: PCSINIT was executed without a SYSPRINT DD statement. Message PCS013E was already issued. These are the heading lines for the output listing.

System Action: None.

Operator Response: None.

Programmer Response: None.

PCS013I UNABLE TO OPEN SYSPRINT DATASET

Explanation: The OPEN for the output dataset failed.

System Action: PCSINIT will begin using the operator console to output all messages.

Operator Response: None.

Programmer Response: Check the allocation for the SYSPRINT dataset.

PCS020E UNABLE TO OPEN SYSIN DATASET, TERMINATING
Explanation: PCSINIT was unable to OPEN the parameter dataset.

System Action: PCSINIT will terminate without initializing PIE/TSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Check the allocation for the SYSIN dataset.

PCS021E NO PIE/TSO PARAMETERS FOUND, TERMINATING
Explanation: The parameter dataset pointed to by the SYSIN DD statement was empty.

System Action: PCSINIT will terminate without initializing PIE/TSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Check the allocation for the SYSIN dataset.

PCS022E PIE/TSO MULTITSO REQUIRED PARAMETERS MISSING
Explanation: PCSINIT must find three valid parameter keywords in PARMLIB(SESSIONS). They are the CPU ID/password parameter (CPUID), the expiration date parameter (EXPDT), and the installation SVC number (SVC). One or more of the above parameters were missing from the parameter dataset.

System Action: PCSINIT will terminate without initializing PIE/TSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Supply all required parameters.

PCS023E PCSINIT TERMINATED DUE TO PARAMETER ERRORS
Explanation: PCSINIT detected an error in the PARMLIB(SESSIONS) dataset. More information will appear in the SYSPRINT output dataset.

System Action: PCSINIT will terminate without initializing PIE/TSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Check the output from PCSINIT and correct any errors flagged.

PCS024E EXPIRATION DATE OF yy.ddd HAS EXPIRED, PIE TERMINATING
Explanation: The license period for PIE/TSO has expired. Do not attempt to tamper with the EXPDT parameter in the parameter dataset. Any attempt to do so will result in PIE/TSO becoming unusable.

System Action: PCSINIT will terminate without initializing PIE/TSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Contact UNICOM Systems, Inc. to extend your license.

PCS025E PIE/TSO MULTITSO NOT LICENSED ON THIS CPU
Explanation: MultiTSO has not been licensed on this CPU. Do not attempt to tamper with the CPUIDs in the parameter dataset. Any attempt to do so will result in MultiTSO becoming unusable.

System Action: PCSINIT will terminate without initializing MultiTSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: If the primary CPU is unusable, alternate CPU's can be substituted on the license agreement, either temporarily or permanently. To do so, contact UNICOM Systems, Inc. to modify your license.

TSC PCS026E PCSINIT EXECUTED AS A JOB OR TSO COMMAND
Explanation: PCSINIT was executed as a batch job or from TSO. This is not allowed except for the ACT=TEST option.

System Action: PCSINIT will terminate without initializing MultiTSO.

Operator Response: None.

Programmer Response: Do not attempt to execute PCSINIT except as a started task or with the ACT=TEST option.

PCS027E PCSIPL MUST BE AN AUTHORIZED PROGRAM
Explanation: PCSIPL was executed from a non-APF authorized library or has been link edited without the AC=1 parameter.

System Action: PCSINIT will terminate without initializing MultiTSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Move PCSIPL to an APF authorized library and ensure that it has been link edited with APF authorization.

PCS030E INVALID PARM cccc, IGNORED
Explanation: PCSINIT was executed with an invalid ACT parameter.

System Action: PCSINIT will terminate without initializing MultiTSO. A TEST function will be performed.

Operator Response: Ensure that the ACT parameter specified in the START PCSINIT command is valid. See the PIE/TSO Administrator Guide for valid parameters. If the START command was issued using the default installation supplied ACT, contact the systems programmer responsible for PIE/TSO immediately. Issue S PCSINIT,ACT=x with the desired ACT parameter.

Programmer Response: Ensure the default ACT keyword in the PCSINIT procedure is correct.

PCS031E PIE/TSO MULTITSO NOT ACTIVE, CAN'T STOP
Explanation: PCSINIT was executed with an ACT=STOP or REPLACE parameter, but MultiTSO was not active.

System Action: PCSINIT will terminate without stopping or replacing MultiTSO.

Operator Response: Restart MultiTSO if desired by starting PCSINIT with an ACT=START.

Programmer Response: None.

PCS032E UNABLE TO STOP PIE/TSO MULTITSO
Explanation: PCSINIT has found that the user SVC table entry used by PIE/TSO changed after PCSINIT started PIE/TSO. This message will always appear with message PIE033E below.

System Action: PCSINIT will terminate without deleting or replacing PIE/TSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately. Issue the DUMP command if possible.

Programmer Response: A storage overlay of CSA or the nucleus has probably occurred, overlaying MultiTSO read-only control blocks. Use the DUMP command to obtain a dump of CSA and the nucleus.

PCS033E PIE SVC TABLE ENTRY AT hhhhhh ALTERED FROM vvvvvvvv TO xxxxxxxx
Explanation: PCSINIT has found that the user SVC table entry used by MultiTSO changed after PCSINIT was used to start MultiTSO. This message will always appear with message PIE0323E above.

hhhhhhhh is the nucleus address of the suspected overlay. vvvvvvvv is the original value stored at this address by PCSINIT. xxxxxxxx is the current value at this address.

System Action: PCSINIT will terminate without deleting or replacing MultiTSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately. Issue the DUMP command if possible.

Programmer Response: A storage overlay of the nucleus has probably occurred, corrupting MultiTSO read-only control blocks. Use the DUMP command to get a dump of the nucleus.

PCS034I PIE/TSO MULTITSO TSO INTERFACE TERMINATED
Explanation: In response to a STOP or REPLACE request, PCSINIT has terminated MultiTSO.

System Action: MultiTSO is terminated.

Operator Response: None.

Programmer Response: None.

PCS035I PIE/TSO MULTITSO NOT ACTIVE, IGC0009C IS AT xxxxxxxx
Explanation: PCSINIT has been run with the TEST option. MultiTSO is not currently active. The address of SVC 93 is reported as xxxxxxxxx.

System Action: None.

Operator Response: None.

Programmer Response: None.

PCS036I PIE/TSO MULTITSO ACTIVE, PIE/TSO MULTITSO SVC AT xxxxxxxx PCSPARMS AT yyyyyyyy IGC0009C AT zzzzzzzz
Explanation: PCSINIT has been run with the TEST option. MultiTSO is active. The address of MultiTSO SVC is xxxxxxxx, MultiTSO parameters are at yyyyyyyy, the address of SVC 93 is zzzzzzz.

System Action: None.

Operator Response: None.

Programmer Response: None.

PCS037E CSA OVERLAY DETECTED AT @SMPARMS ADDRESS hhhhhhhh CONTENTS IS yyyyyyyy
Explanation: PCSINIT has found that a MultiTSO read only control block, @SMPARMS at address hhhhhhhh, has been overlaid in CSA.

System Action: PCSINIT will abend with a user completion code of 900.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: A storage overlay of CSA has probably occurred, corrupting a MultiTSO read-only control block. Use the DUMP taken to check for a CSA overlay.

PCS040E UNABLE TO OPEN SYSLIB DATASET, TERMINATING
Explanation: PCSINIT has found that the SYSLIB dataset required for START and REPLACE could not be opened.

System Action: PCSINIT will terminate without initializing or replacing MultiTSO.

Operator Response: If APF was entered with the “S PCSINIT” command ensure there was no typing error.

If there was no error, contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Check the SYSLIB DD statement in the PCSINIT procedure.

PCS041I PIE/TSO MULTITSO rr.vv.mm SUCCESSFULLY INITIALIZED
Explanation: In response to a START or REPLACE request, PCSINIT has initialized MultiTSO. rr.vv.mm is the release level.

System Action: PCSINIT will terminate. MultiTSO is initialized.

Operator Response: None.

Programmer Response: None.

PCS042E ERRORS WITH SALLOC LOCK - MULTITSO TERMINATING
Explanation: In response to a START or REPLACE request, PCSINIT was unable to obtain the SALLOC lock.

System Action: PCSINIT will abend with a user 300 completion code.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Contact UNICOM Systems, Inc. for assistance.

PCS043E CSA OVERLAY DETECTED AT @SMPARMS ADDRESS hhhhhhhh CONTENTS IS vvvvvvvv
Explanation: An attempt was made to START MultiTSO while MultiTSO was already initialized. At this time a CSA overlay of vvvvvvvv at address hhhhhhhh was detected in a MultiTSO read only control block.

System Action: PCSINIT will abend with a user 900 completion code.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Use the dump provided to check the CSA overlay.

PCS044W UNABLE TO START PIE/TSO MULTITSO - ALREADY ACTIVE
Explanation: An attempt was made to START MultiTSO while MultiTSO was already initialized.

System Action: PCSINIT will terminate.

Operator Response: None.

Programmer Response: None.

PCS045E SQA/CSA CRITICAL, REQUEST ABORTED
Explanation: An attempt was made to START MultiTSO. There was not enough SQA or CSA available.

System Action: PCSINIT will terminate with a user 120 abend.

Operator Response: Contact Systems Programming.

Programmer Response: Correct the CSA or SQA shortage problem. See the PIE/TSO Installation and Customization Guide for details.

PCS046E LOAD UNSUCCESSFUL REASON CODE n - PIE/TSO MULTITSO TERMINATING
Explanation: An internal error occurred attempting to load MultiTSO modules during a START or REPLACE.

System Action: PCSINIT will terminate with a user 220 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc..

PCS047E CVT MODIFIED - PIE/TSO MULTITSO TERMINATING
Explanation: The MVS CVT was modified during PCSINIT execution of START or REPLACE.

System Action: PCSINIT will terminate with a user 600 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc.

PCS048E CVT MODIFIED DURING LOAD - PIE/TSO MULTITSO TERMINATING
Explanation: The MVS CVT was modified during PCSINIT execution of START or REPLACE.

System Action: PCSINIT will terminate with a user 690 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc.

PCS049E PIE SVC LOAD MODULE INVALID - PIE/TSO MULTITSO INITIALIZATION FAILED
Explanation: The PCSSVC01 load module was found to be invalid during PCSINIT execution of START or REPLACE.

System Action: PCSINIT will terminate with a user 700 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc. Ensure the SYSLIB DD statement is valid.

PCS050E PCSINIT INTERNAL ERROR ON cccccccc PIE/TSO MULTITSO TERMINATING
Explanation: An internal error was detected in PCSINIT. cccccccc is the internal error designation.

System Action: PCSINIT will terminate with a user 710 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc.

PCS051E CAN'T FIND cccccccc MODULE
Explanation: A required module, cccccccc, was not in the dataset pointed to by the PCSINIT procedure SYSLIB DD statement during a START or REPLACE request.

