INSTALLATION GUIDE

PLATO LESSON DELIVERY AND AUTHORING 1

RELEASE 34.2 - NOS 2.3 LEVEL 617

RELEASE 34.2 - NOS 2.4.1 LEVEL 630

Table of Contents

1   Preface

SECTION 1 - INTRODUCTION

1.1   Introduction

This manual describes how to install the "PLATO Lesson Delivery and Authoring System" application on your operating system.

If the PLATO application media you ordered requires an upgrade of your operating system, we assume that that upgrade has already been installed. This guide will only discuss the PLATO application installation.

This manual has been written for system analysts and Engineering Services personnel who have a working knowledge of the hardware and software involved, including information in the operating system reference manual, the operating system operator's guide and the operating system analysis handbook.

The Operations Guide for the PLATO application contains operations information. The installing analyst should read the "PLATO Overview" section of the Operations Guide before starting the installation.

The Configuration Handbook for the PLATO application contains information on setting up the configuration parameters which are referenced in this manual.

The PLATO Software Release Bulletin for a particular release contains last-minute information about installations. That document should be read before beginning the installation. Any conflicts between information contained in the SRB and the Installation Guide should be resolved in favor of the SRB.

1.2   Operating System Compatibility

Since the PLATO application is delivered as binaries which
have been assembled for a specific level of the operating system,
you must make sure that the PLATO Release you are installing
is compatible with your operating system.

Release 33 of the PLATO application software was released
only under NOS 2.3 level 617.

Release 34.1 of the PLATO application software was released
only under NOS 2.4.1 level 630.

Release 34.2 of the PLATO application software was released
under both NOS 2.3 level 617 and NOS 2.4.1 level 630. This
required that there be two different versions of the PLATO
Release 34.2 materials. These two different versions are
assembled for a specific NOS level and are not interchangable.
That is, the PLATO Release 34.2 for NOS 2.3 level 617 will
not run on NOS 2.4.1 level 630 and vice versa. If you are
running PLATO Release 34.2 under NOS 2.3 level 617 and wish
to upgrade your operating system, but continue to use the
same PLATO Release, you must reorder the installation materials
and reinstall the PLATO application.

Beginning with PLATO Release 35, each new PLATO Release will
run under the most current level and the two most recent past
levels of the operating system.

1.2.1   Operating system changes

The following section describes changes in the past several
NOS releases which have required changes to the operation
or installation of the PLATO application. This is included
to inform analysts performing an upgrade installation or
reinstallation of the PLATO application of important changes
in operational procedures.

The following information highlights those NOS changes
which may have the greatest impact on the PLATO application.
Rather than duplicate other documentation on NOS, it
describes areas of significance and cites references for
further information.

The most intorastive documents addressing NOS changes
are the following:

> NOS Software Release Bulletin (SRB)
> NOS Feature Notes
> NOS V2 Installation Handbook (IHB)
> NOS V2 Operations Handbook (OHB)
> NOS V2 Analysis Handbook (AHB)

CHANGES FOR NOS 2.3 LEVEL 617

> * Service Class Validation for System Origin Jobs

> The service class validation is now checked for jobs routed
> from system origin jobs. Due to the restrictions that are
> now properly enforced, jobs which used to run may now abort
> with "INCORRECT SERVICE CLASS". The effect of this change
> on the PLATO application is seen as a "submitx error"
> when attempting to submit system origin jobs from PLATO
> lessons. To fix this problem, the VM parameter in the
> user's NOS user name must be updated to allow submitting
> jobs with system service class. This change must also be
> made to the user names specified by the "subun" and "prtun"
> PLATO configuration file entries before attempting to
> bring up the PLATO application.

CHANGES FOR NOS 2.4.1 LEVEL 630

> There were no operating system changes which required changes
> in the operation of the PLATO application for this level.

1.3   Organization

This Installation Guide is organized as follows:

1.   Introduction

2.   New system installation procedures.
      Procedures to be used when initially installing
      the PLATO Lesson Delivery and Authoring System.

3.   Optional installation procedures.
      Procedures to be used to install optional features
      of the PLATO application.

4.   Add-on Product installation procedures.
      Procedures to be used to install or upgrade optional,
      separately-ordered PLATO products.

5.   Update installation procedures.
      Procedures to be used to upgrade the PLATO Lesson
      Delivery and Authoring System from one full release
      to another (e.g., from Release 33 to Release 34).

6.   "Mini-release" installation procedures.
      Procedures to be used to upgrade the PLATO Lesson
      Delivery and Authoring System to correct critical
      problems (e.g., from Release 34.1 to Release 34.2).

7.   Reinstallation procedures.
      Procedures to be used to reinstall the same PLATO
      Release on a new operating system level.

8.   PLATO Bill of Materials.
      A description of the components of the PLATO Lesson
      Delivery and Authoring System.

The following convention is used in describing computer
console commands.

      - upper-case letters indicate terms to be entered
        exactly as given

      - lower case letters indicate terms which vary from
        system to system, and which should be replaced with
        values applicable to your system.

1.4   Installation Materials

As part of your installation package, you will receive two
copies of the Operations Guide and the Configuration Hand-
book. You will also receive the following:

TAPE VSN	FILE No.	CONTENTS

PLAT1A	1	Installation procedures
		   INSTALL
		   UPGRADE
		   LOAD4F
		   BKSPRU

	2	Release binaries for the PLATO application
	3	Release submit file (PLATOD)
	4	Release configuration file
	5	Release procedure MFNX
	6	Release procedure MFDX
	7	Modifications to NOS to support the PLATO application
	8	Modifications to NAM to support the PLATO application
	9	Modifications to CCP to support the PLATO application
	1C	LIBDECK entries for PLATO application
	11	Terminal resident load files
	12	PLATO load procedures
	13	Empty file
PLAT2A	1	System lesson master file (part 1)
	2	Empty file
PLAT2B	1	System lesson master file (part 2)
	2	Empty file
PLAT2C	1	New installation master file "newins"
	2	Empty file (Sent to new installations only, not to sites updating existing software)

1.5   Where to Start

Check to make sure that you have all the materials you will
need for the installation. Read the PLATJ Software Release
Bulletin. Then:

a) If you are performing a new system installation, begin
with the "New System Installation Procedures" section.

b) If you are performing an upgrade installation from one
PLATJ Release to another (for example, from R33 to R34),
begin with the "Update Installation Procedures" section.

c) If you are performing a "mini-release" installation (for
example, from R34.1 to R34.2), begin with the "Mini-Release
Installation Procedures" section.

d) If you are reinstalling the same PLATJ Release on a new
operating system level, begin with the "Reinstallation
Procedures" section.

2   New System Installation Procedures

SECTION 2 - NEW SYSTEM INSTALLATION PROCEDURES

2.1   Introduction

This section describes how to install the PLATO application
for the first time.

The following information will be asked of you during
the installation procedures:

- Your system routing identifier (RID).

  This is a unique identifier which has been assigned to
  your system by Control Data Software Manufacturing and
  Distribution. You will find your RID on the CDC tape
  label affixed to each of your tapes. The RID will be
  the last three characters of the "Customer Name"
  section of the tape. For example:

        CUSTOMER NAME   CFN NUMBER
        SMITH COMPANY   9315-00042222

  In this example, the system RID is "MNA".

- Your computer hardware configuration.

  This information will be used to determine installation
  procedures.

For future reference, keep this information with your PLATO
Operations guide. It will also be required should you have
to call or write for help with the PLATO application and
when you order updates for the PLATO application or add-on
products.

The following is a list of the steps to be taken to install
the PLATO application. Each of these steps is discussed in
detail in the following sections.

1. Determine disk system configuration.

2. Create PLATO system user name(s).

3. Obtain files from installation tape.

4. Modify the ASCII network.
   a. Modify network configuration file.
   b. Reinstall NAM and CCP.

5. Reinstall selected NOS programs.

6. Modify system deadstart file text records.

7. Modify release procedure files.
   a. PLATO configuration file
   b. Submit file PLATOD
   c. MFNX
   d. MFDX

8. Build new deadstart file.

9. Initialize ESM.

10. Deadstart on new deadstart file.

11. Load PLATO master files.

12. Create binary master file.

13. Load PLATO via PLAINS DSD-command.

14. Initialize system files.
    a. Installation parameters
    b. Set logical site assignments
    c. Allocate extended memory
    d. System groups
    e. Notes files
    f. Bulletin board
    g. Central print options
    h. AIDS package
    i. AUTHORS package
    j. OPJOBS package

15. Install optional features.

16. Install add-on products.

17. Execute post-installation cleanups.
    a. Delete "install" from group "p".
    b. Run account cleanup programs.
    c. Purge installation-only files.
    d. Convert master files.

18. Install published courseware.

19. Reload PLATO via PLATO OSJ-command.

2.2   Determine disk system configuration.

Before beginning the installation of the PLATO Lesson Authoring
and Delivery application, you must decide how the disk packs
to be used for the PLATO application will be defined.

The NOS V2 Installation Handbook, the NOS V2 Reference Manual
Volume 3 and the NOS V2 Analysis Handbook have information on
how the disk system can be configured.

The default disk system configuration uses the default family
devices. All release procedures are already configured for
this disk system configuration.

If you wish to use an alternate family for all PLATO files,
you will have to modify the release procedures. Each affected
section will describe the changes which need to be made if you
are using an alternate family for all PLATO files.

You may also use either the default family or an alternate
family for the PLATO application, but place the PLATO master
files on auxiliary devices. Each affected section will describe
the changes which need to be made if you are using auxiliary
devices for the PLATO master files.

2.3   Create PLATO system user name(s).

Before you can create the PLATO system user name(s), the NOS
validation files must be created.

If you are using the default family for PLATO files, these
validation files were created during the installation of the
operating system.

If you are using an alternate family for PLATO files, the
NOS validation files for the family must be created. This
may be done by entering the following commands on the computer
console:

                    X.DIS.
                    FAMILY(family name)
                    GENVAL.
                    DROP.

To add required user names when using the default family,
enter the following command on the computer console.

                    X.MODVAL.

Or, if you are using an alternate family device for PLATO
files, enter the following command on the computer console.

                    X.MODVAL(FM=fnam,JP=K)

The MODVAL job will then request the K-display. You must
assign this display to the job by the following procedure.

   1.   Find the job which is requesting the K-display on
        the console B-display.

   2.   Note the job sequence number (JSN) of the job. This
        is a four character identifier for the job.

   3.   Enter the following command on the computer console.

           K,jsn.

Using the MODVAL options documented in the NOS Administration
Handbook, create the following user names. Suggested names
are given, but any legal NOS user name may be used.

   1.   A user name for PLATO system tasks (usually "sys").

   2.   A user name for PLATO print jobs (usually "prints").

Give both user names all possible privileges (AW=ALL) and
maximum resource limits (RL=ALL). The validation mask
parameter for these user names must be set to allow them to
submit system service class jobs (VM=ALL).

The PLATO application also requires the user names PLATOMF
and SYSTEMX. These user names were automatically created
when the operating system was installed.

The passwords for user names PLATOMF and SYSTEMX are needed
later in this procedure. They were set to "platomf" and
"systemx", respectively, when the user names were created.

User name PLATOMF is used when loading the PLATO master files.
You must set the mass storage pru limit for this user name to
"unlimited".

Once these steps have been completed, you will next obtain
the PLATO application files from the installation tape.

2.4   Obtain files from installation tape.

This step will load the files which make up the PLATO
application from the first installation tape.

Mount the first installation tape (the one labeled PLAT1A)
on an available tape drive. Make sure the tape drive is
available by checking the console E,T-display.

Enter the following commands on the computer console:

```
X.DIS.
FAMILY(family name)   if using alternate family
USER(sys,password)    sys is the system user name
                      created in the previous step
LABEL(TAPE,VSN=PLAT1A)
GTR(TAPE,INSTALL)PROC/INSTALL
BEGIN(INSTALL,INSTALL,PPW=ps1,SPW=ps2)

      where  ps1 = password for user name PLATOMF
                   (default = "platomf")
             ps2 = password for user name SYSTEMX
                   (default = "systemx")
```

This procedure creates the following files.

User sys:	PLABINS	PLATO release binaries
	PLATO	PLATO submit file
	CONFIG	PLATO configuration file
	MFNX	procedure to attach master files
	MFOX	procedure to dump master files
	PLANOS	modifications to NOS
	PLANAM	modifications to NAM
	PLACCP	modifications to CCP
	LIBDIR	LIBDECK entries for PLATO programs
	LOADMF	procedure to load installation master files
	BKSPRU	program used to load installation master files

User PLATOMF:	ISTLFOO	terminal resident load file
	ISTLFO1	terminal resident load file
	ISTLFO3	terminal resident load file
	ISTLFC3	terminal resident load file

User SYSTEMX:	PLATO	PLATO load procedure
	PLAINS	PLATO load procedure
	PLAUPD	PLATO load procedure

The following empty files are also created.

User PLATOMF:	AUDIT
	AUDITX
	BACKDIR
	COMBUF
	DUMPDIR

GOBACKO

User SYSTEMX: RAFMON

See the PLATO Bill of Materials section for more information
about these files.

You will next modify the ASCII network to support the PLATO
application.

2.5   Modify the ASCII network.

It is assumed that NAM and CCP have already been installed
on any system which will be using the PLATO ASCII Network.
It is also assumed that the NAM configuration files (Network
Definition Language (NDL) files) have been initialized so
that terminals may be logged into NAM.

This section describes changes which may be needed to the
NAM configuration to allow the use of the PLATO application.

2.5.1   Modify network configuration file.

Make changes to the NAM network definition configuration
file as described below:

   IST terminals are defined as type TTY (tc=a33).

   CDC-721 (Viking) terminals are defined as type 721
   (tc=721).

   Micro-computers used as PLATO terminals are defined as
   type TTY (tc=a33).

   In order to allow IST and Viking terminals to automatically
   load their resident, "pappl" must be set to "plato"
   ("pappl=plato"). Refer to the sample NDL file below for
   an example.

   ENSURE that the NOS user numbers, specified for the
   muser and *juser* parameters in the NDL source file,
   are validated for the "plato" application via MODVAL.

2.5.1.1   Sample NDL file.

For more information on the NCL and NDL Processor, refer to
the NAM Network Definition Language Reference Manual.

A sample program using NDL is shown below. In this example,
it is assumed that a NOS user number of "namnet" exists and
is validated for access only to application "plato".

In order for automatic loading of the terminal resident to
occur, that line must have "pappl" set to "plato".

Only the Viking Terminal is supported at 9600 baud.

ncffile: nfile.

comment   plato network definition.

comment   these defines describe the various ways to define
comment   a port and terminal to the plato/nam interface..

```
platil2:    define    tiptype=async,lspeed=1200.
platl24:    define    tiptype=async,lspeed=2400.
platl48:    define    tiptype=async,lspeed=4800.
platl9c:    define    tiptype=async,lspeed=9600.
pnitty:     define    tc=a33,pri.

comment.    begining of first npu...npu0..

npu0:       npu       node=5,variant=npl,dmp=yes.
suplink     linsme=linkc.
cplrC:      coupler   node=2,hname=nostl.
linkC:      loglink   ncnsme=npu0.

claC:       group     port=01,itype=a2,platil2,nl=1.
pnl0a:      termdev       pnitty,autolog=yes.
cla0:       group     port=02,itype=a2,platl48,nl=1.
pnl0a:      termdev       pnitty,autolog=yes.
cla0:       group     port=03,itype=a2,platil2,nl=9.
pnl0a:      termdev       pnitty,autolou=yes.
cla0:       group     port=0c,itype=a2,platl96,nl=1.
pnl0a:      termdev       pnitty,autolog=yes.
cla0:       group     port=0d,itype=a2,platil2,nl=2.
pnl0a:      termdev       pnitty,autolog=yes.


lcffile:    ifile.

plauser:        define afam=0,muser=namnat,pappl=plato.

plato:      appl.

comment.    user defined for npu0.
pnl0a01:        user  plauser.
pnl0a02:        user  plauser.
pnl0a03:        user  plauser.
pnl0aC4:        user  plauser.
pnl0a05:        user  plauser.
pnl0aC6:        user  plauser.
pnl0a07:        user  plauser.
pnl0a08:        user  plauser.

comment     etc.
```

2.5.2   Reinstall NAM and CCP.

Inspect the contents of files PLANAM and PLACCP under the
PLATO system user name. If either of these files contains
modsets, it will be necessary to reinstall NAM and CCP
with these modsets. Refer to the NOS Installation Handbook
for instructions.

The PLATO application requires only the standard asynchronous
TIP in the CCP build procedure.

Save any modsets in PLANAM and PLACCP so they may be used
again if NAM and CCP must be reinstalled in the future.

You will next reinstall selected NOS programs.

2.6   Reinstall selected NOS programs.

Inspect the contents of file PLANOS under the PLATO system
user name. If this file contains modsets, it will be
necessary to reinstall the affected programs with these
modsets. Refer to the NOS Installation Handbook for more
information.

By looking at the contents of this file, determine what NOS
programs will need to be reassembled. Only the programs
which have modifications will need to be reassembled.

Be sure that, any time in the future, when these programs
are reassembled, that these modifications are included.

Save the binaries produced for replacement on your deadstart
file after all the PLATO materials are ready to be installed.

Save any modsets in PLANOS so they may be used again if the
affected NOS programs must be reinstalled in the future.

You will next modify the system deadstart file text records.

2.7   Modify deadstart file text records.

The system deadstart file text records are used to define
the configuration of your computer and its equipment and
the operating system.

You must make changes to the following deadstart text records
to allow the PLATO application to function correctly.

```
CARDECK
EQPDECK
LIBDECK
IPRDECK
```

This can be done using O26 or some other text editor available
on your system. Instructions for using O26 may be found in
the NOS V2 System Programmer's Instant manual.

Save the modified text records for replacement on your dead-
start file after all the PLATO materials are ready to be
installed.

Refer to the section titled "Deadstart File" in the PLATO
Configuration Handbook for descriptions of the specific
changes which must be made.

The following examples assume you are using O26 as the text
editor and that you are using the first (0) copies of the
text records (CMRD00, EQPD00, LIBD00, IPRD00).

    Example: Modifying a text record.

```
X.DIS.
FAMILY(family name)   if using alternate family
USER(sys,password)
COMMON(SYSTEM)
GTR(SYSTEM,EQPDECK)TEXT/EQPD00
O26.
READ.EQPDECK.
* make required changes.
DIS.
SAVE(EQPDECK)
```

    Example: Inserting PLATO entries into LIBDECK.

```
X.DIS.
FAMILY(family name)   if using alternate family
USER(sys,password)
GET(LIBDIR)
COMMON(SYSTEM)
GTR(SYSTEM,LIBDECK)TEXT/LIBD00
REWIND(*)
SKIPEI(LIBDECK)
COPYBR(LIBDIR,LIBDECK)
PACK(LIBDECK)
SAVE(LIBDECK)
```

You will next modify the release procedure files.

2.8   Modify release procedure files.

The following files have been created under the PLATO system
user name.

                    CONFIG
                    PLATOD
                    MFNX
                    MFOX

These files may require changes depending on the configuration
of your system. You may use O26 or any other text editor
available on your system to make these changes.

The following procedure may be used to modify these files.

                    X.DIS.
                    FAMILY(family name)   If using alternate family
                    USER(sys,password)
                    GET(file name)
                    O26.
                    READ.file name.
                    * make required changes.
                    DIS.
                    REPLACE(file name)

The following sections describe the changes which may be
required for each of these files.

2.8.1   PLATO configuration file

The following changes to your PLATO configuration file are
required. Refer to the PLATO Configuration Handbook for
more information on the configuration file entries.

1. Change the "rid" entry to the routing identifier
    assigned to your system by Control Data. See the
    section titled "Required Information" for instructions
    on how to determine your routing identifier.

2. You should change the "sid" entry to a name you
    wish your system to be known by. This name may be
    a string of any alphanumeric characters of up to
    seven characters.

3. If you did not use "sys" as the PLATO system user
    name, you must change the "subun" entry to the user
    name actually used.

4. If you did not use "prints" as the PLATO print job
    user name, you must change the "prtun" entry to the
    user name actually used.

5. If you are using an alternate family for PLATO files,
    you must change the "famly" entry to the family name
    used.

The following changes to your PLATO configuration file are
recommended, but are not critical to the installation
procedure. These changes may be made now or at a later time.

1. If your mainframe is not a CYBER 180-810, you should
    change the "cpspd" entry to the recommended value
    for your mainframe.

2.9.2   Submit file PLATDD

This file contains calls to the CCL procedures on the system
deadstart file which initiate the PLATO application jobs. The
release file contains the following.

```
            NORERUN.
            PLATX(CP=$+1$)   PLATO
            FRAMX(CP=$+2$)   FORMATTER
            PNIX(CP=$+3$)    PNI
            CONDX(CP=$+4$)   CONDENSOR
            EXIT.
```

The "CP" argument of each procedure call indicates which
control point is occupied by that PLATO job relative to the
control point specified on the ENABLE,PLA entry you have
added to the IPRDECK.

The release submit file assumes that you have used a low-
numbered control point for the ENABLE,PLA IPRDECK entry. If
you wish to use a high-numbered control point, you should
modify the submit file to read as follows:

```
            NORERUN.
            PLATX(CP=$-1$)   PLATO
            FRAMX(CP=$-2$)   FORMATTER
            PNIX(CP=$-3$)    PNI
            CONDX(CP=$-4$)   CONDENSOR
            EXIT.
```

Since the field length of the condensor is dynamic, it is
best to keep it as the last control point to prevent storage
moves for the other control points.

The release submit file assumes there are four contiguous
control points available following the control point specified
by the ENABLE,PLA IPRDECK entry. If this is not the case,
you should modify the PLATO submit file to skip control
points which are occupied by subsystems or change the control
point assignments of other subsystems so there are four
contiguous control points available.

2.8.3 Procedure MFNX

This procedure file is called by the MASTOR job to attacn
all required PLATO master files.

Initially, this file is set up to attacn master files SYS11,
NEWINS, and BINARY. These files will be loaded from tha
installation tapes or created in later steps of tnis
installation procedure.

This file initially contains the following:

```
.PROC,MFNX,M=R.
.*
SUI(377773b)
.*
ATTACH(BINARY)
ATTACH(SYS11)
ATTACH(NEWINS)
.*
REVERT.
```

If you are using an alternate family for PLATO master files,
you must add a FAMILY command to this procedure.

If you are using auxiliary devices for PLATO files, you must
add the appropriate RESOURC and PACKNAM commands to tnis
procedure. Be sure to add the PACKNAM commands following
the SUI command since SUI clears any PACKNAM previously
entered.

2.8.4   Procedure MFDX

This procedure is used by the "backups" file dump procedure
to copy master files to tape or disk. See the PLATO Operations
Guide for more information on this procedure.

Initially, this procedure is set up to dump master files
SYS1I and NEWINS.

This file initially contains the following:

```
.PROC,MFDX,VSN=.
.*
IFE($VSN$ .EQ. $$, DUMP3)
.*
COPYMF(NA,V,MF=SYS1I)
COPYMF(NA,V,MF=NEWINS)
.*
ELSE(DUMP3)
.*
COPYMF(NA,V=VSN,MF=SYS1I)
COPYMF(NA,V=VSN,MF=NEWINS)
.*
ENDIF(DUMP3)
.*
REVERT. MFDX COMPLETE
.*
SKIP(NEXT1)
EXIT.
** ERRORS IN MFDX
WAIT. MFDX ABORTED
REVERT,ABORT. MFDX ABORTED
ENDIF(NEXT1)
```

The release procedure uses the system default tape density.
If you wish to use a different density for master file dumps,
you must add the "D" parameter to each of the COPYMF commands.
Refer to the PLATO Operations Guide for information on using
this parameter.

You will next build a new deadstart file.

2.9   Build new deadstart file.

Build a new deadstart file which includes the following:

1. The NAM which was reinstalled with the modsets in file PLANAM, if any were present.

2. The NOS programs which were reinstalled with the modsets in file PLANOS, if any were present.

3. The CMRDECK, EQPDECK, LIBDECK and IPRDECK which were modified earlier in this procedure.

4. The PLATO procedure files CONFIG, MFNX and MFDX which were loaded from the installation tape and modified earlier in this procedure.

5. The PLATO release binaries which were loaded from the installation tape earlier in this procedure.

The following is a sample procedure which could be used to build the deadstart file.

```
X.DIS.
FAMILY(family name)   If using alternate family
USER(sys,password)
COMMON(SYSTEM)
ATTACH(nam binary file)
ATTACH(nos binary file)
GET(CMRDECK)
GET(EQPDECK)
GET(LIBDECK)
GET(IPRDECK)
GET(CONFIG)
GET(MFNX)
GET(MFDX)
ATTACH(PLABINS)
UNLOAD(DIR)
NOTE(DIR,NR).*FILE nam binary file
NOTE(DIR,NR).*FILE nos binary file
NOTE(DIR,NR).*FILE CMRDECK
NOTE(DIR,NR).*FILE EQPDECK
NOTE(DIR,NR).*FILE LIBDECK
NOTE(DIR,NR).*FILE IPRDECK
NOTE(DIR,NR).*FILE CONFIG
NOTE(DIR,NR).*FILE MFNX
NOTE(DIR,NR).*FILE MFDX
NOTE(DIR,NR).*FILE PLABINS
PACK(DIR)
REWIND(*)
LIBEDIT(P=SYSTEM,I=DIR)
REWIND(*)
*   tape              or       disk
ASSIGN,xx,T,F=I.      or    INSTALL(NEW=EQxx)
COPY(NEW,T,V)
```

You will next initialize Extended Semiconductor Memory, if
your system is using it.

2.10   Initialize ESM.

The PLATO application on 17x or 170-700 series mainframes
uses either Extended Core Storage (ECS) or Extended Semi-
conductor Memory (ESM) to store lesson material. The 800
series mainframes use Unified Extended Memory (UEM). This
section is only applicable to systems which use ESM. Users
of other types of memory should go on to the next step.

ESM may be run in either ECS mode or ESM mode. You should
skip this section if you are using your ESM in ECS mode.

Sites which will be using ESM should read the section titled
"ESM Management" in the PLATO Configuration Handbook before
proceeding with the installation of the PLATO application.

ESM must be properly initialized using the following procedure
before the PLATO application can be run.

   1.   Deadstart the system WITHOUT an ESM equipment (OE or
        OP) entry in the EQPDECK.

   2.   Create an indirect access file named ESMRM under
        NOS user name SYSTEMX. This file is used to hold
        the relocation memory map.

   3.   Enter "X.ESM." at the computer console.

   4.   Assign the K-display to the job.

   5.   Enter "K.CONFIG." at the console. This will initialize
        the relocation memory to your current ESM configuration.

   6.   Enter "K.GO." at the console.

   7.   Re-deadstart the system WITH the ESM equipment in
        the EQPDECK.

You will next deadstart with the new deadstart file created
earlier in this procedure.

2.11   Deadstart on new deadstart file.

You should now deadstart your system on the newly-created
deadstart file.

You will next load the release PLATO master files.

.

2.12   Load PLATO master files.

Use the PLATO procedure LOADMF to copy the PLATO master
files from the installation tapes to disk.

The formats of this procedure call are:

```
            LOADMF(MF=mf)                   default family

            LOADMF(MF=mf,FAMILY=fn)         alternate family

            LOADMF(MF=mf,PN=pn,R=dt)        auxiliary device
```

        where:

```
            mf   = master file name
            fn   = family name
            pn   = auxiliary pack name
            dt   = device type if PN argument is used
```

Mount each installation tape as it is requested via the
E,P-display.

    Example: Default family.

If you are using the default family device for PLATO
master files, enter the following on the computer console.

```
            X.DIS.
            USER(sys,password)
            GET(LOADMF)
            LOADMF(MF=SYS11)
            LOADMF(MF=NEWINS)
```

    Example: Alternate family.

If you are using an alternate family device for PLATO
master files, enter the following on the computer console.

```
            X.DIS.
            USER(sys,password)
            FAMILY(family name)
            GET(LOADMF)
            LOADMF(MF=SYS1I,FAMILY=family name)
            LOADMF(MF=NEWINS,FAMILY=family name)
```

    Example: Auxiliary devices.

To load the PLATO master files onto an auxiliary device
named "PLATOA" of type "DL", enter the following on the
computer console.

```
            X.DIS.
            USER(sys,password)
```

```
FAMILY(family name)   If using alternate family
GET(LOADMF)
LOADMF(MF=SYS1I,PN=PLATOA,R=OL)
LOADMF(MF=NEWINS,PN=PLATOA,R=OL)
```

You will next create the binary master file.

2.13   Create binary master file.

Use the PLATO utility MFCREAT to create a binary master
file. See the PLATO Operations Guide for more information
about the MFCREAT utility.

The customary name for this master file is BINARY. If you
wish to use a different name, you must change the name of
the binary master file attached in procedure MFNX.

You should use the default size for this file for the
device type on which it is being created.

        Example: Default family.

                X.DIS.
                USER(PLATOMF,password)
                PURGE(BINARY/NA)
                MFCREAT(MF=BINARY,PT=BINARY)
                DROP.

        Example: Alternate family.

                X.DIS.
                USER(PLATOMF,password)
                FAMILY(family name)
                PURGE(BINARY/NA)
                MFCREAT(MF=BINARY,PT=BINARY)
                DROP.

        Example: Auxiliary device.

                X.DIS.
                USER(PLATOMF,password)
                PACKNAM(pack name)
                PURGE(BINARY/NA,R=device type)
                MFCREAT(MF=BINARY,PT=BINARY,DT=device type)
                DROP.

You will next load the PLATO application.

2.14   Load PLATO via PLAINS.

You will now load the PLATO application via the PLAINS
DSD-command.

1.   If your system is to use the PLATO ASCII network, NAM
       must be brought up at this point.

2.   Load the PLATO application by entering the following on
       the computer console.

                PLAINS.

       When PLATO (job "PLA1") first comes up, the following
       system dayfile message will be seen:

                ERROR - SYSTEM ID .NE. ROUTING ID
                        SYSTEM ID =
                        ROUTING ID =

       This message appears because the system routing identifer
       has not yet been set. This will be done as the first step
       of the system file initialization.

       Following this message, another error message will be
       seen:

                (SOINIT)   ERROR IN CWARE ACCESS COMMON

       This message indicates that there is no published
       courseware on the system. The file containing the
       courseware access limit common will be delivered
       with the first published courseware shipment. If no
       published courseware is to be available on a system,
       this message will be seen every time the application
       is brought up. This message should be ignored.

3.   Using a PLATO terminal, sign into the PLATO application
       with the following sign-on information.

                name: "install"
                group: "p"
                password: "install"

You will next initialize PLATO system files.

2.15   Initialize system files.

During the following system file initialization, you will
be working with lessons and files which nave been designed
to prompt you through their use. This prompting is
accomplisned through the use of arrows where input is
requested as well as prompts (key words or function keys
such as BACK, NEXT, LAB) and explanations.

2.15.1   Installation parameters

Set system installation parameters.

1.   Set the system identifier as follows:

   a.   Enter the lesson name "ipedit" on the AUTHOR MODE
        display as shown in Figure 1a. Press DATA after
        the name to begin execution of the lesson. This
        procedure will be referred to as "executing lesson
        "ipedit"" later in this document.

   b.   From the first display of options (shown in Figure
        1b), choose the "Network management" option.

   c.   On the next display (shown in Figure 1c), choose the
        "System Identifier" option.

   d.   Take note of the WARNING (as shown in Figure 1d) that
        you are given regarding the assignment of the system
        identifier. After reading the warning, press NEXT.

   e.   Set the System Identifier (RID discussed earlier).
        It must match the "rid" PLATO configuration file
        entry and must NEVER be changed after it has been
        entered for the first time.

   f.   Once you have set the system identifier, press BACK
        once to return to the "Network management" display
        (Figure 1c).

2.   Add your system to the network system table.

   a.   Choose the "System Table" option. This takes you to
        the "Network System Table Management" display (Figure
        1f).

   b.   Choose the "Add a new system to the table" option.

   c.   On the next display, enter your system name. This
        must match the "sid" PLATO configuration file entry.

   d.   On the next display, choose the "Directly connected
        by PCM" option. This is done so the AUTHORS database
        may be used on your system.

   e.   On the next display (Figure 1d), press "1" and enter
        your system routing identifier. This must match the
        "rid" configuration file entry and the system identifier
        entered in step 1 of this section.

   f.   Also on this display, press "J" to set the "Authors
        database available" flag to "yes".

3.   Return to the "Network System Table Management" index
     (Figure 1f) and choose the "Update the EM copy of the

system table" option. Press SHIFT-HELP when instructed
to do so.

4. Now press BACK to return to the main index of
    "Installation Parameters" ("ipedit") shown in Figure 1b
    and set the remaining desired values. Most, if not all,
    of the parameters should be adequate for the time being.

5. After setting all desired installation parameters, press
    SHIFT-STOP to return to the Author Mode display.

You are now ready to set the logical site assignments for
the "runner" stations.

2.15.2   Set logical site assignments

A number of programs run at the highest physical site number
defined on your system. These programs perform functions
which must be periodically done such as maintaining space on
the binary master files, checkpointing student records and
student datafiles to disk, checking for messages to be sent
to users, etc. These so-called "runner" programs should be
assigned to the highest numbered logical site for memory
allocation purposes. Use the following procedure to assign
all stations on the highest numbered physical site for your
system to the highest numbered logical site.

1. Execute lesson "allocate".

2. Choose the "Inspect / Change Station Assignment(s)"
    option.

3. Enter the highest physical site number for your system.
    This would be the value of the "nsite" configuration
    file keyword minus one. The value to be entered is
    also shown on this display as the maximum allowable
    entry.

4. The next display is a list of all site and station
    numbers for the selected physical site. Press "c" to
    change the logical site assignment for a station.

5. The characters ">>" appear in the upper right corner of
    the display.

6. Enter the desired logical site number. For the runner
    programs, this would be logical site number 03.

7. Continue to press "c" and enter the logical site number
    until the last station has been assigned.

8. Press BACK twice to return to the main index display.

You are now ready to allocate extended memory to the logical
sites on your system.

2.15.3   Allocate extended memory

When your system is first brought up, two logical sites,
named "network" and "system", have already been created
with a minimum amount of extended memory allocated to them
to allow you to sign on and initialize the system files.
In this step, you will allocate additional extended memory
to logical site "network" for use by other users.

1. Execute lesson "allocate".

2. Choose the "Inspect / Change EM Allotment(s)" option.

3. The next display lists the first few logical sites. The
    zero (0) entry should be logical site "network". At the
    arrow at the bottom of the display, enter the logical
    site number for logical site "network" and press DATA to
    change its EM allotment.

4. At the top of the next display is a section similar to
    the following:

        Enter New EM Allotment For Site 0   (network)

        Currently Allotted = 40000 of 350000
                             310000 remaining

        Currently Allotted To Site 0 = 20000

5. At the arrow, enter 80% of the number listed as "remaining"
    in the top section of the display as shown above and press
    NEXT.

6. Press BACK twice to return to the main index display.

7. Choose the "Special Functions" option.

8. Choose the "Update System Copy Of Allocation Table"
    option.

9. Press SHIFT-HELP when requested.

10. Press SHIFT-STOP to leave lesson "allocate".

You are now ready to initialize the system groups.

2.15.4   System groups

The following functional groups are to be considered for the
following steps.

        p   = Controllers
        o   = Operations
        pso = Consultants
        m   = Communications

Add sign-ons as needed to these groups using this procedure:

1. Enter the name of the group you wish to edit on the
   Author Mode display and press NEXT.

2. On the first display you see (Figure 2a), select the
   "Roster operations" option by pressing NEXT.

3. On the Roster Options display (Figure 2b), choose the
   "Add someone to the roster" option.

4. You will be asked to choose the type of record you wish
   to create (Figure 2c). Choose to create an "author" sign-on.

5. On the next display (Figure 2d), you will be asked to type
   the name of the new author. If you have already added
   all the sign-ons you wish, press SHIFT-STOP on this
   display to return to the Author Mode display.

6. You will then be prompted for a password (twice, to make
   sure you have entered it correctly), and an expiration
   date.

7. Press DATA to edit this user's sign-on. You should see
   a display resembling Figure 2e.

8. Choose the "choose allowable author OPTIONS" option.
   You will be shown a display which will allow you to
   press DATA to give the person all the options you have
   (Figure 2f). Press DATA.

9. If you are editing group "o" or "p", press NEXT to return
   to editing the sign-on. Otherwise, press BACK to add
   another sign-on (step 5).

11. Choose the "special options" from Figure 2e. You will see
    a display similar to the one in Figure 2g.

12. Press the letter to "change batch jobs allowed flag" to
    permit this user to submit batch jobs.

13. Next, press the letter to "change NOS user number/family".

14. At the first arrow, enter the NOS user name to be
    associated with this user's sign-on. Customarily, it
    is set to be the same as the "subun" PLATO configuration

file entry.

15. At the second arrow, either press NEXT or LAB to use the
    family under which PLATO will normally submit jobs (as
    specified by the "family" configuration file entry), or
    enter some other valid NOS family name.

16. Press BACK to return to Figure 2e.

17. If you are not editing group "p", press BACK to add
    another new sign-on (step 5).

18. Choose the "Choose allowable author OPTIONS" option
    again.

19. Select the "General Record Editing Options" option on
    the next display (Figure 2f).

20. Turn the "set NOS user number" option ON if it is not
    already ON by choosing the letter next to that option.
    (See Figure 2n).

21. Press BACK to return to Figure 2e, then BACK again to
    add another sign-on (step 5).

NOTE

Be sure to add yourself to group "p".

You are now ready to initialize system notes files.

2.15.5   Notes files

User access to any of the following notes files may now be
changed if you do not wish to use the default access. These
notes files are discussed in the PLATO Bill of Materials
section of this document.