System Action: PCSINIT will terminate without initializing MultiTSO.

Operator Response: If APF was entered with the “S PCSINIT” command ensure there was no typing error.

If there was no error, contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Check the validity of the SYSLIB DD statement.

PCS052E BLDL ERROR ACCESSING MODULE cccccccc RETURN CODE n REASON CODE y
Explanation: An error occurred during a BLDL for module cccccccc. The return code n and reason code y are provided.

System Action: PCSINIT will terminate without initializing MultiTSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Check the validity of the SYSLIB DD statement. The BLDL return codes and reason codes can be found in the IBM Data Management Services Guide.

PCS053W SVC TABLE ENTRY NOT AS EXPECTED X'hhhhhhhh' X'hhhhhhhh'
Explanation: PCSINIT checks the SVCTABLE entry for SVC 93 to ensure that it is running at a known level of MVS. The level of MVS at your account is not known, or the SVC 93 entry has been overlaid. The message contains a hex dump of the entry.

System Action: PCSINIT will continue initializing MultiTSO.

Operator Response: Contact the systems programmer responsible for PIE/TSO immediately.

Programmer Response: Contact UNICOM Systems, Inc. immediately.

PCS054I @SPARM LOADED AT X'address'
Explanation: This message indicates where @SPARM has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS055I @SPARMX LOADED AT X'address'
Explanation: This message indicates where @SPARMX has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS056I PT@PDIES LOADED AT X'address'

Explanation: This message indicates where PT@PDIES has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS057I PCSSVC01 LOADED AT X'address'

Explanation: This message indicates where PCSSVC01 has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS058I PT@CARO1 LOADED AT X'address'

Explanation: This message indicates where PT@CARO1 has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS059I PT@CARO2 LOADED AT X'address'

Explanation: This message indicates where PT@PCARO2 has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS060I PT@PRCTS LOADED AT X'address'

Explanation: This message indicates where PT@PRCTS has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS061I PCSSVC02 LOADED AT X'address'

Explanation: This message indicates where PCSSVC02 has been loaded in global storage. It is issued after first time PIE/TSO activation and after starting PIE/TSO with ACT=REPLACE.

System Action: None.

Operator Response: None.

Programmer Response: None

PCS120I END OF PARAMETERS
Explanation: This message will appear only on the SYSPRINT dataset. The end of the parameter dataset has been reached.

System Action: None.

Operator Response: None.

Programmer Response: None.

PCS121W EXPIRATION DATE OF yy.ddd WILL EXPIRE IN n DAYS
Explanation: The MultiTSO license agreement will expire on yy.ddd. Do not attempt to modify the EXPDT parameter or MultiTSO will become unusable.

System Action: None.

Operator Response: None.

Programmer Response: Contact UNICOM Systems, Inc. for a license extension.

PCS220E UNKNOWN KEYWORD
Explanation: This message will appear only on the SYSPRINT dataset. The previous statement contained a keyword not known to PCSINIT.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS221E *
Explanation: This message will appear only on the SYSPRINT dataset. The previous statement contained an error. The asterisk (*) flags the column where the error was detected.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS222E DUPLICATE PARAMETER, IGNORED
Explanation: This message will appear only on the SYSPRINT dataset. The previous statement contained a keyword already processed from the parameter dataset. It will be ignored.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS223E INVALID SWITCH CHARACTER c
Explanation: This message will appear only on the SYSPRINT dataset. The specified switch character, c, is not a valid switch character. Valid switch characters are non-alphanumeric.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS224E ccc INVALID SWITCH KEY
Explanation: This message will appear only on the SYSPRINT dataset. The switch key specified is not in valid format. Specify PF1 to PF24 or ENTER.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS225E INVALID SWITCH KEY - cccc ONLY PF KEYS ALLOWED
Explanation: This message will appear only on the SYSPRINT dataset. The switch key specified is not PF1-PF24.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS226E INVALID APF PARAMETER cccccc
Explanation: This message will appear only on the SYSPRINT dataset. The APF parameter was not SWITCH or NOSWITCH.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS227E cccc INVALID EXPIRATION DATE FORMAT
Explanation: This message will appear only on the SYSPRINT dataset. The EXPDT parameter was not in the format yy.ddd. Do not tamper with the EXPDT parameter or MultiTSO will become unusable.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS228E cccc IS AN INVALID VALUE FOR HELP LINES
Explanation: This message will appear only on the SYSPRINT dataset. The HELPLN parameter was invalid.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS229E ccccccccccccccccccccc IS AN INVALID VALUE FOR A DSNAME
Explanation: This message will appear only on the SYSPRINT dataset. The HELPDS parameter was invalid.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS230E INVALID TMP NAME cccccccc
Explanation: This message will appear only on the SYSPRINT dataset. The TMP parameter was invalid for a load module.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS231E nn IS AN INVALID VALUE FOR MAXIMUM SESSIONS
Explanation: This message will appear only on the SYSPRINT dataset. The MAXSES parameter was not numeric between 2 and 12.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS232E INVALID SECONDARY TMP NAME ccccccc
Explanation: This message will appear only on the SYSPRINT dataset. The SECTMP parameter was invalid for a load module.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS233E USER SVC NUMBER NO LONGER NEEDED
Explanation: This message will appear only on the SYSPRINT dataset. The SVC is not needed.

System Action: PCSINIT will continue to process input parameters and execute.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS234E INVALID INPUT CARD
Explanation: This message will appear only on the SYSPRINT dataset. An input statement did not begin with a keyword, asterisk, or CPU serial number.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS235W MINIMUM HELPLN IS 200, CHANGED TO 200
Explanation: This message will appear only on the SYSPRINT dataset. The HELPLN parameter specified was less than the minimum allowed. The HELPLN parameter was defaulted to 200.

System Action: PCSINIT will continue to process input parameters and execute.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS236E cccccccc IS AN INVALID VALUE FOR A DDNAME
Explanation: This message will appear only on the SYSPRINT dataset. The HELPDD parameter specified an invalid format DD name.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS237E TEXT MUST START WITH A SINGLE QUOTE
Explanation: This message will appear only on the SYSPRINT dataset. The value for a CMDTXT, HDRTXT, TITLE1, or TITLE2 parameter does not start with a single quote.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS238E TITLE1 MORE THAN 70 CHARACTERS
Explanation: This message will appear only on the SYSPRINT dataset. The value for TITLE1 has more than 70 characters.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Specify the first 70 characters of the MultiTSO menu title using the TITLE1 parameter. Specify the remaining characters using the TITLE2 parameter.

PCS239E INVALID ACCESS PARAMETER
Explanation: ACCESS must be CANCEL or NOCANCEL.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS240E INVALID MCLASS/UCLASS/SWSCAN NUMBER n
Explanation: An invalid or non-numeric value was specified for the listed parameter, either MCLASS, UCLASS, or SWSCAN.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS241E INVALID PREFIX CHARACTER
Explanation: An invalid character was coded for the prefix value. Valid characters are $, @, #, alphabetic, or numeric.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS242E INVALID USERS TABLE NAME
Explanation: An invalid name was coded for the USERS operand. The name must be 8 characters or less and begin with an alpha or national character.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS243E TITLE2 MORE THAN 9 CHARACTERS
Explanation: This message will appear only on the SYSPRINT dataset. The value for TITLE2 has more than 9 characters.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Specify the first 70 characters of the MultiTSO menu title using the TITLE1 parameter. Specify the remaining 9 characters using the TITLE2 parameter.

PCS244E CMDTXT MORE THAN 12 CHARACTERS
Explanation: This message will appear only on the SYSPRINT dataset. The value for the CMDTXT parameter is more than 12 characters.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS245E HDRTXT MORE THAN 64 CHARACTERS
Explanation: This message will appear only on the SYSPRINT dataset. The value for the CMDTXT parameter is more than 12 characters.

System Action: PCSINIT will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PCS250E TSO LEVEL WAS NOT DETERMINED - STOP PIE
Explanation: Program PCSTSVFY could not determine what release of TSO was active.

System Action: PCSTSVFY will abend with a User 102 code. MultiTSO will be disabled.

Operator Response: None.

Programmer Response: Save all output from PCSINIT, including this dump and all SYSOUT datasets. Call UNICOM Systems, Inc. for assistance.

PCS251E TSO VERIFICATION ROUTINE WAS NOT A JOBSTEP
Explanation: Program PCSTSVFY was not run as a job step.

System Action: PCSTSVFY will abend with a User 100 code. MultiTSO will be disabled.

Operator Response: None.

Programmer Response: Correct the JCL used to run PCSTSVFY. PCSTSVFY may only be run as a started task.

PCS252E TSO VERIFICATION ATTEMPTED FROM FOREGROUND
Explanation: Program PCSTSVFY was run as a command in a TSO user's address space.

System Action: PCSTSVFY will abend with a User 101 code. MultiTSO will be disabled.

Operator Response: None.

Programmer Response: PCSTSVFY may only be run as a started task. Do not execute it as a TSO command.

Abend Completion Codes

The codes listed below are all user abend codes issued by PCSINIT.

100
Explanation: PCSEFT01 was not executed from the job step TCB. It was attached or otherwise entered from a lower level TCB.

Programmer Response: Do not attempt to execute the PCSEFT01 program as anything other than the job step TCB. Change the TSO logon procedure used.

110
Explanation: PCSEFT01 was not executed in a TSO address space.

Programmer Response: Do not attempt to execute the PCSEFT01 program other than in a TSO address space.

120
Explanation: PCSIPL was not able to GETMAIN the CSA or SQA required to initialize MultiTSO. This abend is always preceded by message PCS045E. If SQA was specified, the amount requested was less than 128 bytes.

If CSA was specified, the amount requested was less than 4K.

Programmer Response: Check to see if there are any problems with CSA or SQA. If not save the dump provided, and contact UNICOM Systems, Inc.

200
Explanation: Unable to attach the TMP or SECTMP specified. MultiTSO is active.

Programmer Response: Check the TMP or SECTMP specified in the PIE/TSO MultiTSO PARMLIB dataset and the logon procedure.

204
Explanation: Unable to XCTL to the TMP or SECTMP specified. MultiTSO is not active.

Programmer Response: Check the TMP or SECTMP specified in the MultiTSO PARMLIB dataset and the logon procedure.

220
Explanation: An internal error occurred attempting to load MultiTSO modules during PCSINIT. This abend is always preceded by message PCS046E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

300
Explanation: Unable to obtain the SALLOC lock during PCSIPL processing. This abend will always be accompanied by a PCS042E message.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

310
Explanation: Unable to free the SALLOC lock during PCSIPL processing.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

320
Explanation: An internal error occurred attempting to load MultiTSO modules during PCSINIT. This abend is always preceded by message PCS046E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

500
Explanation: The calculated password does not match the specified password. MultiTSO will not initialize.

Programmer Response: The value specified for EXPDT or for CPUID is not correct. Change whichever one is incorrect to the values supplied by UNICOM Systems. If the password has expired, please call UNICOM Systems, Inc.

600
Explanation: The CVT was dynamically modified before the execution of PCSINIT. This abend is always preceded by message PCS047E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

666
Explanation: Internal error.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

690
Explanation: The CVT was dynamically modified before the execution of PCSINIT. This abend is always preceded by message PCS048E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

700
Explanation: The PCSSVC01 load module from SYSLIB was invalid during PCSINIT. This abend is always preceded by message PCS049E.

Programmer Response: Check that the dataset specified on the SYSLIB DD statement, or APF parameter is valid. Save the dump provided, and contact UNICOM Systems, Inc.

710
Explanation: An internal error occurred during PCSINIT. This abend is always preceded by message PCS050E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

788
Explanation: In response to a user request made with the “ABENDPIE DUMP” command, MultiTSO has taken a dump.

Programmer Response: None

789
Explanation: In response to a user request made with the “ABENDPIE NODUMP” command, MultiTSO has abended without a dump.

Programmer Response: None

800
Explanation: A MultiTSO attempt to process the PCSSWEQT table received a bad return code from the GQSCAN macro.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

900
Explanation: A MultiTSO read only control block in CSA has been overlaid. This abend will always be preceded by a PCS037E message, or a PCS042E message and a PCS043E message.

Programmer Response: Check for a probable CSA overlay. Save the dump, and contact UNICOM Systems, Inc.

901, 902, 903, 904
Explanation: A MultiTSO control block in an address space has been overlaid.

Programmer Response: Check for a probable overlay. Save the dump, and contact UNICOM Systems, Inc.

BAD
Explanation: Internal error.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

Assembly MNOTEs

The MNOTEs below may appear in MultiTSO assemblies.

PCS300E INVALID LENGTH SPECIFIED n, MUST BE 1 TO 4
Explanation: The length parameter, n, specified in an @DFIELD macro was not between 1 and 4.

Programmer Response: Correct the macro.

PCS301E @SXMAP MUST PRECEDE THIS MACRO.
Explanation: A @SXMAP macro must precede the @DFIELD macros in PCSDFLD@. One was generated for you, but check to ensure that the user @DFIELD macros are after the @SXMAP macros already in the PCSDFLD@ assembly.

Programmer Response: Correct the @DFIELD macro placement.

PCS302E @DFIELD OUT OF ORDER, AFTER END INDICATOR
Explanation: A @DFIELD macro appeared after a null @DFIELD macro, indicating the end of the table. All @DFIELD macros must appear before the end of the table.

Programmer Response: Correct the @DFIELD macro placement.

PCS310I ASSEMBLY OF PCSVPSWD EXIT BYPASSED _ DEFAULT EXIT USED
Explanation: The OPTIONS parameter for the PCSVPSWD assembly specified NONE. The default password verification code will be used.

Programmer Response: If RACF, ACF2, or TOPSECRET is not installed at your site, this is a normal message. SYS1.UADS will be used for password verification for the LOCK command.

If you have a security system installed, rerun job PCSVPSWD and specify the appropriate security system in the OPTIONS parameter.

PCS311E INCORRECT OR UNKNOWN SECURITY SYSTEM SPECIFIED
Explanation: The OPTIONS parameter for the PCSVPSWD assembly specified a value other than NONE, ACF2, RACF, or TOPS. The default password verification program will be used.

Programmer Response: If you do not have a security system installed, you many ignore this message. The default password verification program will be used. It assumes passwords are stored in SYS1.UADS.

If RACF, ACF2, or TOPSECRET is installed at your site, you must run job PCSVPSWD. Code the appropriate value for your system.

If you have another security system installed, please call UNICOM Systems, Inc. Customer Services for assistance.

PCS350W NO IKTcccc MACRO WAS FOUND IN THE MACLIB
PCS351W CONCATENATION. IF YOU HAVE VTAM, CONSIDER
PCS352W THIS A FATAL ERROR. IF YOU ONLY HAVE TCAM
PCS353W IGNORE THESE WARNINGS.
Explanation: The TSO VTAM macro specified was not found in the SYSLIB concatenation during assembly of PCSDFLD@.

Programmer Response: Verify whether the macro specified is available at your site. Sites with only TCAM TSO may not have the IBM TSO VTAM macros.

Sites using TSO VTAM should have copies of the IBM TSO VTAM macros. If you plan to run MultiTSO with TSO VTAM, consider these messages as errors. Locate the missing TSO VTAM macros and reassemble PCSDFLD@ via the PCSDFLD@ job.

If you do not have TSO VTAM and only plan to run MultiTSO with TSO TCAM, ignore these messages.

PCS354W NO IKJEFLWA MACRO WAS FOUND IN THE MACLIB
PCS355W CONCATENATION. IF YOU HAVE TSO/E, CONSIDER
PCS356W THIS A FATAL ERROR. IF YOU ONLY HAVE TSO
PCS357W IGNORE THESE WARNINGS.
Explanation: The TSO IKJEFLWA macro specified was not found in the SYSLIB concatenation during assembly of PCSDFLD@.

Programmer Response: Verify whether the macro specified is available at your site. Sites with only non-TSO/E do not have the IBM IKJEFLWA macro. Sites using TSO/E should have it.

If you do not have TSO/E ignore these messages.

If you have TSO/E consider these messages as errors. Locate the missing IKJEFLWA macro and reassemble PCSDFLD@ via the PCSDFLD@ job.

PCS400E &APF MUST BE 'SWITCH' OR 'NOSWITCH'
Explanation: A @SWITCH macro specified an operand of APF that was neither SWITCH or NOSWITCH.

Programmer Response: Correct the @SWITCH macro.

Chapter 2
MultiTSO Use

Messages

Numbered Messages

PCS014I INCORRECT VERSION OF @USER MACRO USED

PCS014I IN PCSUSERS PROCEEDING WITHOUT PCSUSERS
Explanation: During LOGON processing for a MultiTSO user, the PCSUSERS table was found to be incorrect. It had not been created with the correct version of the @SUSER MACRO.

System Action: LOGON proceeds, but the user will not be restricted by the PCSUSERS table. The user will have the defaults set up by the PARMLIB(SESSIONS) dataset from PCSINIT.

Operator Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Reassemble PCSUSERS using the correct PIE MACLIBs. Member PCSUSERS in the PIE CNTL library has the JCL to assemble PCSUSERS and to link it correctly into PCSTABLE.

PCS501W PIE/NETWORK ACCESS NOT INITIALIZED
Explanation: A MultiTSO user could not find the PNATVINI module during logon to TSO.

System Action: LOGON proceeds, but the user will not be able to use PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: The PNATVINI load module must be available in SYS1.LPALIB, a link library, or a step library for MultiTSO users, too. Check the PIE/TSO Installation and Customization Guide for further information.

Unnumbered Messages

Some MultiTSO messages appear only on the MultiTSO menu. These have no message numbers, so they are sorted alphabetically. The messages will overlay line three on the MultiTSO menu. If they appear press ENTER to restore line three. If you need HELP for the command, press PF1.

ccccccccc has an invalid or missing operand
Explanation: A MultiTSO menu command was entered, but a required operand was missing or invalid.

System Action: The data entered will remain on the command line.

User Response: Correct the command.

ccccccccc has been executed
Explanation: A command executed with PIEEXEC has been executed.

System Action: None.

User Response: None.

ACCESS session non-cancelable - use logoff sequence to terminate it
Explanation: You tried to cancel a Network Access session, but the ACCESS start-up option is set to NOCANCEL. You cannot cancel Network Access sessions.

System Action: The session is not canceled.

User Response: Logoff the application being accessed as you would from a terminal without Network Access. Then retry the CANCEL command.

ACCESS session(s) non-cancelable - terminate them before CANCEL ALL
Explanation: You entered CANCEL ALL while Network Access sessions were active. the ACCESS start-up option is set to NOCANCEL. You cannot cancel Network Access sessions, so the CANCEL ALL command will not execute.

System Action: No sessions are canceled.

User Response: Logoff the applications being accessed as you would from a terminal without Network Access. Then retry the CANCEL ALL command.

Entered command string is unauthorized for this session
Explanation: The command string entered would have resulted in the IBM Session Manager being invoked incorrectly.

System Action: The entered data remains on the command line.

User Response: Correct the command.

Concurrent users exceeded - proceeding with TSO
Explanation: There are more users active on your PIE/TSO system than are allowed by your license agreement.

System Action: The logon continues without MultiTSO.

User Response: None.

Entered primary command is invalid
Explanation: A HELP request was entered on the HELP session command line, and the request is more than 8 characters.

System Action: The entered data remains on the command line.

User Response: Correct the command or member name. It must be from 1 to 8 characters long.

Help request is invalid
Explanation: A HELP request was entered from the MultiTSO menu, and the member supplied was invalid.

System Action: The HELP command is displayed on the command line, for correction.

User Response: Correct the command.

Insufficient user region available to start session
Explanation: There is not enough user region available to start another session. Too many applications are already active.

System Action: None.

User Response: Close another session and try again.

Programmer Response: If this message occurs frequently when only a small number of sessions are open, you can turn off the virtual storage check using the VSTOR start-up option.

Insufficient SWA/LSQA/AUK available to start session
Explanation: There is not enough high private storage available to start another session. Too many applications are already active.

System Action: None.

User Response: Close another session and try again. If you get this message frequently when only a small number of sessions are open, contact the person responsible for PIE/TSO.

Programmer Response: If this message occurs frequently when only a small number of sessions are open, you can try to prevent it by decreasing region size by 1 M increments. (Reducing region size provides more high private storage.) Or you can turn off the virtual storage check using the VSTOR start-up option.

Jump PFKEY changed
Explanation: A JMKey command was entered, and the Jump Key has been changed.

System Action: None.

User Response: None.

PIE/TSO MultiTSO and MVS levels are incompatible - inform system programmers
Explanation: During logon, PIE/TSO found that your the MVS level is not the same as your MultiTSO level.

System Action: The logon will continue as if PIE/TSO was not available. PCSEFT01 will XCTL to the IBM TMP specified.

User Response: Inform the systems programmer responsible for PIE/TSO.

Programmer Response: Check the level of the PCSDFLD@ module, and reassemble if required. For details, see the PIE/TSO Administrator Guide.

PIE/TSO MultiTSO encountered internal initialization error
Explanation: During logon MultiTSO encountered an internal error.

System Action: The logon will continue as if MultiTSO was not available. PCSEFT01 will XCTL to the IBM TMP specified.

User Response: Inform the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc.

PIE/TSO MultiTSO not initialized due to user exit request
Explanation: A PCSDFLD@ installation exit requested that MultiTSO not be initialized for this logon.

System Action: The logon will continue as if MultiTSO was not available. PCSEFT01 will XCTL to the IBM TMP specified.

User Response: Inform the systems programmer responsible for PIE/TSO.

Programmer Response: A PCSDFLD@ initialization exit has completed with a return code greater than 4.