        "announce"
        "lessnotes"
        "opsnotes"
        "pbnotes"
        "psonotes"
        "sysin"
        "s0sysmsg"

To change the access for a notes file, do the following:

1. Enter the name of the notes file you wish on the Author
    Mode display and press NEXT.

2. From the main display of the notes file, press SHIFT-DATA
    for director options.

3. If someone has used the director options before you, you
    will see a display giving the information about who has
    used an option. Press NEXT to go to the Director Options
    display.

4. Press NEXT to inspect or change the access list.

5. You will now see a display titled "Notes Access Options"
    (Figure 3a). Press NEXT to see or edit access for people
    on the same system as the notes file.

6. On the "Local" system display (Figure 3b), enter the
    group, account, or name/group you wish to inspect or
    change, and press NEXT.

7. You will now have several options to grant privileges to
    users of the notes file (Figure 3c). You may select upper
    case (SHIFTed) letters to set the associated flags, or
    you may press individual lower case letters as they
    correspond to the privileges you wish to grant.

8. When you are done changing the access list, press BACK
    until you are at the main notes index. Then, press
    SHIFT-STOP to return to the Author Mode display.

Follow this procedure for all of the notes files mentioned
above.

The option "to see all listed sign-ons" from the notes access
options index (Figure 3a) will allow you to see all the sign-
ons that you have added to the access list.

Once you have completed this procedure for each file, you

will be ready to initialize the "Bulletin Board".

2.15.6   Bulletin board

Customize the PLATO Bulletin Board for your system.

1. Edit file "bullfile". If someone has edited this file
    before you, you will see a display giving information
    about who was the last editor of the file. Press NEXT
    to begin editing the file.

2. Edit block "bull" by pressing the letter associated with
    it.

3. Customize the Bulletin Board to suit your system.
    It is not necessary to add any code to this file.
    Simply add the text as you wish it to be displayed.

4. Return to the Author Mode display by pressing BACK.

You are now ready to initialize the central print options.

2.15.7   Central print options

This section describes how to customize the jobs submitted
through the central system print request lesson, "prints".
The jobs submitted by lesson "prints" are initially set to
use an upper case only print train.

1. Edit file "prtsub".

2. Customize the procedures in the following blocks only
    if you wish to do something special with prints. Other-
    wise, you may continue with step "3".

        - "upperlower" = upper/lower case prints
        - "upper case" = upper case only prints

3. Press BACK until you have returned to the Author Mode
    display.

4. Execute lesson "prints".

5. On the next display, you may see a message which says
    "There are no prints to do". If this is the case, press
    NEXT to go to the "Print Requests" options (Figure 4a).
    Otherwise, you will see a page which shows options to
    press SHIFT-DATA and SHIFT-LAB (Figure 4b). These options
    are used when you are actually going to request a print,
    and wish to use a different format from the default format.
    On this display, press SHIFT-BACK for the "Print Requests"
    options, as shown in Figure 4a.

6. If you wish to change the print train, choose the "Print
    Log MAINTENANCE" option.

7. On this display (Figure 4c), choose "Setup Print Types".

8. On this display (Figure 4d), choose the print type that
    you want. Press BACK when done.

9. Now return to the Log Maintenance index (Figure 4c) and
    customize all other parameters on the index.

You are now ready to customize the AIDS package.

2.15.8   AIDS package

You should edit the contents of the blocks in the following
"aids" files to contain specific information about your
system:

          a^psoless
          a^ssl

You are now ready to customize the AUTHORS package.

2.15.9   AUTHORS package

Create an "authors" database for the home system as follows:

1. Execute lesson "authors". You will first see a display
    similar to Figure 5a. Press NEXT.

2. Press SHIFT-NEXT to initialize the authors database for
    your system.

3. When asked about expected values (Figure 5b), your answers
    are not critical at this time. They may be changed later.

4. After the database creation is complete, press SHIFT-
    STOP to return to the Author Mode display.

Activate all other systems for which databases exist as
follows:

1. Execute lesson "authors" again.

2. Press SHIFT-DATA for director options.

3. You will note the message at the top of the display (as
    shown in Figure 5c), which instructs you to press DATA
    to turn on "authors". Do so at this time.

4. On the display shown in Figure 5d, select option "o" to
    turn "authors" ON.

5. In addition, on the same display, make sure your sign-on
    is shown as the person responsible for this system's
    database. If it is not, select option "a" to change it
    and enter your sign-on name and group as prompted.

6. Now return to the Director Options Index (Figure 5c) and
    select the "Database management options".

7. Choose the "active systems" option on the next display
    (Figure 5e).

8. Press SHIFT-NEXT to install all system databases. This
    option turns the AUTHORS package off again.

9. When asked for a file prefix (as in Figure 5f), press
    NEXT for "none". Processing will begin automatically.
    Disregard messages that might be displayed.

10. After processing is complete, you will be asked if you
    would like to turn "authors" back on. Press NEXT to do
    so, then press SHIFT-STOP to exit "authors".

You are now ready to customize the UPJOBS package.

2.15.10   OPJOBS packaje

Refer to the "Usage Tracking" section of the PLATO Operations
Guide for information about the use of "opjobs" to track
PLATO usage before initializing this feature.

1.   Execute lesson "opjobs".

2.   Press LAB "to see/edit operating parameters."

3.   Set the operating parameters to suit your system.

4.   Edit file "prtsub".

5.   Customize the procedures in the following blocks, if you
     wish to do something special with the routing of prints.

          -   "opjobs1"   = disposes of output after processing
                               account files via "opjobs"
          -   "opjobs2"   = disposes of output after generating
                               an accounting summary report

You are now ready to initialize any optional features your
PLATO application might have.

2.15   Install optional features.

There are several optional features which may or may not be
initialized depending on the resources available and the
desire for the feature. Some of these features are NOT
available on all systems.

These optional features may be installed during the initial
system installation or at any time afterward.

See the "Optional Installation Procedures" section for more
information and procedures to initialize these optional features.

If you do not wish to install any of these optional features
at this time, or, after completing the installation of desired
optional features, you should continue with the installation
of add-on products as described in the following section.

2.17   Install add-on products.

At this time, any optional "add-on" PLATO products, such as
the PCD2 On-Line Authoring Module, which have been purchased
may be installed. Refer to the "Add-on Products" section
of this document for installation instructions.

After completing the installation of Add-On Products, you
should continue with the Post-Installation Cleanup, as
described in the following section.

2.19   Execute post-installation cleanups.

The following cleanup procedures should now be executed:

- remove the installation signon record

- make sure all files are in the proper PLATO account

- purge installation-only disk files

- convert release PLATO master files

2.19   Execute post-installation cleanups.

The following cleanup procedures should now be executed:

- remove the installation signon record

- make sure all files are in the proper PLATO account

2.18.1   Delete "Install" of group "p".

You should now delete sign-on "install" of group "p".
Follow these steps:

1. Sign off the system and sign back on with your own
    sign-on in group "p".

2. Edit group "p".

3. Select "Roster Options".

4. Select the option to delete a sign-on.

5. Enter the name "install" when asked what name to delete.

6. Press SHIFT-HELP.

7. Press BACK until you have returned to the Author Mode
    display.

2.18.2   Run account cleanup programs.

Run an account cleanup on all accounts on the system.   This
procedure involves the following:

1. Execute lesson "accountu".

2. Choose the option "update an account file name table".

3. Choose "b" for more processing time.

4. Press NEXT to build a new list of files.

5. After the list of files has been sorted, you will be
   prompted for an account name. Press SHIFT-NEXT to
   process all accounts.

6. Press SHIFT-HELP to start the process.

                    NOTE

        The process can be very lengthy
        depending on how many master files
        are on your system.

7. When the word "DONE" appears, press SHIFT-STOP to return
   to the Author Mode display.

2.18.3   Purge disk files.

The following NOS permanent files may be purged to save
space since they are needed only during the installation
process.

```
PLABINS    PLATO release binaries
CONFIG     PLATO configuration file
MFNX       procedure to attach master files
MFDX       procedure to dump master files
LIBDIR     LIBDECK entries for PLATO programs
LOADMF     procedure to load installation
           master files
BKSPRU     program used to load installation
           master files
```

Enter the following commands on the computer console:

```
X.DIS.
FAMILY(family name)   if using alternate family
USER(sys,password)
PURGE(PLABINS)
    .
    .
    .
```

2.18.4   Convert master files.

These procedures are not critical. The PLATO application
will work correctly if these procedures are not done until
some time in the future.

You may want to rename your master files to use a naming
convention which is popular at other sites.

To rename master files SYS1I and NEWINS to SYSTEM and AMAST,
do the following:

1.   Take the PLATO application down.

2.   Execute the following job at the computer console.

```
            X.DIS.
            FAMILY(family name)  if using alternate family
            USER(PLATOMF,password)
            PACKNAM(pack name)  if using auxiliary devices
            CHANGE(SYSTEM=SYS1I)
            CHANGE(AMAST=NEWINS)
            ATTACH(SYSTEM,AMAST/M=w)
            MFALTER(MF=SYSTEM,N=SYSTEM)
            MFALTER(MF=AMAST,N=AMAST)
            DROP.
```

4.   Change procedure MFNX on the current deadstart file to
     change the master file names.

5.   Change procedure MFDX on the current deadstart file to
     change the master file names.

6.   Make a new deadstart file with the changed MFNX and MFDX
     procedures and reload the system with it.

7.   Bring up the PLATO application.

8.   Rename the master files in the required master files
     table in lesson "lpedit".

9.   Use program BACKMDD to rename the master files in the
     list of master files to be dumped.

If you wish to expand the disk space available for user
files on your system, you may do one of the following:

     - Add another required master file to your system. Refer
       to the "Adding/Changing a Required Master File" section
       of the PLATO Configuration Handbook for information on
       how to do this.

     - Use procedure MFPACK to increase the size of the existing
       type general master file. This would permit you to
       take better advantage of the capacity of the master file

directories. In order to do this, you may have to add
another disk pack to your system.

The following example assumes you have renamed master
file NEWINS to AMAST. The SP parameter given in the
example is the maximum size for a master file on an
834 (DD) device. If your master file is on a different
device type, use the maximum for that device given in
the section describing the MFCREAT command in the PLATO
Operations Guide.

1.   Take the PLATO application down.

2.   Increase the size of the master file by executing
      one of the following jobs at the computer console.

      Default family or alternate family:

          X.DIS.
          FAMILY(family name) If using alternate family
          USER(PLATOMF,password)
          MFPACK(MF=AMAST,N=AMAST,PT=GENERAL,SP=3680)
          DROP.

      Auxiliary devices:

          X.DIS.
          USER(PLATOMF,password)
          MFPACK(MF=AMAST,N=AMAST,PN=pack name,
            R=device type,NPN=new pack name,NR=new
            device type,PT=GENERAL,SP=3560)
          DROP.

You will next install published courseware.

2.19   Install publisned courseware.

Tne Initial shipment of new courseware to your site will be
composed of any or all of the following courseware Categories:

    Category I   - Acadeaic (Libraries A/B)
    Category II   - Authors Library
    Category III   - Educational (Libraries C/E/G/H/M and LDEC)
    Category IV   - Training (Libraries D/F/I/J/K/L)

The tapes containing the courseware will be labeled as follows:

    Category I   - catia / catlb / catlc / ...
    Category II   - cat2a / ...
    Category III   - cat3a / ...
    Category IV   - cat4a / ...

In addition, tne Initial shipment will include a site-specific
tape, wnose label will correspond to the site name. It contains
the published courseware catalog (accessed by pressing snifted
"F" from the Author Mode display) and tne flle that holds tne
courseware access limit for your site. This limit is tne
maximum number of users who may simultaneously be accessing
puolished courseware. This tape also contains other course-
ware management work files.

        NOTE: The courseware access limit will not take effect
               until (1) tne site-specific tape nas been in-
               stalled, and (2) tne PLATO application nas been
               taken down and brought back up, since tne limit
               is read in and set only at tnat time. Until
               then, nobody will be able to access any of tne
               courseware.

If your site nas ordered courseware by title, rather than by
entire Categories, tne Initial shipment tapes tapes will be
labeled like the site-specific tape: "sitel", "site2", etc.

For every tape received, there will be an accompanying
MFPRINT, listing all the files within tnat master flle tape.

Regardless of tne tape labels and the contents, eacn tape
contains a single, 2000-part master flle. The metnod for
Installing tne courseware involves adding each master flle
as a required master flle and performing accouit clean up
procedures.

Tne following steps snoulc be followed for all tapes EXCEPT
tne site-specific tape, wnicn snould be installed as described
In the "Courseware Installation" section of tne PLATO Operations
Guide after all otner tapes have been installed. Tnese steps
are described In more oatall in the following sections.

a.   Ensure sufficient disk space under NJS.

b. Copy all tapes to disk via MFTLOAD.

c. Use the procedure described in the "Changing a Required
    Master File" section of the PLATO Configuration Hand-
    book to change the master file name and the master file
    type for all the newly-copied courseware master files.

d. Execute the "Search for duplicate files" option of
    lesson "installu".

e. Execute the "Update account file name table" option of
    lesson "accountu".

f. Execute the "Account directory and file checks" option
    of lesson "accountu".

g. Execute the "Search for files not in an account" option
    of lesson "accountu".

h. Install the site-specific tape as described in the
    "Courseware Installation" section of the PLATO Operations
    Guide.

2.19.1   Courseware Installation - details

The following is a detailed explanation of each of the steps
described above:

a. "Ensure sufficient disk space under NOS."

    A single disk part is 35 sectors, so each 2000-part
    master file will occupy 70,000 sectors. The number of
    free tracks required for each master file on each
    supported device type is shown below.

                    DD   6670 tracks
                    DG   175L tracks
                    DI   1217b tracks
                    DJ   465b tracks
                    DK   1161b tracks
                    DL   465L tracks
                    DM   156b tracks
                    DO   156b tracks

    If you received more courseware than there is room for
    on your system, please contact your sales representative.
    Your representative should then, in turn, call PLATO
    Courseware Delivery to discuss the resolution of this
    problem.

b. "Copy all tapes to disk via MFTLOAD."

    Refer to the PLATO Operations Guide for details on how
    to use MFTLOAD to copy master file tapes to disk.

c.   "Use the procedure described in the "Changing a Required
     Master File" section of the PLATO Configuration Hand-
     book to change the master file name and the master file
     type for all the newly-copied courseware master files."

     Using MFALTER (refer to the PLATO Operations Guide for
     details), change the master file names from names like
     "catla", etc., to a name similar to others in use on
     your system. Traditionally, the names "amast", "bmast",
     etc., have been used. In addition, to ensure the proper
     operation of your system, the first four (4) letters of
     all master file names must be unique.

     At the same time as each master file name is being
     changed, its type should be changed to "general".

d.   "Execute the "Search for duplicate files" option of
     lesson "installu"."

     The way the various Categories are built is such that
     each of them is self-sufficient, as opposed to requi-
     ring that a different Category also be installed. For
     this reason, it is necessary to delete any duplicate
     files that may exist on your system. The steps to
     follow are:

     1.   Execute lesson "installu".

     2.   Select the "Search for duplicate files" option.

     3.   At the arrow, just press SHIFT-NEXT, to indicate
          that all master files are to be searched.

     You will then see each master file name displayed, and
     every 64th file shown. Should a duplicate be encoun-
     tered, you will be shown the information for each file
     (account, last edit info, master file, length, etc.).
     Based on that information, with primary consideration
     of the last edit date/time and the master file of resi-
     dence (the copy of the file on the earlier-numbered
     master file will be the copy accessed by users), press
     either SHIFT-HELP or SHIFT-DATA to delete one file or
     the other.

     Follow this procedure until all duplicate files have
     been deleted, before proceeding to the next step.

2.19.2   Courseware Installation - details

e.   "Execute the "Update account file name table" option of
     lesson "accountu"."

     This step is performed to update the directories of all
     accounts on the system so that they accurately represent
     the file and space count for the files in each account.

The procedure to be performed is as follows.

1. Reload the PLATO application using the PLAINS
    command or turn off file management options through
    lesson "accountl".

2. Execute lesson "accountu".

3. Choose the "update an account file name table"
    option.

4. Press "b" to indicate that processing is to be
    done in "background" mode (gives more processing
    time).

5. Press NEXT to "build a new list of files".

6. After the list is built and sorted, you will be
    given the option to press SHIFT-NEXT to process
    all accounts. Do so.

7. Press SHIFT-HELP to start the process. When com-
    pleted, the screen will erase and the message
    "DONE" will appear.

f. "Execute the "Account directory and file checks" option
    of lesson "accountu"."

    This option checks each file on the system, and
    sets the published flag, if the account is published,
    and resets the account ordinal in the file directory.
    Before actually performing the steps, it might be of
    use to press HELP from within this option to see the
    list of features, followed by a list of the possible
    errors and what to do in case of each.

    The steps to be followed are:

    1. From the index of "accountu", choose the "Account
        directory and file checks" option.

    2. Press "b" to indicate that processing is to be
        done in "background" mode (gives more processing
        time).

    3. In the following display, choose the options as
        follows:

            File space computation?   y
            NL space computation?   n
            Directory pointer check?   y
            Log errors in datafile?   y (see below)
            Display errors on screen?   n (see below)

        NOTE: Since this process will take a long time,
        it is assumed that you will not want to stand
        by and wait for an error to be encountered, but

rather that you will examine the error log (file
"accerrlog") after processing, to see if any errors
were encountered.

4.   When prompted for an account name, press NEXT
     with no input, to indicate that all accounts are
     to be processed.

5.   You will then be given the choice of pressing NEXT
     to begin processing or SHIFT-HELP to clear the
     datafile. Press SHIFT-HELP.

When all accounts have been processed, the message
"COMPLETED" will be displayed. At that point, check
the log ("accerrlog") to see what kinds of problems
may have come up. To find out how to correct them,
refer back to the HELP sequence. If there were no
errors, looking at "accerrlog" will show that 0 blocks
are in use; otherwise, you will be given the option to
look through the file contents.

2.19.3   Courseware Installation - details

g. "Execute the "Search for files not in an account" option
    of lesson "accountu"."

    This option involves going through each master file on
    the system, and logging every file that does not appear
    in the "list of files" prepared earlier. Seeing as the
    existing files/accounts are all in accordance with each
    other, the only files that show up in this search will
    be those which indicate they are in non-existent accounts.
    The steps to follow are:

    1.   Choose the "Search for files not in any account"
         option from the main index of "accountu".

         (From this point, you may wish to press HELP to
         get a more detailed explanation of what this op-
         tion does.)

    2.   Press "b" to indicate that processing is to be
         done in "background" mode (gives more processing
         time).

    3.   If you feel certain of the integrity of the
         earlier-built list of files, press SHIFT-NEXT
         to proceed; otherwise, press NEXT to build and
         sort a new list.

    4.   Assuming that all problems previously logged in
         "accerrlog" have been resolved, press SHIFT-HELP
         when prompted to initialize the datafile.

    You will then see a display where each master file name
    will be plotted, and then every "offending" file name

shown, along with the alleged account name. Since the
displayed entries are being logged, there is no need to
watch/record the file names as they appear.

When processing is complete, the message "DONE" will be
displayed at the bottom of the display.

As indicated in the HELP sequence, problem files should
be moved into legitimate accounts via "accounti".

h.   "Install the site-specific tape as described in the
        "Courseware Installation" section of the PLATO Operations
        Guide."