PIE/TSO MultiTSO was unable to load PCSDFLD@
Explanation: During your logon the PCSDFLD@ module could not be loaded.

System Action: The logon will continue as if MultiTSO was not available. PCSEFT01 will XCTL to the IBM TMP specified.

User Response: Inform the systems programmer responsible for PIE/TSO.

Programmer Response: Ensure that the PCSDFLD@ assembly has completed properly.

Please terminate active sessions first or enter CANCEL ALL
Explanation: An END command was entered from the MultiTSO menu before all active sessions had been terminated.

System Action: None.

User Response: Terminate all active sessions, or use the CANCEL ALL command.

Return code nn from TPUT
Explanation: During HELP processing to a non-3270 terminal, an unknown TPUT return code was received.

System Action: None.

User Response: Retry the HELP command, and if the above message persists, contact the systems programmer responsible for PIE/TSO.

Programmer Response: Check the TPUT return code and if necessary contact UNICOM Systems, Inc.

Session already active - choose another session ID
Explanation: A START command was issued specifying both the session number and the TSO command line. The session number specified was already active.

System Action: The user will be placed in the MultiTSO menu with the entered START command on the command line.

User Response: Modify the command entered.

Session(s) unhidden
Explanation: An UNHIDE command was successfully issued.

System Action: None.

User Response: None.

Session(s) hidden
Explanation: A HIDE command was successfully issued.

System Action: None.

User Response: None.

Specify ID of active session to store notes
Explanation: A NOTE command was entered for a session that does not exist or no session ID was specified.

System Action: None.

User Response: Correct the command–specify an active session ID.

Specified user exit not found by PIE/TSO MultiTSO
Explanation: An installation specified PCSDFLD@ initialization exit could not be found during logon.

System Action: Logon will continue as if MultiTSO is not active. PCSEFT01 will XCTL to the IBM TMP specified.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Ensure all installation exits are available.

Switch character changed
Explanation: A SwChar command was entered and the switch character has been changed.

System Action: None.

User Response: None.

Switch has been disabled from this authorized program
Explanation: You are trying to switch out of an APF authorized program. However PIE/TSO start-up options have been set to prevent switching out of APF authorized programs.

System Action: None.

User Response: End out of the program to switch.

Switch PFKEY changed
Explanation: A SwKey command was entered and the switch key has been changed.

System Action: None.

User Response: None.

There has been an I/O error on ccccccc
Explanation: During HELP processing, an I/O error was encountered on DD name ccccccc. Other information concerning the I/O error will follow the DD name.

System Action: HELP processing will end.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Ensure that the largest blocksize is specified on the first dataset in the HELP concatenation.

There has been an I/O error on the Help file
Explanation: During HELP processing, an I/O error was encountered during FIND processing.

System Action: HELP processing will end.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Attempt to browse the HELP member involved.

There is no Help available for ccccccc
Explanation: The member you requested for HELP does not exist.

System Action: HELP processing will end.

User Response: Restart HELP with a valid member name or omit the member name and select it from the HELP menu.

There is not enough memory for Help
Explanation: During HELP processing, FIND returned an out of memory return code.

System Action: HELP processing will end.

User Response: Terminate some sessions and try again.

Unable to allocate Help information
Explanation: During HELP processing, the HELP dataset could not be allocated.

System Action: HELP processing will end.

User Response: Check that your procedure has a DD statement for the MultiTSO HELP dataset. HELP will attempt to find the installation specified DD name first, and then attempt to allocate the installation specified MultiTSO HELP dataset name.

Unable to find your Help information
Explanation: During HELP processing, the HELP command was unable to OPEN the HELP dataset.

System Action: HELP processing will end.

User Response: Check that your procedure has a valid MultiTSO HELP DD statement.

Unable to start a new session - limit reached
Explanation: A Start command was issued, but the limit on the number of sessions has been reached.

System Action: If the command was issued from TSO, the MultiTSO menu will be displayed.

User Response: Terminate a session if desired.

Virtual storage is unavailable for Help information
Explanation: The installation specified HELP buffer could not be obtained for your HELP request.

System Action: None.

User Response: Terminate some sessions and try again.

You are not authorized to use Asynchronous Switching
Explanation: The ASYS option must be set to YES either in the SESSIONS member or, for your user ID or terminal, in the PCSUSERS table.

System Action: None

User Response: To obtain authorization, contact the person responsible for maintaining PIE/TSO.

You are not authorized to use MultiTask
Explanation: The MTASK option must be set to YES either in the SESSIONS member or, for your user ID or terminal, in the PCSUSERS table.

System Action: None

User Response: To obtain authorization, contact the person responsible for maintaining PIE/TSO.

You are unauthorized to switch from the current environment
Explanation: You tried to switch from a session executing under an authorized Request Block (RB). MultiTSO will never allow this to happen. If you did switch at this point, you could lock up your address space.

System Action: None.

User Response: End the current command before switching out.

You can't cancel inactive session - choose another session
Explanation: The user tried to CANCEL a session that was not active.

System Action: None.

User Response: Enter command again specifying an active session ID.

Abend Completion Codes

The codes listed below are all user abend codes issued by MultiTSO.

100
Explanation: PCSEFT01 was not executed from the job step TCB. It was attached or otherwise entered from a lower level TCB.

Programmer Response: Do not attempt to execute the PCSEFT01 program other than as the job step TCB. Change the TSO logon procedure used.

110
Explanation: PCSEFT01 was not executed in a TSO address space.

Programmer Response: Do not attempt to execute the PCSEFT01 program other than in a TSO address space.

120
Explanation: PCSIPL was not able to GETMAIN the CSA or SQA required to initialize MultiTSO. This abend is always preceded by message PCS045E. If SQA was specified, the amount requested was less than 128 bytes.

If CSA was specified, the amount requested was less than 4K.

Programmer Response: Check to see if there are any problems with CSA or SQA. If not save the dump provided, and contact UNICOM Systems, Inc.

200
Explanation: Unable to attach the TMP or SECTMP specified. MultiTSO is active.

Programmer Response: Check the TMP or SECTMP specified in the PIE/TSO MultiTSO PARMLIB dataset and the logon procedure.

204
Explanation: Unable to XCTL to the TMP or SECTMP specified. MultiTSO is not active.

Programmer Response: Check the TMP or SECTMP specified in the MultiTSO PARMLIB dataset and the logon procedure.

220
Explanation: An internal error occurred attempting to load MultiTSO modules during PCSINIT. This abend is always preceded by message PCS046E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

300
Explanation: Unable to obtain the SALLOC lock during PCSIPL processing. This abend will always be accompanied by a PCS042E message.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

310
Explanation: Unable to free the SALLOC lock during PCSIPL processing.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

320
Explanation: An internal error occurred attempting to load MultiTSO modules during PCSINIT. This abend is always preceded by message PCS046E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

500
Explanation: There is an error in the SESSIONS dataset. MultiTSO will not initialize.

Programmer Response: Check the log for errors. It is possible that the password values are incorrect. Check the EXPDT, CPUID, and/or xxxCMP values are incorrect. Change them to the values supplied by UNICOM Systems, Inc. If the password has expired, please call your UNICOM Systems, Inc account manager.

600
Explanation: The CVT was dynamically modified before the execution of PCSINIT. This abend is always preceded by message PCS047E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

666
Explanation: Internal error.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

690
Explanation: The CVT was dynamically modified before the execution of PCSINIT. This abend is always preceded by message PCS048E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

700
Explanation: The PCSSVC01 load module from SYSLIB was invalid during PCSINIT. This abend is always preceded by message PCS049E.

Programmer Response: Check that the dataset specified on the SYSLIB DD statement, or APF parameter is valid. Save the dump provided, and contact UNICOM Systems, Inc.

710
Explanation: An internal error occurred during PCSINIT. This abend is always preceded by message PCS050E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

788
Explanation: In response to a user request made with the “ABENDPIE DUMP” command MultiTSO has taken a dump.

Programmer Response: None.

789
Explanation: In response to a user request made with the “ABENDPIE NODUMP” command, MultiTSO has abended without a dump.

Programmer Response: None.

800
Explanation: A MultiTSO attempt to process the PCSSWEQT table received a bad return code from the GQSCAN macro.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

900
Explanation: A MultiTSO read only control block in CSA has been overlaid. This abend will always be preceded by a PCS037E message or a PCS042E message and a PCS043E message.

Programmer Response: Check for a probable CSA overlay. Save the dump, and contact UNICOM Systems, Inc.

901, 902, 903, 904
Explanation: A MultiTSO control block in an address space has been overlaid.

Programmer Response: Check for a probable overlay. Save the dump, and contact UNICOM Systems, Inc.

BAD
Explanation: Internal error.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

Chapter 3
View

Numbered Message

PVT901A MODULE PVTTIOS NOT FOUND. INSTALLATION ERROR
Explanation: During initialization of a View session, the module named PVTTIOS could not be located.

System Action: View session is not started.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: The module must be placed in SYS1.LPALIB, the user's STEPLIB, or a LNKLST library. Check the PIE/TSO Installation and Customization Guide and verify that installation was done according to the instructions provided. For further information, consult UNICOM Systems, Inc.

Unnumbered Messages

Some View messages appear only on the View menu. These have no message numbers, so they are sorted alphabetically. The messages will overlay line three on the View menu. If they appear, press ENTER to restore line three. If you need HELP for a command, press PF1.

A PIE/VIEW screen has been sent by “userid”
Explanation: The userid specified has sent a screen image to you.

System Action: None.

User Response: To display the screen image sent, issue the View RECEIVE command. You may then edit, browse, save, or print the screen image received. You may also modify it immediately after issuing the RECEIVE command and issue another RECEIVE command to redisplay the modified image.

Programmer Response: None.

Entered primary command is invalid
Explanation: The command you entered in the command area is not a invalid View command or the name of an existing screen image member.

System Action: Data entered remains on command line.

User Response: Enter an existing member name or a valid command.

Programmer Response: None.

Invalid member name. re-enter member name
Explanation: When requesting a save for a screen image, the member name entered was too long or contained illegal characters.

System Action: None.

User Response: Enter a valid member name of 1-8 characters, containing only alphanumeric or national characters.

Programmer Response: None.

Invalid or inactive userid specified
Explanation: The userid specified for a SEND command either does not exist or does not currently have an active View session.

System Action: No screen image is sent or received.

User Response: Verify the userid and ensure that the user has an active View session.

Programmer Response: None.

Member exists. Press ENTER to save or enter new member name
Explanation: When attempting to save a screen image, if a member already exists with the name you have entered, that member will be replaced if you do not enter a new member name.

System Action: The member is not saved. View waits for your response.

User Response: Enter a new member name for the screen image you are about to save or press ENTER to overlay the existing member with the screen image you are saving. To cancel the save, either issue the RETURN command or clear the member-name field and issue the END command.

Programmer Response: None.

No screen received. A screen must be sent for RECEIVE function
Explanation: You are attempting to receive a screen image, but no image has been sent to you.

System Action: None.

User Response: Ask the appropriate user to send you the screen image you want to receive. You may also send yourself a screen image.

Programmer Response: None.

RECEIVE failed: Incompatible screen size from sender
Explanation: You have attempted to receive a screen image from a user whose terminal screen size (43 lines by 80 columns, for instance) is defined as larger than yours (24 lines by 80 columns, for instance).

System Action: The screen image is not received.

User Response: Either logon to a terminal with a screen size compatible with or larger than that of the screen image you are receiving or ask the sender to send you the screen image from a terminal whose screen size is the same size as or smaller than yours.

Programmer Response: None.

RECEIVE was unsuccessful
Explanation: You have entered the View RECEIVE command but an internal system error caused the request to fail.

System Action: The screen image is not received.

User Response: Ensure that you are in a View session and retry the RECEIVE command. If the error persists, contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc.

SAVE failed. Error during open for PIE/VIEW dataset
Explanation: During the screen image SAVE that you requested, an unrecoverable external error occurred while attempting to open the PIEVIEW dataset.

System Action: The screen image is not saved.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Check for I/O error, data check. If that check does not resolve the problem, contact UNICOM Systems, Inc.

SAVE failed. I/O error occurred during SAVE processing
Explanation: An unrecoverable I/O error occurred while attempting to save a screen image.

System Action: The screen image is not saved.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Check for I/O error, data check. If that check does not resolve the problem, contact UNICOM Systems, Inc.

SAVE failed. PIE/VIEW dataset is in use by another user/session
Explanation: During the screen image SAVE you requested, the View dataset was unavailable because another user was already using the dataset or because you are updating the member in another session of your own. Until the member is released, you may not save into it.

System Action: The screen image is not saved.

User Response: If your View dataset is widely shared, retry the save once or twice. If this fails, you must wait for the other user to complete the operation in progress on the dataset or ask the user to release the dataset.

Programmer Response: None.

SAVE was unsuccessful
Explanation: The screen image save that you requested could not be completed because there was insufficient space in the View dataset. The dataset is out of space and/or directory blocks.

This message is usually accompanied by a system D.3.7, B37, or E37 abend.

System Action: Abend D37, E37, or B37.

User Response: Get the dataset name from the IEC301I message notifying you of the system D37 abend. If you need assistance, your PIE/TSO administrator or systems programmer can help you reallocate or compress this dataset. Press ENTER after receiving this message to resume other activity.

Programmer Response: Help the user as requested to either reallocate the View dataset or delete member from it and compress it.

SEND was unsuccessful
Explanation: The send you requested could not be completed because the userid you specified is not a View user.

System Action: The screen image is not sent.

User Response: Verify the userid specified. Then wait till the user logs on or notify the user that you want to send a screen image and that he or she must start a View session in order to receive a screen image from you. Once the user is logged on, resend the screen image.

Programmer Response: None.

Specified image or member not found
Explanation: The member name entered does not exist in the View dataset or is invalid, or the command entered does not exist.

System Action: Data entered remains on command line.

User Response: Enter a valid View dataset member name or command.

Programmer Response: None.

VIEW screen is blank. Must capture screen image before SEND
Explanation: You have attempted to send a screen without first capturing its image.

System Action: None.

User Response: As the message indicates, first capture your screen image as indicated in the PIE/TSO User Guide. Then request the send again.

Programmer Response: None.

VIEW screen is blank. Must capture screen image before SHOW
Explanation: You have requested an edit without first capturing a screen image.

System Action: None.

User Response: As the message indicates, first capture your screen image as indicated in the PIE/TSO User Guide. Then request the edit.

Programmer Response: None.

Chapter 4
Network Access PNAIPL

Numbered Messages

PNA000I
Explanation: PNAIPL was executed without a SYSPRINT DD statement. Message PNA013E will also be issued, as well as the headings, PNA010I, PNA011I and PNA12I.

PNAIPL will list all parameter data set statements on the console using this message identifier.

System Action: None.

Operator Response: None.

Programmer Response: None.

PNA010I PIE/TSO NETWORK ACCESS RELEASE r.r.r PAGE nn
PNA011I ON CPU cpuid model
PNA012I AT hh:mm:ss ON yy.ddd
Explanation: PNAIPL was executed without a SYSPRINT DD statement. Message PNA013E was already issued. These are the heading lines for the output listing.

System Action: None.

Operator Response: None.

Programmer Response: None.

PNA013I UNABLE TO OPEN SYSPRINT DATASET
Explanation: The OPEN for the output data set failed.

System Action: PNAIPL will begin using the operator console to output all messages.

Operator Response: None.

Programmer Response: Check the allocation for the SYSPRINT data set.

PNA020E UNABLE TO OPEN SYSIN DATASET, TERMINATING
Explanation: PNAIPL was unable to OPEN the parameter data set.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Check the allocation for the SYSIN data set.

PNA021E NO PIE/TSO NA PARAMETERS FOUND, TERMINATING
Explanation: The parameter data set pointed to by the SYSIN DD statement was empty.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Check the allocation for the SYSIN data set.

PNA022E PIE/TSO NA REQUIRED PARAMETERS MISSING
Explanation: PNAIPL must find three valid parameter keywords in the parameter data set. They are the CPU serial number (CPUID) plus one or more CPU ID/password statements, expiration date (EXPDT) and the installation SVC number (SVC). One, or more of the above parameters were missing from the parameter data set.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Check that all required parameters are supplied.

PNA023E PNAIPL TERMINATED DUE TO PARAMETER ERRORS
Explanation: PNAIPL detected an error in the input parameter data set. More information will appear in the SYSPRINT output data set.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Check the output from PNAIPL and correct any errors flagged.

PNA024E EXPIRATION DATE OF yy.ddd HAS EXPIRED, PIE TERMINATING
Explanation: The license period for Network Access has expired. Do not attempt to tamper with the EXPDT parameter in the parameter data set. Any attempt to do so will result in Network Access becoming unusable.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Contact UNICOM Systems, Inc. to extend your license.

PNA025E PIE/TSO NA NOT LICENSED ON THIS CPU
Explanation: Network Access has not been licensed on this computer. Do not attempt to tamper with the CPUIDs in the parameter data set. Any attempt to do so will result in Network Access becoming unusable.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: If the primary CPU is unusable, you can substitute alternate CPUs on your license agreement, either temporarily or permanently. To do so, contact UNICOM Systems, Inc. to modify your license.

TSC PNA026E PNAIPL EXECUTED AS A JOB OR TSO COMMAND
Explanation: PNAIPL was executed as a batch job or from TSO. This is not allowed except for the ACT=TEST option.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: None.

Programmer Response: Do not attempt to execute PNAIPL except as a started task or with the TEST option.

PNA027E PNAIPL MUST BE AN AUTHORIZED PROGRAM
Explanation: PNAIPL was executed from a non-APF authorized library, or has been link edited without the AC=1 parameter.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Move PNAIPL to an APF authorized library, and ensure that it has been link edited with APF authorization.

PNA030E INVALID PARM cccc, IGNORED
Explanation: PNAIPL was executed with an invalid ACT parameter.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access. A TEST function will be performed.

Operator Response: Ensure that the ACT parameter specified in the START PNAIPL command is either TEST, REPLACE, STOP, or START. If the START command was issued using the default installation supplied ACT, contact the systems programmer responsible for PIE/TSO Network Access immediately. Issue am “S PNAIPL,ACT=x” with the desired ACT parameter.

Programmer Response: Ensure the default ACT keyword in the PNAIPL procedure is correct.

PNA031E PIE/TSO NA NOT ACTIVE, CAN'T STOP
Explanation: PNAIPL was executed with an ACT=STOP or REPLACE parameter, but PIE/TSO Network Access was not active.

System Action: PNAIPL will terminate without stopping or replacing PIE/TSO Network Access.

Operator Response: Restart PIE/TSO Network Access if desired by starting PNAIPL with an ACT=START.

Programmer Response: None.

PNA032E PIE/TSO SESSIONS NOT ACTIVE, CAN'T EXECUTE
Explanation: PNAIPL has found that PIE/TSO MultiTSO is not active. PIE/TSO MultiTSO must be active, (PIEIPL must have executed), before PIE/TSO Network Access can be started, stopped or replaced.

System Action: PNAIPL will terminate without starting, stopping or replacing PIE/TSO Network Access.

Operator Response: Make sure that PIE/TSO MultiTSO has been successfully initialized with a “S PIEIPL” command, before attempting to initialize PIE/TSO Network Access.

Programmer Response: None.

PNA034I PIE/TSO NA TSO INTERFACE TERMINATED
Explanation: In response to a STOP or REPLACE request, PNAIPL has terminated PIE/TSO Network Access.

System Action: PIE/TSO Network Access is terminated.

Operator Response: None.

Programmer Response: None.

PNA035I PIE/TSO NA NOT ACTIVE
Explanation: PNAIPL has been run with the TEST option. PIE/TSO Network Access is not currently active.

System Action: None.

Operator Response: None.

Programmer Response: None.

PNA036I PIE/TSO NA ACTIVE, PIE/TSO NA PNAPARMS AT xxxxxxxx
Explanation: PNAIPL has been run with the TEST option. PIE/TSO Network Access is active. The address of PIE/TSO Network Access parameters are at xxxxxxxx.

System Action: None.

Operator Response: None.

Programmer Response: None.

PNA040E UNABLE TO OPEN SYSLIB DATASET, TERMINATING
Explanation: PNAIPL has found that the SYSLIB dataset required for START and REPLACE could not be opened.

System Action: PNAIPL will terminate without initializing or replacing PIE/TSO Network Access.

Operator Response: If APF was entered with the “S PNAIPL” command ensure there was no typing error.

If there was no error, contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Check the SYSLIB DD statement in the PNAIPL procedure.

PNA041I PIE/TSO NA SUCCESSFULLY INITIALIZED
Explanation: In response to a START or REPLACE request, PNAIPL has initialized PIE/TSO Network Access.

System Action: PNAIPL will terminate. PIE/TSO Network Access is initialized.

Operator Response: None.

Programmer Response: None.

PNA042E ERRORS WITH SALLOC LOCK - PIE/TSO NA TERMINATING
Explanation: In response to a START or REPLACE request, PNAIPL was unable to obtain the SALLOC lock.

System Action: PNAIPL will abend with a user 300 completion code.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Contact UNICOM Systems, Inc. for assistance.

PNA043E CSA OVERLAY DETECTED AT @CAPARMS ADDRESS hhhhhhhh CONTENTS IS vvvvvvvv
Explanation: An attempt was made to START PIE/TSO Network Access while Network Access was already initialized. At this time a CSA overlay of vvvvvvvv at address hhhhhhhh was detected in a PIE/TSO Network Access read only control block.

System Action: PNAIPL will abend with a user 900 completion code.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Use the dump provided to check the CSA overlay.