This concludes the initial installation of courseware on
your system.

2.20   Reload PLATO via PLATO.

Take down the PLATO application as described in the Operations
section of the PLATO Operations Guide.

Bring the PLATO application back up by entering "PLATO."
at the computer console. This is the normal entry to bring
up the PLATO application when not doing a system installation
or upgrade. This will allow any users to sign in.

The installation of the PLATO Lesson Authoring and Delivery
1 application is now complete.

3   Optional Installation Procedures

SECTION 3 - OPTIONAL INSTALLATION PROCEDURES

There are several optional features of the PLATO Lesson
Delivery and Authoring System which may be initialized
depending on the resources available and the desire for
the feature. Some of these features are NOT available
on all systems.

These optional features may be initialized during the initial
system installation or at any time afterward.

See the following sections for more information and procedures
to initialize these optional features.

3.1   Archiving

Please read the section in the PLATO Operations Guide which describes the Archiving Utility and its operation.

The archiving utilities are included in the software sent to every system. To use this feature on your system, install as follows:

1. Set the warning and deletion cycles for archive files as follows:

    a. Execute lesson "lpedit".

    b. Choose option "archive recycle period".

    c. Enter settings as desired.

    d. Press SHIFT-STOP to exit "lpedit".

2. Create file "sOarcnds" according to the specifications given in the PLATO Bill of Materials section of this document.

3. Execute lesson "archiver". It will prompt for initialization when executed the first time. Press SHIFT-NEXT to initialize.

4. Enable the archive feature as follows:

    a. Execute lesson "account1".

    b. Choose "file backup and archive options".

    c. If archiving is currently disallowed, choose the option to "allow archiving".

Before an account can use the archiving utility, it must be assigned "archive rights". See the PLATO Operations Guide for instructions on how to do this.

3.2   Backups

Please read the section in the PLATO Operations Guide which describes the Backups Utility and its operation.

The Backup procedure BACKDMP requires that either SORTMRG4 or SORTS be available on the system.

The procedure to initialize the Backups utility for use is described below.

1. Use the procedure "Setting Up Dump Directory Datasets" in the PLATO Operations Guide to set up 6 datasets.

2. Create file "sobacka" according to the specifications given in the PLATO Bill of Materials section of this document.

3. Create file "sobackc" according to the specifications given in the PLATO Bill of Materials section of this document.

4. Run the following job at the console:

                X.BACKMOD.
                K,JSN.
                K.INITIALIZE
                K.WR
                K.END
                X.BACKCPY.

5. Use the procedure "Initializing the Slot Table" in the PLATO Operations Guide to set up the desired dump cycle.

6. Use the "Changing the Master Files to be Dumped" procedure in the PLATO Operations Guide to initialize the list of master files you wish to dump. You should add master files SYSII and NEWINS to this list.

3.3   Central Micro PLATO executor

The Central Micro PLATO Executor (CMP) allows lessons written
in the Micro PLATO Language to be executed on the central
system. It aids in development of lessons written in the
Micro PLATO Language, but also increases the amount of EM
used as overhead by the PLATO system, reducing the amount
of EM available for user lessons.

To install the Micro PLATO Executor:

1. Add the following to the PLATO configuration file:

        cmp=on.

2. Create file "s0cmp0" according to the specifications
    given in the PLATO Bill of Materials section of this
    document.

3. Edit file "s0cmp0".

4. Press DATA to edit the file information.

5. Choose the "associated files" option.

6. Set the -use- file to "s0cmpuse".

7. Press BACK to return to the file information index.

8. Choose the "Micro PLATO level" option.

9. Set the Micro PLATO level to "3" and the execution mode
    to "Central System execution".

10. If you are using more than one condensor, repeat this
    procedure for files "s0cmp1" and "s0cmp2".

3.4   Network datebase

A network database describing all aspects of your communi-
cations network may be built using lesson "pnet". This
database may be quickly scanned to aid in the isolation
of network problems.

A HELP-section in lesson "pnet" provides additional
information.

The network database is built via options in "pnet" as
follows:

1. Enter all locations in the location table.

2. Enter all equipment types in the equipment table.

3. Enter all terminal types in the terminal type table.

4. Define any desired utility fields.

5. Add all circuits and rotaries in your network. As
    each entry is made, add all known information to the
    corresponding information fields.

3.5   Computer Interface Unit network

The CIU (Central Interface Unit) and SIU (Site Interface Unit) are unique to the PLATO application. This network is not available on CYBER 170-800 series machines.

1. Add the CI and O2 EQPDECK entries. See the PLATO Configuration Handbook for more information.

2. Add / Change the following entries in the PLATO configuration file.

         c0sit
         clsit
         nc0sl
         nclsl

Lesson "cludiaj" is the CIU diagnostics utility. If you wish to use this optional feature, do the following:

1. Create file "s0cluns" according to the specifications given in the PLATO Bill of Materials section of this document.

2. Execute lesson "s0cluin" and execute all initialization options.

3. Add the following to the runner list according to the specifications given in the PLATO Configuration Handbook.

         s0cluru

3.5   TRANSMIT Utility

The TRANSMIT Utility allows users to transfer data from
PLATO files into NOS files for batch jobs to process.

To install TRANSMIT, the following steps should be followed:

1. One or more datasets must be created for use in trans-
    ferring data. These files should be created in account
    "sOfiles", and have the following characteristics:

            record size: 64 words
            length: minimum of 10 parts. Performance
                            improves as the length approaches
                            63 parts. (The length determines
                            how many separate batch jobs must
                            be submitted to transfer the files
                            to NOS files.)

    The number of datasets created determines how
    many users can use TRANSMIT simultaneously. Any number
    from one to 60 is acceptable, assuming sufficient sys-
    tem resources are available.

2. Enter the names of all datasets created in leslist
    "sOxait" in file "sCcommon".

3. Enable the use of TRANSMIT on the system:

    a. Execute lesson "ipedit".

    b. Choose the "batch submission control" option.

    c. Edit the appropriate mainframe entries to enable
        both the "User jobs" and "Transmit" features on
        the desired mainframe(s).

Accounts must be allowed to use this utility. See the
"TRANSMIT" section in the PLATO Operations Guide for more
information on how to do this.

3.7   PLATO Inter-System Link

Before beginning the installation of the PLATO Inter-System
Link optional product, please read the overview given in the
PLATO Operations Guide.

Before you install the PLATO Inter-System Link, you must
first install the PLATO application (level 34.2 or higher)
and the RHP application materials. Also, be sure to check
the SOLVER data base for any corrective modsets for the RHP
application. In particular, for NOS 2.4.1 level 630, you must
incorporate PSR INSA096 and PSR RHFB173 modsets.

The following is a list of the steps to be taken to install
the PLATO Inter-system Link. Each of these steps is discussed
in detail in the following sections.

1. Determine accounting method to be used.

2. Create NOS user names.

3. Modify the network configuration file.

4. Modify the PLATO network system table.

5. Create required system-specific files.

6. Install the required runner programs.

3.7.1   Determine accounting method

Read the "Link accounting" section of the PLATO Configuration
Handbook and decide which method of accounting for link traffic
you wish to use for your system. This will affect the
information you must enter into the PLATO network system table
for each linked system.

3.7.2   Create NOS user names

You need to set up two NOS user names that are used to
transfer data between systems. Follow these steps:

1. You have the option of placing your link data files
    on an alternate family device.

    If you wish to use an alternate family, the NOS
    validation files for the family must be created if they
    do not already exist. This may be done by entering the
    following commands on the computer console:

                   X.DIS.
                   FAMILY(family name)
                   GENVAL.
                   DROP.

2. If you are using the default family, enter the following command:

   X.MJDVAL.

   Or, if you are using an alternate family device for the link files, enter the following command:

   X.MJDVAL(FM=fnam,JP=K)

3. The MJDVAL job will then request the K-display. Assign this display to the job by the following procedure:

   a. Find the job which is requesting the K-display on the console B-display.

   b. Note the job sequence number (JSN) of the job. This is a four character identifier for the job.

   c. Enter the following command on the computer console:

      K,jsn.

4. Using the MJDVAL options documented in the NOS Version 2 Administration Handbook, create the following user names:

   a. User name "PLASEND". This name stores data that is sent to other systems.

   b. User name "PLARECV". This name stores data that is received from other systems.

5. The user names must have certain validations for security reasons. Unless otherwise noted, validations not mentioned are assumed to be site-specific.

   a. You must set the following validations:

      - AW = CLPF, CCNR, CSRP, CSAP, CPLK, CUST, CSAF, CCPR.
      - CC = 778.
      - CP = 0.
      - CS = 0.
      - DB = 773.
      - DF = 773.
      - DS = 773.
      - DT = 0.
      - EC = 0.
      - FC = 7.
      - FS = 3.
      - LP = 0.
      - MS = 773.
      - MT = 0.
      - PT = 0.
      - RP = 7.
      - SL = 778.
      - TL = 778.

- UC = BCCT, RBCT, IACT.
- VM = ALL.

b. Choose a password, and set both user names to use
this password. The user names MUST nave identical
passwords. This password is the key to controlling
access from other systems. You aust exchange
passwords with the administrators of the remote
site whan you wish to perait inter-system data
transfers.

c. If you wisn to set a default charge number for
accounting purposes, you should do so at this time
as described in the PLATO Configuration Handbook.

3.7.3   Modify network configuration file

Now you need to set up your NOS communication network. To
establisn a connection to another system, you need to define
a path through NAM's network configuration file and RHP's
logical identifier (LID) table. This procedure and otner
details aoout the installation ano operation of these
applications can be found in the following references:

    NOS Version 2 Feature Notes
    NOS Varsion 2 Installation Handbook   (60459320)
    NOS Version 2 Analysis Handbook   (60459300)
    Network Definition Language Reference Manual (60480000)

Follow these steps:

1. Update your NOS level communication network.

    a. Update the LID configuration file. Refer to the
    the NOS Version 2 Analysis Handbook for examples.
    You need to specify a LID for your system -- we
    suggest you use your routing identifier as specified
    oy the "rid" PLATO configuration file keyword. You
    will also need to add PIDs and LIDs for all the
    other systems in your network. You will need to
    share this information with the administrators of
    other sites.

    b. Update your NDL file with changes necessary to run
    the RHP applications. Here are some examples of
    NDL entries you will need to make:

        * LINE definitions

        line: LINE,PORT=port,LTYPE=ltype,TIPTYPE=tiptype,
                PSN=osn,NSVC=svcric,DFL=ofl,FRAME=frame,
                RTIME=timer,RCOUNT=count,DCE=yn2.
        device: TEKMDEV,STIP=stiptyp,NCIR=nuacir,NEN=encir,
                DT=devtyp.

        * APPL statements

```
PTF:      APPL,PRU,NETXFR,MXCOPYS=4.
PTFS:     APPL,PRU,RS,NETXFR,MXCOPYS=4.
```

* INCALL and OUTCALL statements for X.25

```
INCALL    ANAME=ptfs,FAM=famnama,UNAME=usernam,
          SNODE=srcnode,PORT=portnum,DNODE=dstnode,
          DBZ=dwnlsiz,UBZ=upbsize,DPLS=dpls.
OUTCALL   NAME1=ptfs,PID=pidname,SNODE=srcnode,
          DNODE=dstnode,PORT=portnum,DBZ=dwnlsiz,
          UBZ=upbsize,DPLS=dpls,SHOST=srchost,
          DHOST=dsthost,DTEA=dtea.
```

* INCALL and OUTCALL statements for shared 2550

```
INCALL    ANAME=ptfs,FAM=famname,UNAME=usernam,
          DBL=dwnblla,ABL=abl.
OUTCALL   NAME1=ptfs,PID=pidname,SNODE=srcnode,
          DNODE=dstnode,DBL=dwnblla,ABL=abl.
```

* INCALL and OUTCALL statements for direct line
*   or TRUNK

```
INCALL    ANAME=ptfs,FAM=famname,UNAME=usernam,
          SNODE=srcnode,PORT=portnum,DNODE=dstnode,
          DBZ=dwnlsiz,UBZ=upbsize,DPLS=dpls.
OUTCALL   NAME1=ptfs,PID=pidname,SNODE=srcnode,
          DNODE=dstnode,PORT=portnum,DBZ=dwnlsiz,
          UBZ=upbsize,DPLS=dpls,SHOST=srchost,
          DHOST=dstnost.
```

Refer to the NOS Version 2 Feature Notes for more
examples.

c.   Build your new network configuration file and
     corresponding local configuration file using the
     NDLP system command. Refer to the Network
     Definition Language Reference Manual for examples.

d.   Update your current NAM start-up file with the new
     NAM start-up file that was created with the RHP
     build process. Also, please make sure that the
     following NOS files are moved to the same user
     index as the NAM start-up file:

          ZZPTFS
          ZZOTFS

e.   Bring down NAM, then reload NAM using the new
     configuration files.

2.   Your NOS communication network is now updated. Be sure
     to test your network.

3.7.4   Modify PLATO network system table

You now need to update the PLATO network system table,

which includes descriptions of the links between your
system and other systems, and the options available to
each link.

First, you must modify the network system table entry for
your own system as follows:

a.   Sign on to PLATO with your "p" signon.

b.   Execute lesson "ipedit".

c.   Choose the "Network Management" option.

d.   Choose the "System Table" option. This takes you
     to the "Network System Table Management" display.

e.   Choose the "Inspect or change a table entry" option.

f.   On the next display, enter your system name and
     press NEXT. This name is the same as that specified
     by the "sid" PLATO configuration file keyword.

g.   Choose the "Network Type:" option.

h.   Choose the "Directly connected by 2550" option.
     The display will replot and show a number of
     additional options.

i.   Choose the "NAM mainframe number" option. Enter
     the number of the mainframe that executes the NAM
     application. On most systems this will be the
     default value of 0.

j.   Choose the "Family name:" option. Enter the NOS
     family name that you set up to send and recieve
     data. If you wish to use the default family,
     simply press NEXT.

k.   Choose the "User name password:" option. Enter the
     the password that you assigned to the user names
     PLASEND and PLARECV. The password entered MUST
     be the one entered for both user names earlier in
     the installation process. There is no default.

l.   Choose the "Charge number:" option. If you will
     not be using NOS charge numbers, or, if you plan to
     use the default charge number you specified for the
     PLARECV/PLASEND user names, press NEXT. If you
     want to account for each system's usage, enter a
     charge number here. Refer to the "Link Accounting"
     section of this document for details.

m.   You may wish to store the sending and receiving
     data files on an auxiliary pack because of space
     constraints on your "system" pack. If so, choose
     the "Pack name:" option and enter the desired NOS
     pack name. If you are using the default pack,

press NEXT.

n.   If you chose to store your files on an auxiliary
     pack, choose tne "Pack device type:" option and
     enter tne device type of the pack you entered. The
     default is 'di'.

o.   Turn on the appropriate oata transfer options.
     For instance, if you want to oe able to send and
     receive pnotes, turn botn these options on.

p.   Press BACK to return to tne "Network Systea Taole
     Hanagement" index.

You shoulj now use the procedure described in the "Adding a
systea" section of the PLATO Configuration Guide to add new
systeas or use the sase procedure, selecting the "Inspect or
change a table entry" option in place of tne "Add a new system
to the table" option on the "Network Systea Table Management"
index, to modify existing systeas to establisn a link.

3.7.5   Create systea-specific files

If the following requlreJ, systea-specific files do not
exist on your systea, you snould now create thea in account
"sCfiles" according to the specifications jiven in tne
PLATJ Bill of Materials section.

          sGrnpds
          sOrnplog
          sOrnpns
          3neting
          3natoutq

3.7.6   Install runner programs

Now you need to set up a runner which will execute tne
PLATJ lesson which looks for link traffic. Follow tnese
steps:

a.   Sign on to tne systea with your group "p" signon.

b.   Execute lesson "runnersys".

c.   Choose tne option to "Add lesson to list".

o.   On tne next oisplay, enter the lesson nase "sOrnp".

e.   On the next diplay, choose tne "cycle" option and
     enter a cycle time of 5 winutes.

f.   No other options need oe set, unless you have sose
     specific site requirements.

g.   On the same display, press SHIFT-LAB to make tne
     runner active.

h.   Use the same procedure to add lessons "pnotes" and
     "sgnotrun" to the runner list. These lessons will
     distribute personal notes and group notes received
     from remote systems to the appropriate files. Use
     the attributes listed in the "Runner Lesson Man gement"
     section of the PLATO Operations Guide when adding
     these lessons.

i.   Press SHIFT-STOP to return to the Author Mode page.

The installation of the PLATO Inter-system Link optional
feature is now complete.

4   Add-on Products

SECTION 4 - ADD-ON PRODUCTS INSTALLATION PROCEDURES

4   Add-on Products

INTRODUCTION

This section describes the initial and update installation
procedures for the separately ordered add-on PLATO products.
These products include:

    a.   PLATO Courseware Development and Delivery (PCD2)

4.1   PCO2

Before beginning the installation of the PCO2 add-on product,
please read the overview given in the "Add-on Products" section
of the PLATO Operations Guide.

4.1.1   PCO2 Installation Procedures

The installation of the PCO2 product requires that you have
already installed the PLATO application and that the PLATO
and PCO2 release level are the same.

The first part of installing PCO2 involves modifying the
currently running PLATO binary. Perform the following steps.

1. Mount the tape labelled PLAT3D on an available tape
    drive.

2. Enter the following at the computer console.

        X.DIS.
        LABEL(TAPE,D=PE,F=I,VSN=PLAT3D)
        COPYBF(TAPE,PSMDX)
        BEGIN(,PSMDX)

    Do NOT enter "DROP." when this process is complete.

3. The resulting file, LGO, is a modified version of the
    currently running PLATO binary which will allow use of
    the PCO2 on-line authoring system. It should be used as
    the binary input to LIBEDIT to build a new deadstart file.

4. Re-deadstart your system with the new deadstart file
    and bring up PLATO in the standard manner.

The second part of the installation process is the optional
step of customizing the PCO2 access list.

By default, all users have access to PCO2 when it is first
installed. If you wish to limit access to specific users,
refer to the "PCO2 Operation" section of the PLATO Operations
Guide.

If you do not wish to limit access to PCO2 at this time,
the installation of the PCO2 product is now complete.

4.1.2   PCD2 Update Installation

MAJOR RELEASE UPDATES

Because PCD2 is an add-on application to be used with PLATO
Lesson Authoring and Delivery Application 1, you must
update PCD2 each time you update your PLATO software.

Updates (releases) are generally announced with a document
called the "Order Information Package" (OIP) which briefly
describes the major features of a new release. This document
comes from Control Data Software Manufacturing and Dist-
ribution (SMD) and is used to pre-order the planned update.
This pre-ordering ensures that you will receive the new
software as soon as it is available through SMD.

It is important to keep in mind that the release level of
PCD2 that you order must agree with the release level of your
PLATO application, or the level you are ordering.

You should use the "PCD2 Installation Procedures" to update
the PCD2 product with each major release.

CORRECTIVE CODE/MINI-RELEASES

If you should receive corrective code/or mini-releases which
include program A3S/PLATO, this should be installed first,
and then PCD2 will have to be reinstalled following the
"PCD2 Installation Procedures" in this document.