PNA044W UNABLE TO START PIE/TSO NA - ALREADY ACTIVE
Explanation: An attempt was made to START PIE/TSO Network Access while it was already initialized.

System Action: PNAIPL will terminate.

Operator Response: None.

Programmer Response: None.

PNA045E SQA/CSA CRITICAL, REQUEST ABORTED
Explanation: An attempt was made to START PIE/TSO Network Access while there was less than 4K of SQA or CSA available.

System Action: PNAIPL will terminate with a user 120 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Correct the CSA or SQA shortage problem.

PNA046E LOAD UNSUCCESSFUL REASON CODE n - PIE/TSO NA TERMINATING
Explanation: An internal error occurred while attempting to load PIE/TSO Network Access modules during a START or REPLACE.

System Action: PNAIPL will terminate with a user 220 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc.

PNA047E CVT MODIFIED - PIE/TSO NA TERMINATING
Explanation: The MVS CVT was modified during PNAIPL execution of START or REPLACE.

System Action: PNAIPL will terminate with a user 600 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc.

PNA048E CVT MODIFIED DURING LOAD - PIE/TSO NA TERMINATING
Explanation: The MVS CVT was modified during PNAIPL execution of START or REPLACE.

System Action: PNAIPL will terminate with a user 690 abend.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Save the dump provided and contact UNICOM Systems, Inc.

PNA051E CAN'T FIND PNAPARM MODULE
Explanation: A required module, PNAPARM, was not in the dataset pointed to by the PNAIPL procedure SYSLIB DD statement during a START or REPLACE request.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: If APF was entered with the “S PNAIPL” command ensure there was no typing error.

If there was no error, contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Check the validity of the SYSLIB DD statement.

PNA052E BLDL ERROR ACCESSING MODULE PNAPARM RETURN CODE n REASON CODE y
Explanation: An error occurred during a BLDL for module PNAPARM. The return code n and reason code y are provided.

System Action: PNAIPL will terminate without initializing PIE/TSO Network Access.

Operator Response: Contact the systems programmer responsible for PIE/TSO Network Access immediately.

Programmer Response: Check the validity of the SYSLIB DD statement. The BLDL return codes and reason codes can be found in the IBM Data Management Services Guide.

PNA120I END OF PARAMETERS
Explanation: This message will appear only on the SYSPRINT dataset. The end of the parameter data set has been reached.

System Action: None.

Operator Response: None.

Programmer Response: None.

PNA121W EXPIRATION DATE OF yy.ddd WILL EXPIRE IN n DAYS
Explanation: The PIE/TSO Network Access license agreement will expire on yy.ddd. Do not attempt to modify the EXPDT parameter or PIE/TSO Network Access will become unusable.

System Action: None.

Operator Response: None.

Programmer Response: Contact UNICOM Systems, Inc. for a license extension.

PNA220E UNKNOWN KEYWORD
Explanation: This message will appear only on the SYSPRINT dataset. The previous statement contained a keyword not known to PNAIPL.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA221E *
Explanation: This message will appear only on the SYSPRINT dataset. The previous statement contained an error. The asterisk (*) flags the column where the error was detected.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA222E DUPLICATE PARAMETER, IGNORED
Explanation: This message will appear only on the SYSPRINT dataset. The previous statement contained a keyword already processed from the parameter data set. It will be ignored.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA223E INVALID APPLICATION TABLE NAME - cccccccc
Explanation: This message will appear only on the SYSPRINT dataset. The specified APPLS parameter is invalid for a load module.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA224E INVALID {CONTME/RESTME/SIGNAL} TIME - ccc
Explanation: This message will appear only on the SYSPRINT dataset. One of these three parameters was specified incorrectly. The invalid value will be indicated by ccc.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA225W CONTME OF ccc NOT BETWEEN 2 AND 180, DEFAULT USED
Explanation: This message will appear only on the SYSPRINT dataset. The CONTME parameter ccc is not in the range of 2 to 180 seconds.

System Action: PNAIPL will use the default of 5 seconds and continue to process input parameters.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA226E cccc INVALID EXPIRATION DATE FORMAT
Explanation: This message will appear only on the SYSPRINT dataset. The EXPDT parameter is not in the format yy.ddd. Do not tamper with the EXPDT parameter or PIE/TSO Network Access will become unusable.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA227E INVALID UNLOCK TIME - cccc
Explanation: This message will appear only on the SYSPRINT dataset. The UNLOCK parameter is invalid.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA228W UNLOCK OF ccc NOT BETWEEN 5 AND 100, DEFAULT USED
Explanation: This message will appear only on the SYSPRINT dataset. The UNLOCK parameter is out of range.

System Action: PNAIPL will continue to process input parameters, using a default unlock time of 10 seconds.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA229E INVALID USERS TABLE NAME - cccccccc
Explanation: This message will appear only on the SYSPRINT dataset. The USERS parameter is invalid for a load module.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

PNA234E INVALID INPUT CARD
Explanation: This message will appear only on the SYSPRINT dataset. An input statement did not begin with a keyword, asterisk or CPU serial number.

System Action: PNAIPL will continue to process input parameters, and then terminate.

Operator Response: None.

Programmer Response: Correct or delete the statement in error.

Abend Completion Codes

The codes listed below are all user abend codes issued by PIE/TSO Network Access.

120
Explanation: PNAIPL was not able to GETMAIN the CSA or SQA required to initialize PIE/TSO Network Access. This abend is always preceded by message PNA045E. If SQA was specified, the amount requested was less than 128 bytes. If CSA was specified, the amount requested was less than 4K.

Programmer Response: Check to see if there are any problems with CSA or SQA. If not, save the dump provided, and contact UNICOM Systems, Inc.

220
Explanation: An internal error occurred attempting to load PIE/TSO Network Access modules during PNAIPL. This abend is always preceded by message PNA046E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

300
Explanation: Unable to obtain the SALLOC lock during PNAIPL processing. This abend will always be accompanied by a PNA042E message.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

310
Explanation: Unable to free the SALLOC lock during PNAIPL processing.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

320
Explanation: An internal error occurred attempting to load PIE/TSO Network Access modules during PNAIPL. This abend is always preceded by message PNA046E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

432
Explanation: A PIE/TSO Network Access control block in an address space has been overlaid.

Programmer Response: Check for a probable overlay. Save the dump, and contact UNICOM Systems, Inc.

434
Explanation: An internal error has been detected.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

435
Explanation: An internal error has been detected.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

500
Explanation: The calculated password does not match the specified password. PIE/TSO Network Access will not initialize.

Programmer Response: The value specified for EXPDT or CPUID has been modified. Change them to the UNICOM Systems, Inc supplied values. If the password has expired, please call your UNICOM Systems, Inc account manager.

600
Explanation: The CVT was dynamically modified during the execution of PNAIPL. This abend is always preceded by message PNA047E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

666
Explanation: Internal error.

Programmer Response: Save the dump, and contact UNICOM Systems, Inc.

690
Explanation: The CVT was dynamically modified during the execution of PNAIPL. This abend is always preceded by message PNA048E.

Programmer Response: Save the dump provided, and contact UNICOM Systems, Inc.

949
Explanation: A PIE/TSO Network Access control block has been overlaid. Network Access is unable to continue.

Programmer Response: Retry the ACCESS command. Save the dump provided, and contact UNICOM Systems, Inc.

Assembly MNOTEs

The MNOTEs below may appear in PIE/TSO Network Access assemblies.

PNA701E DSECT MUST BE YES OR NO
Explanation: The DSECT keyword in an @AUSER macro was neither YES or NO.

Programmer Response: Correct the @AUSER macro.

PNA702E LENGTH OF NAME MUST BE BETWEEN 1 AND 7
Explanation: A @AUSER macro specified a NAME positional parameter that was omitted or too long.

Programmer Response: Correct the @AUSER macro.

PNA703E VTERMINAL POOL NAME MUST HAVE 4 BYTES
Explanation: A @AUSER macro specified a VTERMINAL parameter that was not 4 bytes long.

Programmer Response: Correct the @AUSER macro.

PNA704E DSECT MUST BE YES OR NO
Explanation: The DSECT keyword in an @APPL macro was neither YES or NO.

Programmer Response: Correct the @APPL macro.

PNA705E LENGTH OF NAME MUST BE BETWEEN 1 AND 8
Explanation: The length of the NAME keyword on an @APPL macro was not between one and eight characters.

Programmer Response: Correct the @APPL macro.

PNA706E LENGTH OF MOD2ENT MUST BE BETWEEN 1 AND 8
Explanation: The length of the MOD2ENT keyword on an @APPL macro was not between one and eight characters.

Programmer Response: Correct the @APPL macro.

PNA707E INVALID RESHOW KEY SPECIFICATION
Explanation: The RESHOW keyword specified was not a hexadecimal value or the words CLEAR, PA1, PA2 or PA3.

Programmer Response: Correct the @APPL macro.

PNA708E LOGMOD MUST BE ACCEPT OR NOACCEPT
Explanation: A @APPL macro specified an operand of LOGMOD that was neither ACCEPT or NOACCEPT.

Programmer Response: Correct the @APPL macro.

PNA710W TITLE MACRO MUST BE THE FIRST ENTRY IN THE TABLE
Explanation: A @ATITLE macro was encountered after the first @APPL macro. The @ATITLE macro must precede all @APPL macros.

Programmer Response: Correct the placement of the @ATITLE macro.

PNA711E USERID AND TERMINAL ID MUTUALLY EXCLUSIVE
Explanation: Both the USERID and TERMID operands were coded on an @APPL macro. Only one or the other may be coded on this macro.

Programmer Response: Remove either one of the operands. If you need to code for both situations, code multiple @APPL macros for the same application instead.

PNA712W USERID OR TERMINAL ID IS TOO LONG
Explanation: The value coded for USERID was longer than 7 bytes, or the value for TERMID was longer than 8 bytes.

Programmer Response: Correct the value coded.

PNA713E INVALID OR NULL VALUE SPECIFIED FOR ERRMSG
Explanation: The value coded for ERRMSG was not YES or NO.

Programmer Response: Correct the value coded.

PNA714E INVALID OR NULL VALUE SPECIFIED FOR CONTME
Explanation: The value coded for CONTME was not valid.

Programmer Response: Correct the value coded. Valid operands are between 20 and 1800.

PNA715E INVALID OR NULL VALUE SPECIFIED FOR UNLOCK
Explanation: The value coded for UNLOCK was not valid.

Programmer Response: Correct the value coded. Valid operands are between 50 and 1000.

PNA716E VT2POOL MUST BE CODED
Explanation: The VT2POOL operand was not coded on the @AUSER macro. This is a required entry.

Programmer Response: Code a VT2POOL operand.

Chapter 5
Network Access Use

Numbered Messages

PNA501E - YOU HAVE NOT SELECTED AN APPLICATION
Explanation: You exited PIE/TSO Network Access menu without selecting an application.

System Action: The ACCESS command will terminate.