4.1.3   PCD2 Bill of Materials

PCD2 BILL OF MATERIALS

The PCD2 product has 3 aspects: central system components,
global PLATO files, and system-specific files and subfiles.

CENTRAL SYSTEM:

The following is a list of the records which reside on the
PCD2 installation tape (VSN = PLAT3D).

PSMDX   (PROC)   procedure which calls program PSMD from the
                    PLAT3D tape. This record is in the first
                    logical file on the installation tape.

PSMD   (ABS)   edits the PLATO binary to make PCD2 available.
                    This record is contained in the second
                    logical file on the tape.

GLOBAL PLATO FILES IN ACCOUNT "system":

The following PLATO files make up the global aspect of PCD2.
All of these files should exist on all systems, although they
will be inoperative on systems which have not installed PCD2.
These files are global, in that their content is the same on
all systems. These global files are included in each release
of the PLATO Application.

        FILE NAME   DESCRIPTION

        pcd2   PCD2 processor (main entry) lesson.

        pcd2aids   PCD2 "Aids" driver lesson.

        sCpcd2   PCD2 user's source file -use- lesson.

        sCpcd2e   PCD2 main editor control lesson.

        sCpcd2g   PCD2 graphics frame editor.

        sCpcd2n   PCD2 "Aids" information.

        sCpcd2k   PCD2 color table data.

        sCpcd2l   PCD2 line frames (Normal, Program, Reply,
                           Test, Cartoon, Animation) editor.

        sCpcd2p   PCD2 print / copy lesson.

        sCpcd2u   PCD2 system -use- lesson.

        sCpcd2x   PCD2 to PLATO Author Language / Micro
                           PLATO translator (part 1).

sCpcd2y   PCD2 to PLATO Author Language / Micro
                   PLATO translator (part 2).

sCsource   PCD2 nameset to lesson code transfer.

SYSTEM-SPECIFIC FILES / SUBFILES:

The following files and subfiles are unique to each system.
They contain system-specific information, and are thus not
shipped with each release. These files are delivered only
with the initial release of the PLATO Application. These
files should be kept in account "sCfiles".

     FILE/SUBFILE NAME   DESCRIPTION

     sCsysfile/pcd2   System-specific PCD2 access list.
                           Edited through lesson "sCcalutil".

5   Update Installation Procedures

SECTION 5 - UPDATE INSTALLATION PROCEDURES

INTRODUCTION

The following sections describe the procedure to be followed
to update your PLATO system from one major release to another
(for example, from PLATO Release 33 to PLATO Release 34).
If you are only upgrading your operating system and wish to
reinstall the same PLATO Release level on it, you should use
the "Reinstallation Procedures" section instead of this one.

If the PLATO Release you are updating to requires a different
level of the operating system, it is assumed that it has already
been installed.

When updating to a new level of the operating system, the
following PLATO-related items must be carried over from the
old operating system deadstart file to the new one.

1. Changes to deadstart file text records (CMRDECKs,
   EQPDECKs, LIBDECKs, IPRDECKs) as described in the
   "Deadstart File" section of the PLATO Configuration
   Handbook.

2. Procedure MFNX.

3. Procedure MFJX.

4. The PLATO configuration file.

If there are changes required to the above for the new PLATO
release, they will be described in the "Change deadstart
file text records" section of the following procedure. Do
not carry over any PLATO binaries or procedures other than
the ones listed above from one operating system to another.

This update installation procedure is designed to allow
analysts to upgrade their system from any PLATO release
more recent than Release 31 to the current release.

Each section which is dependent on the PLATO Release which
is currently running has a separate list of changes required
for each release. To use these update installation procedures,
you must start with the list describing the changes for the
PLATO release immediately following the release level currently
running on your system and proceed through all lists of changes
up to and including the release level being installed.

For example, if you are currently running release level 33
and you wish to upgrade to level 34, you should use only the
list of changes for PLATO R34 in each section. If your
system is currently running at release level 32 and you wish
to upgrade to release level 34, you must use the list of
changes for PLATO R33 and PLATO R34 in all sections.

5.1   PLATO Operations Changes

The following describes major changes to the operation or
installation of the PLATO application. This is included to
inform analysts perforaing the upgrade installation of the
PLATO application of important changes in operational
procedures.

CHANGES FOR PLATO R33

PLATO is now defined to be a NOS subsystem. The various
DSD commands to load PLATO have changed since parameters
cannot be specified on DSD commands. These commands are
shown below.

	Old DSD command	New DSD command
Standard PLATO version	x.manage.	plato.
Update PLATO version	x.manage(install=update)	plaupu.
Initial PLATO version	x.manage(install)	plains.

The control point at which MASTOR is to be loaded is specified
via the ENABLE IPRDECK entry or SUBSYST command, ie. ENABLE,PLA,1.

In order to process the new DSD commands, procedure files exist
under user index 377777 on the system pack with the same name
as the DSD commands. Local copies of these procedures may be
made and modified under different names and DSD will automatically
process the proper procedure when that name is entered to load
PLATO. The only requirement for the names is that they begin
with the letters "pla" and are legal NOS file names.

The procedure "spjx" was used on previous levels to maintain
compatibility between NOS 2.1 and NOS 2.2. This procedure
has now been replaced by the *proute* control statement in all
PLATO procedures. If local procedures use *spjx*, they must
be changed.

Procedures "mastx", "manage", "gogo" and "spjx" are no longer
used and may be deleted from deadstart tapes and LIBDECKS.

To reduce the dependence of the PLATO system on the operating
system level, the 8-display option for the right screen when
using *x.console* has been removed.

The PLATO terminal simulator for the system console has
been changed to squeeze the entire left-screen display
into the upper three-fifths of the screen. This has been
done as part of the support for the CDC 721 console on the
810/830 mainframes.

CHANGES FOR PLATO R34

All documentation for the PLATO application has been completly

rewritten for this release.   Major changes were:

- The Installation Guide was reorganized to make the
  initial installation and update installation process
  simpler.

- Changes were made to the Installation Guide to allow
  for running a given release on more than one level of
  the operating system.

- The PLATO Bill of Materials section of the Operations
  Guide was moved to the Installation Guide.

- All installation-related information was moved from
  the Operations Guide to the Installation Guide.

- The information on system deadstart files was moved
  from the Operations Guide to a new Configuration
  Handbook.

- The PLATO configuration file section of the Operations
  Guide was moved to the Configuration Handbook.

- The Operations Guide was reorganized and a new index
  was written to make it easier to find a particular
  operational procedure.

- All configuration-related information in the Operations
  Guide was moved to the new Configuration Handbook.

5.2   Update Installation procedures

The following is a list of the steps to be taken to upgrade
the PLATO application software. Each of these steps is
discussed in detail in the following sections.

1. Execute pre-installation cleanups.

2. Obtain files from installation tape.

3. Load release master files.

4. Reinstall NAM and CCP.

5. Reinstall selected NOS programs.

6. Change NOS validation files.

7. Change deadstart file text records.

8. Change PLATO configuration file.

9. Build new deadstart file.

10. Clear operator action queues.

11. Complete file dumps.

12. Deadstart on new deadstart file.

13. Change MFNX procedure.

14. Load PLATO via PLAINS DSD-command.

15. Change existing access lists.

16. Change existing PLATO files.

17. Destroy obsolete files.

18. Install new system lessons.

19. Initialize binary master file(s).

20. Reload PLATO.

21. Run required conversion programs.

22. Run file installation check.

23. Change network system table.

24. Reinstall AUTHORS database.

25. Change local AIDS files.

26. Install new optional features.

27. Reinstall Add-on Products.

28. Post-installation cleanups.

29. Dump all master files.

5.2.1   Execute pre-installation cleanups.

There are several checkouts and cleanups that should be
done as part of normal operations. The procedures in the
following sections should be done before the installation
to resolve possible existing problems and to minimize the
cleanups necessary after installation.

1. Execute the following cleanup programs in lesson
    "accountu" for all accounts. Errors are logged in
    "accerrlog".

    a. "Account directory and file checks"

    b. "Search for files not in any account"

2. Execute the "Search for duplicate files" option in
    lesson "installu" for all active master files.

3. Execute the "File Directory Verification Check" option
    in lesson "utility". If the "utility" runner is turned
    on, it should not be necessary to run this manually.

4. If any errors are found as a result of these checks,
    they will be logged. Print the corresponding log file
    and correct any errors before proceeding.

5. A new software release will usually require more disk
    space. The amount required may vary from a few parts
    to over a hundred. Make sure there is adequate space
    on your required master files to handle the new soft-
    ware. This may involve adding another master file to
    your system if you are currently at or close to the
    limit of available space.

6. During the installation process, a temporary master
    file must be loaded onto your disk packs. You must
    make sure there is enough disk space available on
    your disk packs to create one 3530-part master file.
    The number of free tracks required for this master
    file is shown below for each supported device type.
    In addition, there must also be half as many tracks
    of temporary file space available.

    | DD | 14450 tracks |
    | D3 | 3446 tracks |
    | DI | 22640 tracks |
    | DJ | 19670 tracks |
    | DK | 21750 tracks |
    | DL | 16670 tracks |
    | DM | 3115 tracks |
    | DN | 3110 tracks |

7. When installing PLATO Release 34 or above, you may
    need to lengthen common "link" in file "sysfile"

before beginning the installation. Use the following
procedure to determine if you will need to lengthen
this common.

a. Edit file "sysfile".

b. Press "+" until you find the block named "link".

c. Press the letter which appears next to block "link"
   to edit it.

d. Near the top of the next display, you will see the
   current length of the common displayed.

e. If this length is greater than (1 + 10 * the value
   of the "netas" configuration file entry), you will
   not need to lengthen this common.

If the current length of the common is too short, you
should use the following procedure to lengthen it.

a. At a convenient time, back out all users. This is
   necessary to prevent a user from writing a common
   into file "sysfile" while the file is reorganized.

b. Edit file "sysfile".

c. Press "+" until you find the block named "link".

d. Press the letter which appears next to block "link"
   to edit it.

e. Press SHIFT-LAB for "other options".

f. Choose the "change length of common" option.

g. Choose a new length which is the lowest multiple of
   320 greater than (1 + 10 * the value of the "netas"
   configuration file entry).

5.2.2   Obtain files from installation tape.

This step will load the files which make up the PLATO
application from the first installation tape.

Mount the first installation tape (the one labeled PLAT1A)
on an available tape drive. Make sure the tape drive is
available by checking the console E,T-display.

Enter the following commands on the computer console:

```
            X.DIS.
            FAMILY(family name)   if using alternate family
            USER(sys,password)    sys is the PLATO system
                                  user name (usually SYS)
            LABEL(TAPE,VSN=PLAT1A)
            GTR(TAPE,UPGRADE)PROC/UPGRADE
            BEGIN(UPGRADE,UPGRADE,PPW=ps1,SPW=ps2)


            where  ps1 = password for user name PLATOMF
                   ps2 = password for user name SYSTEMX
```

This procedure creates or replaces the following files.

User sys:	PLABINS	PLATO release binaries
	PLANOS	modifications to NOS
	PLANAM	modifications to NAM
	PLACCP	modifications to CCP
	LIBDIR	LIBDECK entries for PLATO programs
	LOADMF	procedure to load installation master file
	BKSPRU	program used to load installation master file
User PLATOMF:	ISTLFC0	terminal resident load file
	ISTLFC1	terminal resident load file
	ISTLFC3	terminal resident load file
	ISTLFC8	terminal resident load file
User SYSTEMX:	PLATO	PLATO load procedure
	PLAINS	PLATO load procedure
	PLAUPD	PLATO load procedure

5.2.3   Load release master files.

Use the PLATO procedure LOADMF to copy the PLATO master files from the installation tapes to disk.

The formats of this procedure call are:

              LOADMF(MF=mf)   default family

              LOADMF(MF=mf,FAMILY=fn)   alternate family

              LOADMF(MF=mf,PN=pn,R=dt)   auxiliary device

        where:

              mf = master file name
              fn = family name
              pn = auxiliary pack name
              dt = device type if PN argument is used

Mount each installation tape as it is requested via the E,P-display.

    Example: Default family.

If you are using the default family device for PLATO master files, enter the following on the computer console.

            X.DIS.
            USER(sys,password)
            GET(LOADMF)
            LOADMF(MF=SYS1)

    Example: Alternate family.

If you are using an alternate family device for PLATO master files, enter the following on the computer console.

            X.DIS.
            USER(sys,password)
            FAMILY(family name)
            GET(LOADMF)
            LOADMF(MF=SYS1,FAMILY=family name)

    Example: Auxiliary devices.

To load the PLATO master files onto an auxiliary device named "PLATOA" of type "DL", enter the following on the computer console.

            X.DIS.
            USER(sys,password)
            FAMILY(family name) if using alternate family
            GET(LOADMF)

LOADHF(HF=SYS1,PN=PLATOA,R=DL)

LOADHF(HF=SYS1,PN=PLATOA,R=DL)

5.2.4   Reinstall NAM and CCP.

Inspect tne contents of files PLANAM and PLACCP under the
PLATO system user name. If either of these files contains
modsets, it will be necessary to reinstall NAM and CCP
with these modsets. Refer to the NOS Installation Handbook
for instructions.

The PLATO application requires only the standard asynchronous
TIP in the CCP build procedure.

Save any modsets in PLANAM and PLACCP so they may be used
again if NAM and CCP must be reinstalled in the future.

5.2.5   Reinstall selected NOS programs.

Inspect the contents of file PLANOS under the PLATO system
user name. If this file contains modsets, it will be
necessary to reinstall the affected programs with these
modsets. Refer to the NOS Installation Handbook for more
information.

By looking at the contents of this file, determine what NOS
programs will need to be reassembled. Only the programs
which have modifications will need to be reassembled.

Be sure that, any time in the future, when these programs
are reassembled, that these modifications are included.

Save the binaries produced for replacement on your deadstart
file after all the PLATO materials are ready to be installed.

Save any modsets in PLANOS so they may be used again if the
affected NOS programs must be reinstalled in the future.

5.2.6   Change NJS validation files.

Changes to the format or content of the NJS validation files
may affect the user names used by the PLATO application or
by users of the PLATO application.

You should make all changes listed for all NOS levels
beginning with the one following the level you are updating
from through the NOS level you are updating to.

CHANGES FOR NOS 2.3 LEVEL 617

    The user names specified by the "subun" and "prtun" PLATO
    configuration file entries and all user names which
    are permitted to submit system origin jobs through PLATO
    must be changed to allow them to submit system service
    class jobs. Use the following procedure to make this
    change.

                X.MODVAL.
                * assign the K-display to the job.
                K.U,sys.
                K.VM=ALL.
                K.END.
                K.END.

CHANGES FOR NOS 2.4.1 LEVEL 630

    There are no NOS validation file changes necessary for
    this NOS level.

5.2.7   Change deadstart file text records.

Changes in the format or content of the NOS deadstart file
text records (CMRDECKs, EQPDECKs, LIBDECKs, IPRDECKs) may
be required because of changes in NOS or in the PLATO
application software.

This can be done using O26 or some other text editor available
on your system. Instructions for using O26 may be found in
the NOS V2 System Programmer's Instant manual.

Save the modified text records for replacement on your dead-
start file after all the PLATO materials are ready to be
installed.

The following example assumes you are using O26 as the text
editor and that you are using the first (O) copies of the
text records (CMRDO0, EQPDO0, LIBDO0, IPRDO0).

    Example: Modifying a text record.

            X.DIS.
            FAMILY(family name)   if using alternate family
            USER(sys,password)
            COMMD0(SYSTEM)
            GTR(SYSTEM,CMRDECK)TEXT/CMRDO0
            O26.
            READ.CMRDECK.
            * make required changes.
            DIS.
            PURGE(CMRDECK/NA)
            SAVE(CMRDECK)

CHANGES TO CMRDECKS

The following change should be made to all CMRDECKs on the deadstart file. If there are on-line copies of these decks, be sure to change them as well. See the "Deadstart File" section in the PLATO Configuration Handbook for a list of all PLATO-related CMRDECK entries.

Change the version line if the release level of the PLATO software is referenced. Also, change the NOS version line, if updating the operating system.

CHANGES TO EQPDECKS

There are no required EQPDECK changes.

CHANGES TO LIBDECKS

The following changes should be made to all LIBDECKs on the deadstart file. If there are on-line copies of these decks, be sure to change them as well. See the "Deadstart File" section in the PLATO Configuration Handbook for a list of all PLATO-related LIBDECK entries.

CHANGES FOR PLATO R33

Add entry:

*CH   PP/4PA,4PB

Delete entries:

*PROC   SPJX
*PROC   MANAGE
*PROC   MASTX

CHANGES FOR PLATO R34

Delete entry:

*CH   PP/4QV

CHANGES TO IPRDECKS

The following changes should be made to all IPRDECKs on the deadstart file. If there are on-line copies of these decks, be sure to change them as well. See the "Deadstart File" section in the PLATO Configuration Handbook for a list of all PLATO-related IPRDECK entries.

CHANGES FOR PLATO R33

Add entry:

ENABLE,PLA,cp.

where cp = the control point where the MASTOR job
is to be run.

CHANGES FOR PLATO R34

There are no required IPRDECK changes for this
release.

5.2.8   Change PLATO configuration file.

The following changes should be made to the PLATO configuration
file on the deadstart file. If there are on-line copies of
this file, be sure to change them as well. Refer to the
PLATO Configuration Handbook for a complete list of all PLATO
configuration file entries.

You may use O26 or any other text editor available on your
system to make these changes.

Save the modified configuration file for replacement on your
deadstart file after all the PLATO materials are ready to be
installed.

The following procedure may be used to modify the PLATO
configuration file.

```
          X.DIS.
          FAMILY(family name)  If using alternate family
          USER(sys,password)
          COMMON(SYSTEM)
          GTR(SYSTEM,CONFIG)TEXT/CONFIG
          O26.
          READ.CONFIG.
          * make required changes.
          DIS.
          PURGE(CONFIG/NA)
          SAVE(CONFIG)
```

     CHANGES FOR PLATO R33

          Delete the following entries, if present.

               lst2x
               plox

          Systems using 800-series mainframes must set the "ncwb"
          parameter to zero. Systems using 700-series mainframes
          should not change this parameter.

     CHANGES FOR PLATO R34

          Delete the following entries, if present.

               antsz
               binds
               bplat
               clxld
               comsr
               dsfos
               fovl
               ljtnm
               mdir

nfilb
nvers

The interpretation of the "jbnks" entry has changed
with this release. This now specifies the total
number of judge buffers in the system instead of the
number of judge buffers for each executor in a multi-
executor configuration. All systems running more
than one executor will need to increase the value of
the "jbnks" entry. This can be done by multiplying
the number of judge buffers formerly used by the
number of executors used, keeping in mind the limit
on the maximum number of judge buffers allowed and
the recommended value, as documented in the PLATO
Configuration Handbook.

The "cbufl" keyword must be changed to "cblth". The
interpretation of the value of this keyword has also
changed so that it must be specified in units of the
TUTOR block length instead of an absolute length.
The new value should be set to the old value divided
by 320.

The "nterm" keyword must be changed to "users". The
interpretation of this keyword has not changed.

The "syslk" keyword must be changed to "sysac". The
interpretation of this keyword has not changed.

The default values of the following keywords have
been changed. They are now dependent on the amount
of memory available. See the PLATO Configuration
Handbook for more information.

cblth	emgr3	emgr4	fastl	foral
lesns	nparc			

The default values of the following keywords have
been changed to 0. When the value of these key-
words is set to zero, the system will automatically
set the value based on the number of users defined
by the "users" keyword. See the PLATO Configuration
Handbook for more information.

nlob	quesz

5.2.9   Build new deadstart file.

Build a new deadstart file which includes the following:

1.   The NAM which was reinstalled with the modsets in
      file PLANAM, if any were present.

2.   The NOS programs which were reinstalled with the
      modsets in file PLANOS, if any were present.

3.   The CMRDECK, EQPDECK, LIBDECK and IPRDECK which
      were modified earlier in this procedure.

4.   The PLATO configuration file which was modified
      earlier in this procedure.

5.   The PLATO release binaries which were loaded from
      the installation tape earlier in this procedure.

The following is a sample procedure which could be used to
build the deadstart file.

```
                    X.DIS.
                    FAMILY(family name)  if using alternate family
                    USER(sys,password)
                    COMMON(SYSTEM)
                    ATTACH(nam binary file)
                    ATTACH(nos binary file)
                    GET(CARDECK)
                    GET(EQPDECK)
                    GET(LIBDECK)
                    GET(IPRDECK)
                    GET(CONFIG)
                    ATTACH(PLABINS)
                    UNLOAD(DIR)
                    NOTE(DIR,NR).*FILE nam binary file
                    NOTE(DIR,NR).*FILE nos binary file
                    NOTE(DIR,NR).*FILE CMRDECK
                    NOTE(DIR,NR).*FILE EQPDECK
                    NOTE(DIR,NR).*FILE LIBDECK
                    NOTE(DIR,NR).*FILE IPRDECK
                    NOTE(DIR,NR).*FILE CONFIG
                    NOTE(DIR,NR).*FILE PLABINS
                    PACK(DIR)
                    REWIND(*)
                    LIBEDIT(F=SYSTEM,I=DIR)
                    REWIND(*)
                    ASSIGN,50,T,F=I.    or    INSTALL(NEW=EQxx)
                    COPY(NEW,T,V)
```

5.2.1C   Clear operator action queues.

The format of various operator action queues may change
with a new release. Therefore, all requested actions should
be performed just prior to the installation, thus clearing
the queues.

1.   Print all requests in the print request queue.

2.   Perform any file backup requests.

3.   Complete any archive/retrieval requests.

5.2.11   Complete file dumps.

Before you begin the installation, do a complete dump of
all disk packs. If something goes wrong, good backups must
be available to allow retreating to the previous release,
if necessary.

5.2.12   Deadstart on new deadstart file.

Deadstart using the new deadstart file for this release.
DO NOT try to bring up PLATO yet; some changes must be
made first.

5.2.13   Change MFNX procedure.

Procedure MFNX must be changed to attach the new system
lesson master file, SYS1. This master file must be the
first master file attached so that files on this master
file which duplicate ones already on the system are used
when the PLATO application is loaded. You may also need to
change or add a RESOURC command if the new master file will
be on an auxiliary pack other than those already used by
required master files.

In addition, check to be sure that the number of required
master files PLUS the one installation master file does not
exceed the value of the "ndsus" PLATO configuration file
entry. If necessary, obtain a copy of your configuration
file from the deadstart file, edit it to increase "ndsus" as
needed.

Use SYSEDIT to make these changes to MFNX and the PLATO
configuration file part of the running system.

You may use a procedure similar to the following to make
these changes to MFNX and your PLATO configuration file.

```
                X.DIS.
                COMMON(SYSTEM)
                GTR(SYSTEM,MFNX)PROC/MFNX
                *
                *   At this point, procedure MFNX is in NOS
                *   local file MFNX.  Use 026 to edit file MFNX
                *   to attach new master files.
                *
                GTR(SYSTEM,CONFIG)TEXT/CONFIG
                *
                *   Use 026 to change the value of the "ndsus"
                *   entry in local file CONFIG, if necessary.
                *
                NOTE(DIR,NR).*FILE MFNX
                NOTE(DIR,NR).*FILE CONFIG
                PACK(DIR)
                SYSEDIT(I=DIR,B=0)
```

5.2.14   Load PLATO via PLAINS.

You will now load the PLATO application via the PLAINS
DSD-command.

1.   If your system is to use the PLATO ASCII network, NAM
      must be brought up at this point.

2.   Load the PLATO application by typing the following at
      the computer console:

           PLAINS.

      This console command is used only during PLATO application
      installations. It performs the following functions:

      a.   Alters the "cpspd" configuration entry so that the
           PLATO application will allow more foreground processing
           time for running installation programs.

      b.   Prevents groups other than "s", "p" and "convertc"
           from signing onto the system.

      When updating from PLATO release 34.1 or earlier, you
      may see the dayfile message "network table obsolete".
      This message may be ignored during installation since
      the problem being reported will be corrected by a
      conversion to the network system table later in this
      process.

3.   Now, sign on to the PLATO application with a PLATO
      terminal, using your "p" sign-on.

5.2.15   Change existing access lists.

The PLATO application uses several access lists to control
user access to privileged system lessons and options. These
access lists are maintained by the local system. Sometimes
changes must be made to these access lists at the time of
the installation of a new PLATO release.

The following general procedure can be used to update these
access lists.

   1.   Edit file "sOsysfile".

   2.   Press "+" until the name of the access list to be
        updated is seen on the block display of the editor.

   3.   Edit the access list which must be updated by
        pressing the letter which appears next to the name
        of the access list on the block display.

   4.   Select the "edit user option descriptions" option.

   5.   Now, select the "copy descriptions from another
        access list" option.

   6.   Enter "sOsubfil" as the name of the file from which
        to copy.

   7.   Enter the name of the block to be updated as the
        block from which to copy.

   8.   Press "y" to adjust authorizations. (If this is not
        done, all entries in the access list will be of type
        "Special", instead of "Director" or "Operator", etc.).

   9.   Press SHIFT-HELP to copy the descriptions.

   10. Any sign-on listed in the access list which was
        previously of type "Special" still retains that
        classification. You may want to check the individual
        options assigned to that sign-on, and possibly change
        some of them.

CHANGES FOR PLATO R33

     There are no changes needed for this PLATO release.

CHANGES FOR PLATO R34

     There are no changes needed for this PLATO release.

5.2.16   Change existing PLATO files.

For each new PLATO release, there may be changes needed for
existing system database files.

CHANGES FOR PLATO R33

Two new access lists must be added to the system database
file "s0sysfile". One access list is only used if the
PCD2 on-line authoring product is installed, but all systems
must make this change. The other is used by the Calendar
utility, a new feature with Release 33.

Use the following procedure to install these access lists.

1. Edit file "s0sysfile".

2. Choose to add a new block to this file by pressing
   the SHIFTed letter of the block you want the new
   block to follow. You may have to lengthen "s0sysfile"
   through accounts options if there is not enough room
   to add new blocks.

3. Choose the option to copy blocks from another file.

4. Enter "s0subfil" as the file to copy from.

5. Choose block "pcd2" as the block to copy. You may
   get a message at this point which indicates there
   is not enough space in the file to copy all of the
   blocks. If this happens, lengthen the file through
   account options and start this procedure over at
   step 1.

6. After the blocks are copied, press BACK until you
   are out of the file.

7. Repeat this procedure, but use the name "s0schaosys"
   as the name of the blocks to be copied.

CHANGES FOR PLATO R34

One new access list must be added to the system database
file "s0sysfile". This access list is used to control
the options of lessons "lpedit", "s0netsys" and "s0calutil".

Use the following procedure to install this access list.

1. Edit file "s0sysfile".

2. Choose to add a new block to this file by pressing
   the SHIFTed letter of the block you want the new
   block to follow. You may have to lengthen "s0sysfile"
   through accounts options if there is not enough room

to add new blocks.

3. Choose the option to copy blocks from another file.

4. Enter "s0sysfill" as the file to copy from.

5. Choose block "lpedit" as the block to copy. You
   may get a message at this point which indicates
   there is not enough space in the file to copy all
   of the blocks. If this happens, lengthen the file
   through account options and start this procedure
   over at step 1.

6. After the blocks are copied, press BACK until you
   are out of the file.

If access list "s0calutil" exists in file "s0sysfile", it
may be deleted as it is no longer used.

Use the following procedure to delete this access list.

1. Edit file "s0sysfile".

2. Press "+" until a block named "s0calutil" appears
   on the block directory display.

3. Press the letter which appear next to the first
   of these blocks.

4. Press SHIFT-HELP to delete the access list.

5.2.17   Destroy obsolete files.

With each release, there are certain files which become
obsolete and are no longer needed. These files should be
destroyed to save space. This is done through lesson
"transfer" as follows:

1. Execute lesson "transfer".

2. Choose the "Destroy Obsolete System Courseware"
   option.

3. Choose "obsolete system files".

4. Press SHIFT-HELP to begin the operation.

5. Enter the name of a student datafile to use as an
   error log or press NEXT to use the default log file
   when requested.

5. If a particular file did not exist on the system, it
   will be automatically skipped and entered into the
   error log along with any other error that may have
   occurred. If log entries were made, the log should
   be scanned to see if anything unusual happened.

5.2.18   Install new system lessons.

Install new versions of system lessons as follows:

1. Execute lesson "transfer".

2. Choose the "install new courseware" option.

3. Enter "sys1" and press NEXT.

4. You will be asked for the name of the first file to process. Press NEXT to begin with the first file on the master file.

5. Press "n" to specify that customer files are NOT to be replaced.

6. Press "n" to specify that errors are NOT to be automatically skipped.

7. Old system files will be automatically destroyed if they are to be replaced. If a new file duplicates a file found in a non-system account, "transfer" will stop. The file in the non-system account must be renamed.

8. If an error occurs, a message will be displayed. Correct the problem reported and continue. There should be no errors if the account cleanup programs were run prior to the installation.

9. It may be necessary to increase the number of disk parts allocated to account "system" during this installation. If so, set the parts allocated to unlimited.

5.2.19   Initialize binary master file(s).

All binary master files should be reinitialized. While the
system will still function properly without doing this, the
"utility" runner program will generate errors for any old
binary file it finds.

1. Execute lesson "ldr".

2. Choose the "Master File Initializations" option.

3. Enter the binary master file name.

4. Press NEXT to leave the master file name the same.

5. Press NEXT to leave the master file type the same.

6. Press "y" to initialize all files on the master file.

7. Press SHIFT-HELP to initialize the master file.

8. Repeat this procedure for all binary master files.

5.2.20   Reload PLATO.

All new system files have now been transferred to the
required master files. Therefore, master file SYS1
is no longer needed. Reload without it as follows:

1. Take the PLATO application down.

2. Enter the following at the computer console to
   restore procedure MFNX and the PLATO configuration
   file to their original state.

      X.SYSEDIT(R)

3. Reload the PLATO application again using the PLAINS
   DSD command.

5.2.21   Run required conversion programs.

For some releases, conversion programs must be run to update
user files so that they will work on the new release of the
application. The conversions which must be run for this
release are described below. Unless told otherwise below,
run the conversions according to the procedures described
in the "File Conversions" section of the PLATO Operations
Guide.

If there are no conversion programs to be run or after all
conversion programs are run, you must make sure that the
update levels of new files are being set correctly. This
is done as follows:

   1. Execute lesson "lpedit".

   2. Press NEXT on the first display for other options.

   3. Select the "Update levels for new files" option.

   4. Check the list of file types and update levels with
       the list below. The update levels of file types not
       listed below should be zero.

File type	Update level
a	5
f	6
g	1
h	2
i	2
k	1
l	2
o	2
q	1

   4. If there are any differences, enter the file type and
       the new update level. Update levels will always
       increase, do not change an update level to a lower
       value.

CONVERSIONS FOR PLATO R33

    There are no required conversion programs for this release.

CONVERSIONS FOR PLATO R34

    There are no required conversion programs for this release.

5.2.22   Run file installation check.

All system lessons have now been replaced with the new
versions for this release. Test for proper installation
as follows:

1. Execute lesson "installu".

2. Choose "Special lesson list check" option.

3. Press NEXT to scan all active master files.

4. If any file is reported as missing, report it via
PSR. See the "Problem Reporting" section of the
PLATO Operations Guide for information on how to do
this.

5.2.23   Change network system table.

For each PLATO release, there may be required changes to tne
network system table to add or delete systems for which an
AUTHORS database was or is to be installed.

Use tne following general procedure to update the network
system table.

1.   Execute lesson "lpedit".

2.   Choose the "Network management" option.

3.   On the following display, choose tne "System table" option.

4.   Choose tne "Add a new system" or "Delete a system" option
     as instructed.

5.   When adding a new system, you must enter the following.

          a.   tne name of the new system
          b.   tne routing identifier (RID) of the new system
          c.   the link type for the new system (generally,
               this will be "not linked")
          d.   tne "AUTHORS database available" flag

CHANGES FOR PLATO R33

     There are no changes necessary for this release.

CHANGES FOR PLATO R34

     The network system table common in file "s0sysfile"
     must be converted when installing this release. Use tne
     following procedure to convert this common.

     a. Execute lesson "s0netsys".

     b. Press SHIFT-HELP when requested. When the conversion
        is complete, you will be sent to the main index.

     c. Choose tne "Update the EM copy of the system table"
        option and press SHIFT-HELP when requested.

5.2.24   Reinstall AUTHORS database.

Activate the new AUTHORS database files as follows:

   1. Execute lesson "authors".

   2. Press LAB for director options.

   3. Choose the "Database management options".

   4. Choose the "active systems" option.

   5. Press SHIFT-NEXT to install all systems' databases.

   6. When asked for a file prefix, just press NEXT.

   7. At the end of the installation, the option to turn
       AUTHORS back on will be given. Press NEXT to do so.

5.2.25   Change local AIDS files.

Each PLATO release may require changes to the two AIDS files
which are maintained by each system, "aUss1" and "aUpsoless".

CHANGES FOR PLATO R33

    There are no changes necessary for this release.

CHANGES FOR PLATO R34

    There are no changes necessary for this release.

5.2.26   Install new optional features.

Each PLATO release may introduce new optional features for
which further installation procedures are needed. See the
"Optional Installation Procedures" section for more information
and installation procedures for the features listed below.

These new optional features may be installed at this time
or at any future time. If you do not wish to install any
of these features at this time, or, when you have completed
installing them, you should continue with the next section.

CHANGES FOR PLATO R33

    There are no new optional features for this release.

CHANGES FOR PLATO R34

    The PLATO Inter-system Link was added at this release.

5.2.27   Reinstall Add-on Products

At this time, any optional "add-on" PLATO products, such as
the PCD2 On-Line Authoring module, which have been purchased
must be reinstalled.

Add-on products which are to be installed must match the PLATO
level you are updating to.

See the "Add-on Products" section of this document for
update installation instructions.

When you have completed installing all add-on products, you
should continue with the next section.

5.2.28   Post-installation cleanups.

The items listed below are things which must be done fol-
lowing a new PLATO release update, but are not critical.
They may be done as part of the actual installation or
during the following week. If you do not wish to do these
cleanups now, continue on to the next section.

PURGE RELEASE MASTER FILES

The release master file, SYS1, should be purged to
recover the disk space it occupies.

PURGE INSTALLATION-ONLY DISK FILES

The following NOS permanent files may be purged to save
space since they are needed only during the installation
process.

PLABINS	PLATO release binaries
CONFIG	PLATO configuration file
MFNX	procedure to attach master files
MFDX	procedure to dump master files
LIBDIR	LIBDECK entries for PLATO programs
LOADMF	procedure to load installation master files
BKSPRU	program used to load installation master files

CHANGE ON-LINE CONFIGURATION FILE / PROCEDURES

If on-line copies of the PLATO configuration file,
deadstart file text records or PLATO procedures MFNX
and MFDX are used, make the same changes to these copies
that were made when the new deadstart file was built,
if this has not already been done.

REASSEMBLE PROGRAMS IF THERE HAS BEEN A NOS CHANGE

If the operating system has changed with this release,
be sure to reassemble any private programs in use on the
system.

5.2.29   Dump all master files.

The installation is now complete.

All master files should be dumped for protection against
disk problems happening before the next regularly scheduled
dump.

Once the dumps are complete, open the system to users by
taking the PLATO application down and reloading it via the
PLATO DSD command.

6   Mini-Release Installation Procedures

SECTION 6 - MINI-RELEASE INSTALLATION PROCEDURES

INTRODUCTION

The following sections describe the procedure to be followed
to update your PLATO system when you receive a "mini-release"
shipment. Mini-release shipments are used to distribute
fixes for critical problems which cannot wait until the next
major release.

This procedure may be used to update from any mini-release
version of the PLATO system to any other (for example, from
PLATO Release 34.1 to PLATO Release 34.2).

The following is a list of the steps to be taken to install
a mini-release of the PLATO application software. Each of
these steps is discussed in detail in the following sections.

    1. Obtain files from mini-release binaries tape.

    2. Perform special instructions.

    3. Build new deadstart file.

    4. Load mini-release master file.

    5. Deadstart on new deadstart file.

    6. Load PLATO via PLAINS DSD-command.

    7. Change existing PLATO files.

    8. Initialize binary master file(s).

    9. Reinstall Add-on Products.

  10. Post-installation cleanups.

  11. Reload PLATO.

6.1   Obtain files from installation tape.

This step will load the files which make up the PLATO
application from the mini-release binaries tape.

Mount the first installation tape (the one labeled PLAT1A)
on an available tape drive. Make sure the tape drive is
available by checking the console E,T-display.

Enter the following commands on the computer console:

```
X.DIS.
FAMILY(family name)   if using alternate family
USER(sys,password)    sys is the PLATO system
                      user name (usually SYS)
LABEL(TAPE,VSN=PLAT1A)
GTR(TAPE,UPGRADE)PROC/UPGRADE
BEGIN(UPGRADE,UPGRADE,PPW=ps1,SPW=ps2)


where   ps1 = password for user name PLATOMF
        ps2 = password for user name SYSTEMX
```

This procedure creates or replaces the following files.

```
User sys:       PLABINS      PLATO release binaries
                PLANOS       modifications to NOS
                PLANAM       modifications to NAM
                PLACCP       modifications to CCP
                LIBDIR       LIBDECK entries for PLATO programs
                LOADMF       procedure to load installation
                             master file
                BKSPRU       program used to load installation
                             master file

User PLATOMF:   ISTLFC0      terminal resident load file
                ISTLFC1      terminal resident load file
                ISTLFC3      terminal resident load file
                ISTLFC8      terminal resident load file

User SYSTEMX:   PLATO        PLATO load procedure
                PLAINS       PLATO load procedure
                PLAUPU       PLATO load procedure
```

6.2   Perform special instructions.

The following may be changed by a specific mini-release,
but are not generally needed for all mini-releases. Make
these changes only if told to do so in the mini-release memo
included with the mini-release tapes or in the PLATO Software
Release Bulletin.

    a. Reinstall NAM and CCP.

    b. Reinstall selected NOS programs.

    c. Change PLATO configuration file.