User Response: None.

Programmer Response: None.

PNA502E - PIE/TSO NETWORK ACCESS IS NOT ACTIVE FOR YOUR TSO ID
Explanation: You entered an ACCESS command, but PIE/TSO Network Access is inactive for your userid.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Start PIE/TSO Network Access and allow the user to use it.

PNA503E - SESSION MUST BE INITIATED VIA TSO COMMAND
Explanation: You invoked the ACCESS command as a program.

System Action: The ACCESS command will terminate.

User Response: Invoke PIE/TSO Network Access with the TSO ACCESS command.

Programmer Response: None.

PNA504I - NETWORK ACCESS SESSION HAS TERMINATED
Explanation: A user or system terminates a PIE/TSO Network Access session.

System Action: The PIE/TSO Network Access session will terminate.

User Response: None.

Programmer Response: None.

PNA505E - TERMINAL IN USE IS NOT A DISPLAY TERMINAL
Explanation: You have invoked the ACCESS command from a non-display terminal.

System Action: The ACCESS command will terminate.

User Response: Invoke PIE/TSO Network Access from a display terminal.

Programmer Response: None.

PNA506W - MAX AUTHORIZED NETWORK ACCESS SESSIONS ACTIVE
Explanation: You have invoked the ACCESS command, but the maximum installation authorized number of PIE/TSO Network Access sessions are already active for your userid.

System Action: The ACCESS command will terminate.

User Response: Terminate one currently active PIE/TSO Network Access session and try again.

Programmer Response: None.

PNA507E - NETWORK ACCESS IS NOT ACTIVE FOR YOUR userid
Explanation: You entered the ACCESS command, but PIE/TSO Network Access is inactive for your userid.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Start PIE/TSO Network Access and allow the user to use it.

PNA508E - DOWN LEVEL OF @ATVWA DSECT - REASSEMBLE
Explanation: UNICOM Systems, Inc. internal message.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately.

PNA509E - OLD SESSION HAS NOT TERMINATED - END SESSION
Explanation: You entered the ACCESS command, but a previous PIE/TSO Network Access session was not terminated properly.

System Action: The ACCESS command will terminate.

User Response: Enter LOGOFF to terminate current PIE TSO session and then try again.

Programmer Response: None.

PNA510W - YOU ARE UNAUTHORIZED TO ACCESS AN APPLICATION
Explanation: You entered the ACCESS command and selected an application which you are unauthorized to access.

System Action: The ACCESS command will terminate.

User Response: Select only applications your installation allows you to access.

Programmer Response: None.

PNA512W - PRESS ENTER TO CONTINUE OR “END” TO CANCEL ACCESS
Explanation: A PIE/TSO Network Access session was hung and you entered an attention interrupt (PA1).

System Action: The Network Access selection menu will be displayed. Waits for your response.

User Response: To continue the command, press ENTER. To stop execution, enter END.

Programmer Response: None.

PNA520W - THERE ARE NO APPLICATIONS FOR YOU TO ACCESS
Explanation: The ACCESS command cannot start. Your installation does not have a table of applications to access.

System Action: The ACCESS command will terminate.

User Response: Contact your systems programmer.

Programmer Response: Generate valid table of VTAM/TCAM applications for PIE users to access.

PNA521W - cccccccc IS UNKNOWN TO PIE/TSO NETWORK ACCESS
Explanation: You specified an unknown or invalid application name cccccccc for PIE/TSO Network Access to access.

System Action: The ACCESS command will display the PIE/TSO Network Access selection menu.

User Response: Select a valid application to access by moving cursor to the line with your application and typing “S” in first column.

Programmer Response: None.

PNA550E - PIE/TSO NETWORK ACCESS INACTIVE FOR YOU
Explanation: You entered an ACCESS command, but PIE/TSO Network Access is inactive for your userid.

System Action: ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Start PIE/TSO Network Access and allow the user to use it.

PNA551E - UNABLE TO FIND PIE/TSO NETWORK ACCESS MODULES
Explanation: You entered the ACCESS command, but PIE/TSO Network Access could not complete initialization.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Make sure that PIE/TSO Network Access load modules are in a link list library or in SYS1.LPALIB or in the user's STEPLIB.

PNA750A - PLU SNA PROTOCOL VIOLATION - BRACKET STATE
Explanation: PLU Error. The PLU appears not to be following SNA rules.

System Action: The ACCESS command will try to reinitialize current bracket state. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: The application being accessed has violated SNA protocol. To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA752A - PLU SNA PROTOCOL VIOLATION - CHAIN STATE
Explanation: PLU Error. The PLU appears not to be following SNA rules.

System Action: The ACCESS command will attempt to reinitialize the current chain state. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: The application being accessed has violated SNA protocol. To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA753E - UNKNOWN BRACKET STATE - LOGIC ERROR
Explanation: Internal Error.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA754E - UNKNOWN CHAIN STATE - LOGIC ERROR
Explanation: Internal Error.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA755A - PLU SNA PROTOCOL VIOLATION - (DR+FIC/MIC) PLU ERROR. THE PLU APPEARS NOT TO BE FOLLOWING SNA RULES.
Explanation: A DR request accompanied a FIC or MIC. This is not valid; DR can only accompany OIC or LIC.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA756E - UNKNOWN TERM STATE - PROBABLE LOGIC ERROR
Explanation: Internal Error.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA758E - UNEXPECTED RESPONSE RECEIVED
Explanation: PLU Error. The PLU appears not to be following SNA rules.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA759E - BB RECEIVED WITH NON-DATA REQUEST
Explanation: PLU Error. The PLU appears not to be following SNA rules.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA760E - UNEXPECTED RU TYPE RECEIVED IN RECV EXIT
Explanation: Unexpected entry type found in RECEIVE EXIT.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA761E - PENDING BB NOT RECEIVED AS EXPECTED
Explanation: PLU Error. The PLU appears not to be following SNA rules.

System Action: The ACCESS command will attempt to reinitialize. It will wait for user input or application output.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: To determine the cause of the error, run a VTAM BUF and IO trace of the physical terminal and the virtual terminal. Use the MultiTSO DISPLAY command to determine the virtual terminal being used. Recreate the error, and contact UNICOM Systems, Inc. with the trace.

PNA762E - SEND SHUTC FAILED
Explanation: A SEND SHUTC macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, contact UNICOM Systems, Inc.

PNA802A - GENCB FOR EXLST FAILED
Explanation: The GENCB for dynamic EXLST storage failed. Equivalent to an MVS abend 80A. Region size too small.

System Action: The ACCESS command will terminate.

User Response: There is not enough free storage in your region to initiate an ACCESS command. Terminate other sessions and commands, then try again.

If the problem persists contact your systems programmer.

Programmer Response: Increase the region size.

PNA803A - GENCB FOR ACB FAILED
Explanation: The GENCB for dynamic ACB storage failed. Equivalent to an MVS abend 80A. Region size too small.

System Action: The ACCESS command will terminate.

User Response: There is not enough free storage in your region to initiate an ACCESS command. Terminate other sessions and commands, then try again.

If the problem persists contact your systems programmer.

Programmer Response: Increase the region size.

PNA804A - GENCB FOR RPL FAILED
Explanation: The GENCB for dynamic RPL storage failed.

System Action: The ACCESS command will terminate.

User Response: There is not enough free storage in your region to initiate an ACCESS command. Terminate other sessions and commands, then try again.

If the problem persists contact your systems programmer.

Programmer Response: Equivalent to an MVS abend 80A. Region size too small.

PNA805A - GENCB FOR NIB FAILED
Explanation: The GENCB for dynamic NIB storage failed. Equivalent to an MVS abend 80A. Region size too small.

System Action: The ACCESS command will terminate.

User Response: There is not enough free storage in your region to initiate an ACCESS command. Terminate other sessions and commands, then try again.

If the problem persists contact your systems programmer.

Programmer Response: Increase the region size.

PNA806E - MODCB FOR NIB FAILED
Explanation: A MODCB macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, save the dump provided, and contact UNICOM Systems, Inc.

PNA807A - OPEN FOR VIRTUAL TERMINAL cccccccc FAILED; CODE=nn
Explanation: The OPEN ACB macro for VTAM VIRTUAL TERMINAL cccccccc failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: The VTAM error code for OPEN is reported as CODE=nn. See the VTAM MACRO description for OPEN, which describes the codes in detail. Correct the condition and retry.

PNA808E - SETLOGON FAILED
Explanation: The SETLOGON for the PIE/TSO Network Access ACB failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, contact UNICOM Systems, Inc.

PNA809E - INQUIRE FAILED
Explanation: An INQUIRE APPSTAT macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, contact UNICOM Systems, Inc.

PNA810I - APPL NAME cccccccc IS UNKNOWN TO VTAM
Explanation: The PLU APPL name which you are trying to ACCESS is defined to PIE/TSO Network Access, but is unknown to VTAM.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Correct your PIE/TSO Network Access APPL statements, or define the requested APPL to VTAM and then vary the new node active.

PNA811I - MODEENT? xxxxxxxx FOR cccccccc IS UNKNOWN TO VTAM
Explanation: The Mode Entry Table named xxxxxxx for virtual terminal cccccccc is not defined to VTAM.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Be sure the APPL statements in the PIE member of VTAMLST contain a MODETAB operand. The entries coded for MODnENT in PNAAPPLS must be in the MODETAB load module. DO NOT CODE a DLOGMOD operand on the APPL statement.

PNA812E - SESSION INITIATION REQUEST FAILED
Explanation: A REQSESS macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid REQSESS RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, save the dump provided, and contact UNICOM Systems, Inc.

PNA813E - SHOWCB FAILED
Explanation: A SHOWCB macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, save the dump provided, and contact UNICOM Systems, Inc.

PNA814I - cccccccc IS INACTIVE
Explanation: PLU application cccccccc is inactive.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Activate the required application (cccccccc), according to site policy.

PNA815I - cccccccc IS NOT ACCEPTING LOGONS
Explanation: PLU application cccccccc is not accepting logons at this time.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Determine why the PLU is not accepting logons, and correct the situation according to site policy. As an example, if the USERMAX for TSO has been set to 0, any user attempting to ACCESS TSO will receive this message.

PNA816I - CLOSE FOR VIRTUAL TERMINAL cccccccc FAILED; CODE=nn
Explanation: The CLOSE ACB for VTAM VIRTUAL TERMINAL cccccccc failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: The VTAM error code for CLOSE is reported as CODE=nn. See the VTAM MACRO description for CLOSE, which describes the codes in detail. Correct the condition and retry.

PNA817I - VIRTUAL TERMINAL HAS BEEN POWERED OFF
Explanation: a POWER OFF command has been issued, and the virtual terminal acts as though power were lost.

System Action: The ACCESS command will terminate.

User Response: Act as you would when a real terminal is powered off, or contact the systems programmer responsible for PIE/TSO.