6.3 Build new deadstart file.

Build a new deadstart file which includes the PLATO binaries
which were loaded from the installation tape earlier in this
procedure and any items changed under the section on special
instructions.

The following is a sample procedure which could be used to
build the deadstart file.

```
X.DIS.
FAMILY(family name)  if using alternate family
USER(sys,password)
COMMON(SYSTEM)
ATTACH(PLABINS)
UNLOAD(DIR)
NOTE(DIR,NR).*FILE PLABINS
PACK(DIR)
REWIND(*)
LIBEDIT(P=SYSTEM,I=DIR)
REWIND(*)
ASSIGN,50,T,F=I.    or    INSTALL(NEW=EQxx)
COPY(NEW,T,V)
```

6.4   Load mini-release master files.

Load the mini-release master file tapes as described in the
"Courseware Installation" section of the PLATO Operations
Guide.

You must be sure that no users are signed in while this
installation is being done. Since system files are being
replaced, users may begin using the new files before the
installation is complete and results are unpredictable.

When the installation is complete, take the PLATO system
down.

6.5   Deadstart on new deadstart file.

Deadstart using the new deadstart file.

6.6   Load PLATO via PLAINS.

You will now load the PLATO application via the PLAINS
DSD-command.

1.   If your system is to use the PLATO ASCII network, NAM
      must be brought up at this point.

2.   Load the PLATO application by typing the following at
      the computer console:

            PLAINS.

      This console command is only during PLATO application
      installations. It performs the following functions:

      a.   Alters the "cpspd" configuration entry so that the
            PLATO application will allow more foreground processing
            time for running installation programs.

      b.   Prevents groups other than "s", "p" and "convertc"
            from signing onto the system.

      When updating from PLATO release 34.1 or earlier, you
      may see the dayfile message "network table obsolete".
      This message may be ignored during installation since
      the problem being reported will be corrected by a
      conversion to the network system table later in this
      process.

3.   Now, sign on to the PLATO application with a PLATO
      terminal, using your "p" sign-on.

6.7   Change existing PLATO files.

The following may be changed by a specific mini-release,
but are not generally needed for all mini-releases. Make
these changes only if told to do so in the mini-release memo
included with the mini-release tapes or in the PLATO Software
Release Bulletin.

        a. Change existing access lists.

        b. Change existing PLATO database files.

6.8   Initialize binary master file(s).

All binary master files should be reinitialized.   While the
system will still function properly without ooing this, the
"utility" runner program will generate errors for any old
binary file it finds.

    1. Execute lesson "ior".

    2. Choose the "Master File Initializations" option.

    3. Enter the binary master file name.

    4. Press NEXT to leave the master file name the same.

    5. Press NEXT to leave the master file type the same.

    6. Press "y" to initialize all files on the master file.

    7. Press SHIFT-HELP to initialize the master file.

    8. Repeat this procedure for all binary master files.

6.9   Reinstall Add-on Products

At this time, any optional "add-on" PLATO products, such as
the PCD2 On-Line Authoring Module, which have been purchased
must be reinstalled.

Add-on products which are to be installed must match the MAJOR
release level of the PLATO system being run.

See the "Add-on Products" section of the PLATO Installation
Guide for update installation instructions.

6.10   Post-installation cleanups.

The items listed below are things which may be done fol-
lowing a PLATO reinstallation, but are not critical. If
you do not wish to do these cleanups now, continue on to
the next section.

PURGE INSTALLATION-ONLY DISK FILES

The following NOS permanent files may be purged to save
space since they are needed only during the installation
process.

PLABINS	PLATO release binaries
CONFIG	PLATO configuration file
MFNX	procedure to attach master files
MFDX	procedure to dump master files
LIBDIR	LIBDECK entries for PLATO programs
LOADMF	procedure to load installation master files
BKSPRU	program used to load installation master files

6.11   Reload PLATO.

The mini-release installation is now complete.

Open the system to users by taking the PLATO application
down and reloading it via the PLATO DSD command.

7   Reinstallation Procedures

SECTION 7 - REINSTALLATION PROCEDURES

INTRODUCTION

The following sections describe the procedure to be followed
to reinstall your PLATO system on a new level of the operating
system. If you are also upgrading your PLATO system to a new
release level as well as upgrading your operating system, you
should use the "Upgrade Installation Procedures" section instead
of this one.

When updating to a new level of the operating system, the
following PLATO-related items must be carried over from the
old operating system deadstart file to the new one.

   1. Changes to deadstart file text records (CMRDECKs,
      EQPDECKs, LIBDECKs, IPRDECKs) as described in the
      "Deadstart File" section of the PLATO Configuration
      Handbook.

   2. Procedure MFNX.

   3. Procedure MFDX.

   4. The PLATO configuration file.

Do not carry over any PLATO binaries or procedures other than
the ones listed above from one operating system to another.

The following is a list of the steps to be taken to reinstall
the PLATO application software. Each of these steps is
discussed in detail in the following sections.

   1. Obtain files from installation tape.

   2. Reinstall NAM and CCP.

   3. Reinstall selected NOS programs.

   4. Change NOS validation files.

   5. Build new deadstart file.

   6. Deadstart on new deadstart file.

   7. Load PLATO via PLAINS.

   8. Initialize binary master file(s).

   9. Reinstall Add-on Products.

   10. Post-installation cleanups.

   11. Reload PLATO.

7.1   Obtain files from installation tape.

This step will load the files which make up the PLATO
application from the first installation tape.

Mount the first installation tape (the one labeled PLAT1A)
on an available tape drive. Make sura the tape drive is
available by checking the console E,T-display.

Enter the following commands on the computer console:

```
X.DIS.
FAMILY(family name)    If using alternate family
USER(sys,password)     sys is the PLATO system
                       user name (usually SYS)
LABEL(TAPE,VSN=PLAT1A)
GTR(TAPE,UPGRADE)PROC/UPGRADE
BEGIN(UPGRADE,UPGRADE,PPW=ps1,SPW=ps2)

where  ps1 = password for user name PLATOMF
       ps2 = password for user name SYSTEMX
```

This procedure creates or replaces the following files.

```
User sys:        PLABINS    PLATO release binaries
                 PLANOS     modifications to NOS
                 PLANAM     modifications to NAM
                 PLACCP     modifications to CCP
                 LIBDIR     LIBDECK entries for PLATO programs
                 LOADMF     procedure to load installation
                            master file
                 BKSPRU     program used to load installation
                            master file

User PLATOMF:    ISTLF00    terminal resident load file
                 ISTLF01    terminal resident load file
                 ISTLFC3    terminal resident load file
                 ISTLF08    terminal resident load file

User SYSTEMX:    PLATO      PLATO load procedure
                 PLAINS     PLATO load procedure
                 PLAUPD     PLATO load procedure
```

7.2   Reinstall NAM and CCP.

Inspect the contents of files PLANAM and PLACCP under the
PLATO system user name. If either of these files contains
modsets, it will be necessary to reinstall NAM and CCP
with these modsets. Refer to the NOS Installation Handbook
for instructions.

The PLATO application requires only the standard asynchronous
TIP in the CCP build procedure.

Save any modsets in PLANAM and PLACCP so they may be used
again if NAM and CCP must be reinstalled in the future.

7.3   Reinstall selected NOS programs.

Inspect the contents of file PLANOS under the PLATO system
user name. If this file contains modsets, it will be
necessary to reinstall the affected programs with these
modsets. Refer to the NOS Installation Handbook for more
information.

By looking at the contents of this file, determine what NOS
programs will need to be reassembled. Only the programs
which have modifications will need to be reassembled.

Be sure that, any time in the future, when these programs
are reassembled, that these modifications are included.

Save the binaries produced for replacement on your deadstart
file after all the PLATO materials are ready to be installed.

Save any modsets in PLANOS so they may be used again if the
affected NOS programs must be reinstalled in the future.

7.4   Change NOS validation files.

Changes to the format or content of the NOS validation files
may affect the user names used by the PLATO application or
by users of the PLATO application.

You should make all changes listed for all NOS levels
beginning with the one following the level you are updating
from through the NOS level you are updating to.

CHANGES FOR NOS 2.3 LEVEL 617

The user names specified by the "subun" and "prtun" PLATO
configuration file entries and all user names which
are permitted to submit system origin jobs through PLATO
must be changed to allow them to submit system service
class jobs. Use the following procedure to make this
change.

```
X.MODVAL.
* assign the K-display to the job.
K.U,sys.
K.VM=ALL.
K.END.
K.END.
```


CHANGES FOR NOS 2.4.1 LEVEL 630

There are no NOS validation file changes necessary for
this NOS level.

7.5   Build new deadstart file.

Build a new deadstart file which includes the following:

1. The NAM which was reinstalled with the modsets in file PLANAM, if any were present.

2. The NOS programs which were reinstalled with the modsets in file PLANOS, if any were present.

3. The PLATO release binaries which were loaded from the installation tape earlier in this procedure.

The following is a sample procedure which could be used to build the deadstart file.

```
X.DIS.
FAMILY(family name)   if using alternate family
USER(sys,password)
COMMON(SYSTEM)
ATTACH(nam binary file)
ATTACH(nos binary file)
ATTACH(PLABINS)
UNLOAD(DIR)
NOTE(DIR,NR).*FILE nam binary file
NOTE(DIR,NR).*FILE nos binary file
NOTE(DIR,NR).*FILE PLABINS
PACK(DIR)
REWIND(*)
LIBEDIT(P=SYSTEM,I=DIR)
REWIND(*)
ASSIGN,50,T,F=I.   or   INSTALL(NEW=EQxx)
COPY(NEW,T,V)
```

7.6   Deadstart on new deadstart file.

Deadstart using the new deadstart file for this release.

7.7   Load PLATO via PLAINS.

You will now load the PLATO application via the PLAINS DSD-command.

1. If your system is to use the PLATO ASCII network, NAM must be brought up at this point.

2. Load the PLATO application by typing the following at the computer console:

        PLAINS.

    This console command is used only during PLATO application installations. It performs the following functions:

    a. Alters the "cpspd" configuration entry so that the PLATO application will allow more foreground processing time for running installation programs.

    b. Prevents groups other than "s", "p" and "convertc" from signing onto the system.

3. Now, sign on to the PLATO application with a PLATO terminal, using your "p" sign-on.

7.8   Initialize binary master file(s).

All binary master files should be reinitialized. While the
system will still function properly without doing this, the
"utility" runner program will generate errors for any old
binary file it finds.

   1. Execute lesson "ldr".

   2. Choose the "Master File Initializations" option.

   3. Enter the binary master file name.

   4. Press NEXT to leave the master file name the same.

   5. Press NEXT to leave the master file type the same.

   6. Press "y" to initialize all files on the master file.

   7. Press SHIFT-HELP to initialize the master file.

   8. Repeat this procedure for all binary master files.

7.9   Reinstall Add-on Products

At this time, any optional "add-on" PLATO products, such as
the PCD2 On-Line Authoring Module, which have been purchased
must be reinstalled.

Add-on products which are to be installed must match the PLATO
level you are installing. All add-on products must be re-
installed when a new PLATO system is installed.

See the "Add-on Products" section of this document for
update installation instructions.

When you have completed installing all add-on products, you
should continue with the next section.

7.10   Post-Installation cleanups.

The items listed below are things which may be done fol-
lowing a PLATO reinstallation, but are not critical. If
you do not wish to do these cleanups now, continue on to
the next section.

PURGE INSTALLATION-ONLY DISK FILES

    The following NOS permanent files may be purged to save
    space since they are needed only during the installation
    process.

        PLABINS   PLATO release binaries
        CONFIG   PLATO configuration file
        MFNX   procedure to attach master files
        MFDX   procedure to dump master files
        LIBDIR   LIBDECK entries for PLATO programs
        LOADMF   procedure to load installation
                   master files
        BKSPRU   program used to load installation
                   master files

7.11   Reload PLATO.

The reinstallation is now complete.

Open the system to users by taking the PLATO application
down and reloading it via the PLATO DSD command.

8   Bill of Materials

SECTION 6 - PLATO BILL OF MATERIALS

The PLATO "Bill of Materials" is a list of all central
system programs and files and all PLATO files which make
up the PLATO Authoring and Delivery Application 1 product.
These are the items which each site receives with their
initial release materials.

8.1   Central System

Below is an alphabetical list of the records which are to
be placed on the deadstart file. The usage of programs
listed here is documented in the PLATO Operations Guide or
the PLATO Configuration Handbook.

8.1.1   Records: A — L

ACCPRT   (ABS)   print file management logs.

ASM1   (ABS)   generate sorted billing cycle file.

BACKCPY (ABS)   copy "backups" information to PLATO files.

BACKDMP (PROC)   dump PLATO master files.

BACKLIB (ABS)   list "backups" audit trail.

BACKLST (ABS)   list data from "backups" dump directory.

BACKMOD (ABS)   modify "backups" parameters.

BACKONE (ABS)   execute "backups" database merge, phase one.

BACKTWO (ABS)   execute "backups" database merge, phase two.

BKSTART (PROC)   reinitialize "backups" communications buffer.

CIPHER   (ABS)   encryption program for the PLATO Inter-
                   system Link.

CIRCUIT (OVL)   overlay of ABS/NETPRT.

CMDMP   (ABS)   dump central memory to a local file.

CONDEN   (ABS)   CONDENSOR main overlay.

CONDEN   (ULIB)   CONDENSOR overlays.

CONDMAP (TEXT)   CONDENSOR load map.

CONDX   (PROC)   load CONDENSOR.

CONFIGX (PROC)   obtain PLATO configuration file.

CONSOLE (ABS)   PLATO terminal simulator for computer console.

COPYMF   (ABS)   dump master files to tape or disk.

COPYPD   (PROC)   copy PLATO dump files to tape.

COPYPF   (ABS)   recover individual PLATO file or entire
                   master file from tape or disk.

DATESCN (ABS)   scan billing cycle file and generate list

of dates on the file.

DDP   (PP)   called by JVL/DDPT.

DDPT   (JVL)   on-line diagnostic for DDP/low-speed port.

DOCPRT   (ABS)   print documentor files.

DPRINT   (ABS)   print student data files.

DSNCARD   (JVL)   overlay of ABS/NETPRT.

DUMPPRT   (PROC)   print PLATO dump tape or file.

ECSTST   (JVL)   test extended memory.

EFRDUMP   (ABS)   dump extended flag registers.

EMPRT   (PROC)   print EM dump from PLATO dump tape or file.

EMDMP   (OVL)   dump extended memory to a local file.

EMDTAPE   (PROC)   dump extended memory and submit tape copy
                    job.

EPE   (PP)   process extended memory parity errors.

ESM   (ABS)   load ESM relocation memory or monitor and
                    log ESM parity errors.

ESM   (PP)   called by ABS/ESM.

EXEC   (PROC)   load additional PLATO executor.

FORMCMD   (PROC)   format PLATO dump files.

FRAMAT   (ABS)   FRAMAT/FORMAT main overlay.

FRAMAT   (JLIB)   FRAMAT/FORMAT overlays.

FRAMMAP   (TEXT)   FRAMAT/FORMAT load map.

FRAMX   (PROC)   load FRAMAT/FORMAT.

LURBC   (ABS)   generate lesson usage report.

8.1.2   Records: M - R

MAS   (PP)   called by background batch jobs to request
                    action from MASTOR.

MASJOB   (ABS)   translate control card for jobs submitted
                    by MASTOR.

MASTOR   (ABS)   MASTOR main overlay.

MASTOR   (JLIB)   MASTOR overlays.

MASTORN (ABS)   MASTORN overlays.

MASTORN (ULIB)   MASTORN overlays.

MEMPRT   (ABS)   print central and extended memory from PLATO
                   dump tapes or files.

MFDX   (PROC)   dump master files.

MFNX   (PROC)   attach required master files to MASTOR.

MFPACK   (PROC)   change master file name, type and/or length.

MFTCOPY (PROC)   copy master files from disk to tape.

MFTLOAD (PROC)   copy master files from tape to disk.

MFU   (PP)   called by ABS/MFUTIL.

MFUTIL   (ABS)   create and maintain master files.

MODPRT   (ABS)   print PLATO Learning Management (PLM) modules.

MRQ   (PP)   called by ABS/MASTOR and ABS/MASTORN.

MXX   (PP)   called by ABS/MASTOR.

NETPRT   (ABS)   print the "pnet" network database.

NPRINT   (ABS)   print group and student notes files.

PCODE   (ABS)   set file access password for future file
                   accesses via PF command.

PDCAT   (PROC)   catalog contents of a PLATO dump tape.

PDD   (PP)   called by ABS/CONSOLE.

PDPRT   (ABS)   print master file directories.

PF   (ABS)   transfer files between PLATO and batch jobs.

PIO   (PP)   called by ABS/PLATO.

PLATMAP (TEXT)   PLATO load map.

PLATO   (ABS)   PLATO executor main overlay.

PLATO   (ULIB)   PLATO executor overlays.

PLATX   (PROC)   load PLATO executor.

PLMPRT   (ABS)   print PLATO Learning Management (PLM)
                   curriculum files.

PMS   (PP)   called by ABS/MASTOR.

PNA   (PP)   called by ABS/PNI.

PNI   (ABS)   PLATO / NAM Interface program.

PNICARD   (OVL)   overlay of ABS/NETPRT.

PNIMAP   (TEXT)   PNI load map.

PNIX   (PROC)   load PNI.

PORTS   (OVL)   overlay of ABS/NETPRT.

PORTX   (ABS)   generate port (station) usage report.

PPACK   (ABS)   set master file for future file accesses
                    via PF command.

PPRINT   (ABS)   convert PLATO print files to ASCII.

PROUTE   (ABS)   schedule PLATO job at a control point.

RAFPBC   (ABS)   compact and partially reduce raw account file.

RAFPDD   (ABS)   generate PLATO availability report.

RECOVAL   (PROC)   recover all master files on a specific pack
                    or all master files on the entire system.

RECOVMF   (PROC)   recover a single master file.

REQPACK   (ABS)   pause until a desired NOS pack is mounted
                    and available.

ROTARY   (OVL)   overlay of ABS/NETPRT.

8.1.3   Records: S - Numeric

SPF   (PP)   called by ABS/REQPACK.

SORTCC   (OVL)   overlay of ABS/NETPRT.

STATS   (OVL)   overlay of ABS/NETPRT.

SUBMITM   (OVL)   submit a file for NOS execution.

TFORM   (ABS)   format NOS output to be written to a PLATO
                    file with PF command.

TPRINT   (ABS)   print TUTOR lessons, datasets and namesets.

UURBC   (ABS)   generate user usage report.

VERSX   (PROC)   obtains appropriate software version.

WAIT   (ABS)   wait for specified time or operator action.