For example, if power was turned off on a real 3270 running TSO, TSO would await a LOGON RECONNECT. If power was turned off while running CICS, CICS would drop the terminal.

Programmer Response: Inform the user of the consequences of powering off the affected PLU.

PNA818E - POST CODE FM TSO LAYER INVALID FOR ANY ECB
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Save the dump provided. Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA819E - TERMINAL STATE LOGIC ERROR
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Save the dump provided. Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA821E - POST CODE INVALID FOR @VECBRA ECB
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA822E - UNEXPECTED POST CODE SEQUENCE FROM TSO LAYER
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA823E - POST CODE INVALID FOR @AVECBU ECB
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA824E - POST CODE INVALID FOR @AVECBT ECB
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Save the dump provided. Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA827E - SEND FAILED
Explanation: A SEND macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, save the dump provided, and contact UNICOM Systems, Inc..

PNA828E - RECEIVE FAILED
Explanation: A RECEIVE macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, save the dump provided, and contact UNICOM Systems, Inc.

PNA829E - SNA SENSE CODE INVALID FOR @AVECBU ECB
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Save the dump provided. Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA830E - SNA SENSE CODE INVALID FOR @AVECBT ECB
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Save the dump provided. Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA831E - SNA SENSE CODE INVALID - LOGIC ERROR
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Save the dump provided. Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA832I - xxxxxxxx IS NOT AN ACCESSIBLE APPLICATION
Explanation: VTAM has indicated the xxxxxxxx application cannot be accessed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Verify xxxxxxxx is an active application defined to VTAM.

PNA833E - +RSP REQUIRED FOR UNKNOWN REQ TYPE - LOGIC ERROR
Explanation: Internal Error. Unknown or unsupported req type received by access.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA834I - NO VIRTUAL TERMINALS AVAILABLE - (POOL xxxx) - TRY LATER
Explanation: Too many users are attempting to use PIE/TSO Network Access. No PIE virtual terminals were available in pool xxxx. Therefore ACCESS was unable to connect to the application requested.

System Action: The ACCESS command will terminate.

User Response: Try again later. If the condition persists contact the systems programmer responsible for PIE/TSO.

Programmer Response: If necessary add more PIEnnnnn APPL statements to a new or existing PIE Application Major Node in SYS1.VTAMLST. If necessary segregate users by terminal pool.

PNA835I - MAXIMUM NETWORK ACCESS USERS LOGGED ON (POOL xxxxyyyy) - TRY LATER
Explanation: The maximum number of users are logged on using PIE/TSO Network Access virtual terminal pool xxxxyyyy.

System Action: The ACCESS command will terminate.

User Response: Try again later. If the condition persists, contact the systems programmer responsible for PIE/TSO.

Programmer Response: A VTAM nodename beginning with xxxx (and ending with yyyy) is unavailable. The pool of terminals (xxxx) may be exhausted or the APPL name xxxxyyyy has be varied inactive.

In the first case, increase the number of terminals in the pool if you wish. In the second case find out why the virtual terminal was varied inactive, and reactivate if you wish.

PNA836E - SLU TO PLU INBOUND CHAINING LOGIC ERROR
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA848E - @ATVWA DSECT IS DOWNLEVEL - RE-ASSEMBLE VTAM LAYER THEN RE-LOGON
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA850E - OPNSEC FAILED
Explanation: An OPNSEC macro failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Internal VTAM error or overlaid RPL or RPL pointer. Check first for a VTAM problem. If you don't find any, save the dump provided, and contact UNICOM Systems, Inc.

PNA852A - SESSION NOT BOUND - UNACCEPTABLE BIND IMAGE
Explanation: The bind image from the PLU is not acceptable for establishing an LU.2 session with PIE/TSO Network Access.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Correct the bind image. Valid bind images for PIE/TSO Network Access can be found in member M3278X of the PIE Sample Library. If the PLU ignores the suggested bind image by using NIB BNDAREA, contact UNICOM Systems, Inc. Determine this by comparing the suggested image sent to the PLU with the actual one received from the PLU. (You will need a buffer trace of the virtual terminal.)

PNA853E - POST CODE INVALID FOR @AVECBI ECB
Explanation: Internal Error.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA854E - SDT NOT RECEIVED AFTER INIT/CLEAR/BIND
Explanation: An SNA SDT was expected from the PLU, but did not arrive.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA855E - UNKNOWN REQUEST TYPE ARRIVED AT SCIP EXIT
Explanation: A previously undefined SNA request type has arrived at the SCIP exit, and as such, the SCIP exit is unprepared to handle this situation.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA856A - SESSION NOT BOUND - PSERVIC SCREEN SIZE(S)
INVALID FOR LU.2
Explanation: The bind image sent from the PLU to the SLU virtual terminal is not acceptable.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Correct the bind image. Valid bind images for PIE/TSO Network Access can be found in member M3278X of the PIE Sample Library.

PNA901E - UNKNOWN FDBK CODE IN SYNAD EXIT
Explanation: A physical or special case error has occurred and cannot be handled.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

PNA902I - OPERATOR VARIED VIRTUAL TERMINAL INACTIVE
Explanation: A computer operator has issued a “VARY NET,INACT,ID=xxxxxxxx” for your virtual terminal.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Determine the reason for the operator action.

PNA903A - DATA RECEIVED > THAN MAX RU SIZE SPECIFIED IN BIND
Explanation: Data has arrived at the SLU virtual terminal that is longer than the maximum length agreed to by the PLU and SLU at bind time.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA930E - UNKNOWN/RU TYPE ARRIVED AT NSEXIT
Explanation: A previously undefined SNA request type has arrived at the NSEXIT, and as such, the NSEXIT exit is unprepared to handle this situation.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA931E - UNEXPECTED NSPE RU FORMAT AT NSEXIT
Explanation: A previously undefined SNA NSPE RU FORMAT has arrived at the NSEXIT. The NSEXIT exit is unprepared to handle this situation.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Verify that the virtual terminal being used has been defined to the application being ACCESSED. The application may be rejecting the logon request because the virtual terminal is undefined.

If the virtual terminal is defined, be sure it is in service to the application being accessed. Use the application master terminal transaction to do that.

CEMT or CSMT is the transaction to use for CICS.

Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO.

If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA932E - UNEXPECTED NOTIFY RU FORMAT AT NSEXIT
Explanation: A previously undefined SNA NOTIFY RU FORMAT has arrived at the NSEXIT. The NSEXIT exit is unprepared to handle this situation.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: The virtual terminal may be out of service to the application (CICS or IMS, etc.). Verify the application has the virtual terminal in an operational status.

Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a dump and a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA933I - LOGON UNABLE TO COMPLETE
Explanation: A logon request was sent from the SLU to the PLU, but after acceptance by VTAM, the PLU either rejected the request, or became unavailable.

System Action: The ACCESS command will terminate.

User Response: Try again. If the failure persists, contact the systems programmer responsible for PIE/TSO.

Programmer Response: Verify that the requested PLU is up and available for logons. Verify that the virtual terminal attempting logon is defined to the application (i.e., for CICS the TCT must be updated as described in the PIE/TSO Installation and Customization Guide). The terminal must be in service to the application.

Display the virtual terminal from VTAM. The display should say the virtual terminal is an APPL in CONCT status. If the status is ACTIVE with no sessions, vary the APPL inactive and active again.

If you feel this error might be related to PIE/TSO Network Access, try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a dump and a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA934I - xxxxxxxx REJECTED YOUR LOGON REQUEST
Explanation: The PLU named xxxxxxxx rejected your logon request.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: If you feel this might be related to PIE/TSO Network Access, try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with the dump and a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO.

If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA940E - UNEXPECTED EXPEDITED FLOW RU RECEIVED IN DFASY EXIT
Explanation: A previously undefined SNA EXPEDITED FLOW RU has arrived at DFASY exit. The DFASY exit is unprepared to handle this situation.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a dump and a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA941E - SEND SHUTC FAILED
Explanation: An attempt by the virtual terminal to forward a Shutdown Complete to the PLU has failed.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Try the failing sequence on a real SNA.LU2 3270. If the failure disappears, contact UNICOM Systems, Inc. with a dump and a buffer and I/O trace of the virtual terminal encountering the problem. We strongly recommend that you cancel the TSO user with a dump after the user receives this message, and before the user executes any other function under TSO. If this is possible, write down the session ID where the message was issued, and keep this ID with the dump. The dump contains valuable trace information regarding the cause of this problem.

PNA950E - UNKNOWN FDBK CODE IN LERAD EXIT
Explanation: A logical VTAM error has occurred and cannot be handled.

System Action: The ACCESS command will terminate.

User Response: Contact the systems programmer responsible for PIE/TSO.

Programmer Response: Contact UNICOM Systems, Inc. immediately. This error should never occur.

Unnumbered Messages

Some PIE/TSO Network Access messages appear only on the Application menu. These have no message numbers, so they are sorted alphabetically. The messages will overlay line three on the Application menu. If they appear, press ENTER to restore line three. If you need HELP, press PF1.

Help request is invalid.
Explanation: A HELP request was entered from the PIE/TSO Network Access menu, and the name supplied was invalid.

System Action: The HELP command is displayed on the command line for correction.

User Response: Correct the command and continue.

Programmer Response: None.

Session terminated due to attention interrupt.
Explanation: The user replied END in response to message PNA512W.

System Action: None.

User Response: Select a valid application to access by moving the cursor to the line with your application and typing “S” in the first column or enter one of the valid commands.

Programmer Response: None.

Unknown command for PIE/TSO Network Access.
Explanation: You entered an invalid command on PIE/TSO Network Access menu.

System Action: The PIE/TSO Network Access Application Selection menu will be displayed.

User Response: Select a valid application to access by moving the cursor to the line with your application and typing “S” in the first column or enter one of valid commands.

Programmer Response: None.

cccccccc has invalid or missing operand.
Explanation: You entered a valid command on PIE/TSO Network Access Application Selection menu, but specified an invalid operand or no operand when it was needed.

System Action: PIE/TSO Network Access will redisplay the Application Selection menu.

User Response: Select a valid application to access by moving the cursor to the line with your application and typing “S” in the first column or enter a valid command with valid operand(s).

Programmer Response: None.

Index

R

S

T

U

V

Y

TM

PIE/TSO
Master Index

Release 3.2.1

UNICOM
SYSTEMS, INC.

PTSOIG321-01

Printed: October 2002

This manual applies to PIE/TSO release 3.2.1 and to all subsequent releases of the product until otherwise indicated by new editions or updates to this publication.

UNICOM Systems, Inc.
Unicom Plaza - Suite 310
15535 San Fernando Mission Blvd.

Index

The PIE/TSO Master Index contains the index entries for all PIE/TSO manuals, except the Messages and Codes manual. Prefixes on the page numbers refer you to the appropriate manual. The prefixes are:

- A: Administrator Guide
- C: Command Reference
- I: Installation and Customization Guide
- R: Release Notes
- U: User Guide

Symbols

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z