4PA	(PP)	overlay of PP/PMS.
4PB	(PP)	overlay of PP/PMS.
4QA	(PP)	overlay of PP/MRQ.
4QB	(PP)	overlay of PP/MRQ.
4QC	(PP)	overlay of PP/MRQ.
4QD	(PP)	overlay of PP/MRQ.
4QE	(PP)	overlay of PP/MPQ.
4QF	(PP)	overlay of PP/MRQ.
4QG	(PP)	overlay of PP/MRQ.
4QH	(PP)	overlay of PP/MRQ.
4QI	(PP)	overlay of PP/MRQ.
4QJ	(PP)	overlay of PP/MRQ.
4QK	(PP)	overlay of PP/MRQ.
4QL	(PP)	overlay of PP/MRQ.
4QM	(PP)	overlay of PP/MRQ.
4QN	(PP)	overlay of PP/MRQ.
4QO	(PP)	overlay of PP/MRQ.
4QP	(PP)	overlay of PP/MRQ.
4QQ	(PP)	overlay of PP/MRQ.
4QR	(PP)	overlay of PP/MRQ.
4QS	(PP)	overlay of PP/MRQ.
4QT	(PP)	overlay of PP/MRQ.
4QU	(PP)	overlay of PP/MRQ.

8.2   NJS Permanent Files

The following is a list of all NJS permanent files required
to run the PLATO Authoring and Delivery Application 1. The
PLATO system user name is that defined by the "subun" PLATO
configuration file (usually called "sys"). The user names
PLATOMF and SYSTEMX are created when the operating system
is installed.

          (DA) Indicates a direct access file.
          (IA) Indicates an indirect access file.

FILES UNDER THE PLATO SYSTEM USER NAME

CONFIG   (IA)   Release configuration file - used only during
                 installation.

LIBDIR   (IA)   LIBDECK entries for PLATO programs - used only
                 during installation.

MFDX   (IA)   Release procedure to dump master files - used
                 only during installation.

MFNX   (IA)   Release procedure to attach master files -
                 used only during installation.

PLA3INS   (DA)   Release PLATO binaries - used only during
                 installation.

PLACCP   (DA)   Modifications to CCP - used only during
                 installation.

PLANAM   (DA)   Modifications to NAM - used only during
                 installation.

PLANOS   (DA)   Modifications to NJS - used only during
                 installation.

PLATOD   (IA)   This file is submitted by MASTOR to load the
                 other PLATO control points. It must contain
                 calls to procedures to load the required jobs.
                 This is a system-unique file delivered only
                 with the initial release.

8.2.1   NJS Permanent Files (continued)

FILES UNDER THE "PLATOMF" USER NAME

AUDIT   (DA)   This file contains the audit trail used by
                 the file dump/backups procedures and programs.

AUDITX   (DA)   This file is an alternate copy of the contents
                 of file AUDIT.

BACKDIR (DA) This file is an alternate copy of the contents of file DUMPDIR.

CNDDCM (DA) This is a temporary file used to hold the CONDENSOR dump file until it is copied to a tape (also CND1CM and CND2CM).

COM3UF (DA) This file contains the communication buffer used by the file dump/backups procedure.

CONFIGX (IA) This file contains temporary configuration file settings when the PLAINS or PLAUPD USO-command is used to load PLATO.

DUMPDIR (DA) This file contains the dumped file directory used in the file dump/backups procedure.

DUMPLOK (DA) This file is used to prevent more than one job from copying PLATO dumps to tape.

EMDUMP (DA) This is a temporary file used to hold the extended memory dump file while it is being copied to a tape.

EXCCM (DA) This is a temporary file used to hold the PLATO dump file until it is copied to a tape.

FRAMCM (DA) This is a temporary file used to hold the FRAMAT dump file until it is copied to a tape.

ISTLFxx (DA) These are files which contain the terminal residents loaded by PNI. The "xx" in the file name is the load file number in hexadecimal (00, 01, 03, 08).

MASTCM (DA) This is a temporary file used to hold the MASTOR dump file until it is copied to a tape.

PLATEM (DA) This is a temporary file used to hold the extended memory dump file until it is copied to a tape.

PNICM (DA) This is a temporary file used to hold the PNI dump file until it is copied to a tape.

PNILOCK (DA) This is a temporary file used to indicate that PNI is active. It will exist and be attached in write mode when PNI is running to prevent another copy from being loaded.

SDBACKC (DA) This file contains parameters and tables used by the file dump/backups procedures.

FILES UNDER THE "SYSTEMX" USER NAME

ESARM   (IA)   This file is used by program ESM to save a
                  copy of the ESM relocation table.

ORAFMON   (DA)   This file is the previous contents of file
                  RAFMON.

PLAINS   (IA)   This file holds the PLATO load procedure
                  executed when the PLAINS DSD-command is entered.

PLATO   (IA)   This file holds the PLATO load procedure
                  executed when the PLATO DSD-command is entered.

PLAUPD   (IA)   This file holds the PLATO load procedure
                  executed when the PLAUPD DSD-command is entered.

RAFMON   (DA)   This file is used to save the account file
                  for the current billing cycle.

8.3   PLATO System Files

The following sections list the PLATO files supplied as part
of the PLATO system. All of these files will exist on all
systems, although some of them may be inoperative on some
systems if the corresponding product is not available. These
files are global files, meaning that their content is exactly
the same on all systems. Such global files are included in
all system releases.

8.3.1   Files: a - az

ACCOUNTC	part of the user PLATO file management programs.
ACCOUNTP	formats information about an account into a dataset so it may be printed.
ACCOUNTS	main lesson for editing PLATO accounts.
ACCOUNTU	system maintenance utility for PLATO accounts.
ACCOUNT1	system controller options for PLATO accounts.
ACCOUNT2	user PLATO file management functions.
ACCOUNT2A	user PLATO file management functions.
ACCOUNT3	user file archiving options.
AIDS	main AIDS package control lesson.
AIDSDSET	dataset used by AIDS package.
AINFO	system options for AIDS package.
ALARM	allows users to set alarm messages to be sent to them at a specified time.
ALLOCATE	assign stations and allocate EM to logical sites.
ARCHIVER	system controller file archiving options.
ASK	allows users to see and respond to TERM-ask requests.
AUTHORS	main lesson for AUTHORS package.
AUTHORSYS	system controller options for AUTHORS package.

8.3.1.1   Files: ac

ac	all files beginning with the characters "ac" are part of the AIDS package. All these files are listed below.
	acACCESS

```
AOAIDSVOC
AOAIDS1
AOARRAY1
AOARRAY2
AOBRANCH
AOCALC
AOCALCTRY
AOCALC1
AOCALC2
AOCALC3
AOCOCPUBS
AOCHANGE
AOCHAR
AOCHARTRY
AOCOLOR
AOCOMMON
AOCOMMON1
AOCOMPU
AOCONDERRS
AOCONDIT
AODATA
AODATA1
AODATA2
AODEFINE
AODEFINE1
AODEFINE2
AODIALOG
AODISPLAY
AODISPLAY1
AODISPLAY2
AODISPLAY3
AODISPLAY4
AODO
AODOTO
AOERRORS
AOERRORS1
AOEXECERR
AOEXECUTE
AOEXECUTE1
AOEXECUTE2
AOEXTERNAL
AOFILEOP
AOFILES
AOFPVAR
AOGETLINE
AOGRAPH
AOGRAPH1
AOGRAPH2
AOHELPOP
AOIF
AOISTAID
AOJAPANESE
AOJARGON
AOJUDGE
AOJUDGE1
AOJUDGE2
AOJUDGE3
```

```
AOJUDGE4
AOJUDGE5
AOJUDGE6
AOJUDGE7
AOJUDGE8
AOJUDGE9
AOJUMPOUT
AOKEYCODES
AOKEYMAP
AOLEFT
AOLESLIS
AOLIB
AOLIBRARY
AOLISTS
AOLVARS
AOMICRO
AOMISC
AONAMESET
AONEXT
AOPACK
AOPAUSE
AOPICT
AOPICT1
AOPICT2
AOPOINTERS
AOPPTSYS
AOPRINTS
AOPRINTS1
AOPROC
AOQREFCOM
AOQREFNS
AOQREFVOC
AORANDOM
AORANDOM1
AORECORDS
AOROUTER
AOSEARCH
AOSECURE
AOSHOW
AOSITE
AOSPECS
AOSPECS1
AOSTRUCTOR
AOSTUDNOTE
AOSYSHELP
AOSYSNID
AOSYSVARS
AOSYS1
AOSYS2
AOSYS3
AOSYS4
AOTIME
AOTOUCH
AOTTCALC
AOTTCAL2
AOTTDISP
AOTTGRAF
```

```
ACTTJUDG
ACTTJUD2
ACTTSEQ
ACTTUTOR
ADTT2
ADUSEDAIDS
AOXSYS
```

8.3.2   Files: b

BACKUPS   driver for file backups package.

BIN   displays tutor binaries.

BINARY   binary master file cleanup.

8.3.3   Files: c

C   main consult management lesson.

CATAIDS   help program for the CATALOGS package.

CATALOGS   retrieves and displays catalog information.

CATDOC   documentation of CATALOGS package.

CATTEXT   text editor for CATALOGS package.

CATUSE   defines and common routines for CATALOGS package.

CAT1   system and director options for CATALOGS package.

CAT2   editor for CATALOGS package.

CERRORDSET   dataset which holds the detailed condense
                error messages.

CHARSET   character set editor.

CHARSETS   charset library for AIDS package.

CHECKPT   check points commons, datafiles, student records,
                and master file directories to disk; checks for
                disk and extended memory errors.

CIUDIAG   main index and runner control options for the
                CIU diagnostics utility.

CONDERR   condense error display driver.

CONSOLE   utility to display NOS system displays on a
                PLATO terminal

CONSTATS   displays consult call statistics.

CONSULT   reserved file name.

CONSYS   consult system options.

CONTINUE   reserved file name.

CONVERTxx   PLATO file conversion program ("xx" is the
                number of the conversion). Refer to the "File
                Conversions" section of the PLATO Operations
                Guide for more information.

                CONVERT26
                CONVERT33
                CONVERT42
                CONVERT47
                CONVERT48
                CONVERT49
                CONVERT51
                CONVERT54
                CONVERT57
                CONVERT59
                CONVERT60
                CONVERT61
                CONVERT63
                CONVERT64
                CONVERT65
                CONVERT66

CURRICULUM   curricula documention for the AIDS package.

8.3.4   Files: d

DATA   displays student data collected by instructional
                lessons.

DIAG   diagnostics for IST and Magnavox terminals.

DIAGUSE   use file for lesson "diag".

DIAG2   communications diagnostics; terminal errors,
                monitor/master, echo timing.

DINTRO   "documentor" users guide.

DOCUMENTOR   main document editor.

8.3.5   Files: e

EDIT   Author Mode displays.

EDITDEFINE   defines used by TUTOR editor lessons.

EDITDS   dataset/nameset/common editor.

EDITHELPS   help displays for TUTOR editor.

EDITMICRO   microtable and testlist editor.

EDIT1   continuation of TUTOR editor.

EDIT2   continuation of TUTOR editor.

ENFORCER   prevents use of specified lessons by users at
                 a logical site.

EXECERR   execution error display driver.

EXERRDSET   dataset which holds short execution error
                 messages.

8.3.6   Files: f

FILELIST   build and manipulate lists of files.

FILESCAN   utility for users to scan files to see if they
                 meet publishing standards and to search files
                 for references to other files.

FLOPPY   8 inch flexible disk utility. Allows users to
                 prepare and edit flexible disks for use with the
                 Micro PLATO system. Includes options to create
                 lessons, datasets and character sets on flexible
                 disks, to destroy, rename, copy and update
                 flexible disk files and to make copies of flexible
                 disks.

FLOPPYPC   5.25 flexible disk utility. Allows users to
                 prepare and edit flexible disks for use with
                 the Micro PLATO system. Includes options to
                 create lessons, datasets and character sets on
                 flexible disks, to destroy, rename, copy and
                 update flexible disk files and to make copies
                 of flexible disks.

8.3.7   Files: g

GUIDE   main editor lesson for "guide". All displays
                 are created, deleted and edited via this lesson.

GUIDEAIDS   "guide" users guide.

GUIDER   general purpose display driver for "guide".

8.3.9   Files: i

IMODE   default router for instructor sign-ons.

INDEXAIDS   help program for published lesson catalog.

INSTALLU   system/courseware installation utilities.

INSTRUCTOR   instructor options documention for AIDS package.

IPEDIT   installation parameters editor.

8.3.1C   Files: j

JOBSTAT   lesson used to submit/monitor batch jobs.

8.3.12   Files: l

LDR   options to load, unload, inspect, and initialize
               master files.

LIBRARY   library of coding techniques and routines for
               AIDS package.

LINESET   lineset editor.

LINESETS   lineset library for AIDS package.

8.3.13   Files: m

MAINTCX   procedures called by "s0maintx".

MAINTX   contains dayfiles from jobs submitted by
               "s0maintx".

MEM   memory inspect options.

MICROS   microtable library for AIDS package.

MLIBRARY   library of Micro-PLATO coding techniques and
               routines for the AIDS package.

MODIFY   disk edit/inspect utilities.

MPROUTER   Micro-PLATO router/utility lesson.

MPXLATOR   Micro-PLATO translator utility.

MREDIT   group curriculum options.

MREDIT1   continuation of curriculum editor.

MREDIT2   continuation of curriculum editor.

MROUTER   default student router.

MSYSUSE   defines and routines used by "mredit".

MTUTORx   Micro PLATO interpreter binary files. The
               correspondence between these files and the
               terminal type is as follows:

               File Name   Version   Terminal   Level

               MTUTOR1   1   IST-II   4
               MTUTOR2   2   ASCII   4
               MTUTOR3   3   unused
               MTUTOR4   4   unused
               MTUTOR5   5   unused
               MTUTOR6   6   unused

MTUTOR7	7	PPT	1
MTUTORA	8	IST-I	1
MTUTORB	9	unused	
MTUTORC	10	IST-II	2
MTUTORD	11	ASCII	1
MTUTORE	12	IST-II	1
MTUTORF	13	ASCII	2
MTUTORG	14	ASCII	3
MTUTORH	15	IST-II	3

8.3.14   Files: n

NETMON   monitors all sites on the CIU and checks for
                  problems in the network.

NOSAIDS   NOS/PLATO interface program documentation for
                  AIDS package.

NOTES   main general notes control lesson.

NOTESEDIT   text editor used by "notes", "pnotes", the
                  "catalogs" text editor, the condense and
                  execution message editors, and user programs.

NOTESYS   system controller options for "notes".

8.3.15   Files: o

OPCALLS   reserved file name.

OPERATOR   system account/file options.

OPJOBNS   nameset used by "opjobs".

OPJOBS   submits jobs used in the collection of NOS
                  account files used for billing.

8.3.16   Files: p

PACEDIT   print access control editor.

PBOM   list of files which make up the PLATO Bill of
                  Materials used in preparing new installation
                  and update releases.

PCD2   PLATO Courseware Development and Delivery (PCD2).

PCD2AIDS   PCD2 users guide.

PICTURES   charset picture library for AIDS package.

PICTURES2   charset picture library for AIDS package.

PLATO   initial sign-on sequence displays.

PLFIXES   MODIFY directives for the current release
                  program library used to fix problems in the

PLATO software.

PLM	defines and routines for user lessons.
PLMAIDS	driver for PLMAIDS package. Other files in this package are:

PLMAACCT
PLMACRS
PLMAFEAT
PLMAGPCTL
PLMAGRADE
PLMAINFO
PLMAINST
PLMAIUUP
PLMAMAIN
PLMAMODJP
PLMANAMES
PLMAPHILO
PLMAPLMGC
PLMAQUS
PLMASTR
PLMASTRUC
PLMASTU
PLMASTUVW
PLMATECH
PLMATUTOR
PLMAVMS
PLMA110

PLMCURR	PLM curriculum / course editor.
PLMCURR1	continuation of "placurr".
PLMCURR2	continuation of "placurr".
PLMCURR3	continuation of "placurr".
PLMCURR4	continuation of "placurr".
PLMDEFINES	use file for PLM package.
PLMEDIT	inspect/change student records options
PLMLIBRARY	lists of PLM curricula and modules
PLMCREDIT	PLM learning resource editor.
PLMMOD	PLM module editor.
PLMMOD1	continuation of "plmmod".
PLMMOD2	continuation of "plmmod".
PLMMSG	messages used by PLM curriculum editor.
PLMOBJ	student objective mastery status.

PLMOLT   PLM off-line testing.

PLMQUEST   PLM question editor.

PLMQUEST1   continuation of "plmquest".

PLMRECORDS   PLM gradebook and summary statistics.

PLMRECS   student record initialization.

PLMRETURN   student return from CAI tests.

PLMROUTER   PLM student router.

PLMRX   PLM prescription generator.

PLMSCHED   PLM scheduling and group controls.

PLMSCHED1   continuation of "plmsched".

PLMSTAT   PLM gradebook summary statistics collection.

PLMSUM   PLM gradebook group summary displays.

PLMTEST   PLM test driver.

PLMTQA   PLM test question item analysis.

PLMUSERS   displays PLM statistical information.

PLMUTILITY   utilities for PLM authors.

PLMVERIFY   PLM module verification.

PNET   network configuration database editor.

PNETDOC   documentation for "pnet".

PNOTES   main personal notes control lesson.

PNOTESYS   personal notes system options.

PRINT   used to print lessons, datasets, namesets,
                documentor files and screen copies using an
                online printer.

PRINTS   used to request/submit print jobs.

8.3.17   Files: q

Q   reserved file name.

QREF   "quick references aids" driver.

8.3.18   Files: r

RECORDS   main group editor; includes roster management,
                   editing of individual records, and options to
                   set the router, data file, etc.

RECORDS1   continuation of group editor.

RECPRINT   formats information about groups and student
                   records into a dataset for printing.

RESUME   reserved file name.

RETURN   reserved file name.

RSTARTL   reserved file name.

RUNNER   router for runner programs.

RUNNERSYS   system options used to control runner programs.

RUNREXEC   runner executive; manages runner programs and
                   gathers statistics.

8.3.19   Files: sa - sz

S   group used by PLATO support personnel.

SEARCH   allows searching entire lesson for a string.

SERV1   continuation of TUTOR editor - ID/SD options.

SERV2   continuation of ID/SD options.

SIGNON   terminal auto-signon editor.

SITE   logical site director options.

STATS   displays statistics gathered by "stats1".

STATS1   collects peak terminal usage statistics and
                   PLATO up time statistics.

STIM   start/stop stimulator programs.

SYSAIDS   system documentation driver.

SYSLOG   used to display the security log ("loglog").

SYSMICRO   contains microtables used by system lessons.

SYSMTR   monitors system stability.

SYSMTRTST   dataset used by "sysmtr" for disk tests.

SYSOPTS   system operations options (backout, send messages,
                   reload condensor, etc).

SYSTEM1   statistics displays.

8.3.19.1   Files: s0a - s0c

SOACCESS   contains system access lists delivered with each
                release. Subfiles in this file are:

                CATACCESS   access list used to control access
                                    to system options in "catalogs".
                SOCALUTIL   common used by "s0calutil".
                SOMAINT   access list used to control access
                                    to "s0maintx".

SOACCOUNTS   use file / documentation for "accounts".

SOACEDIT   access list editor.

SOACEDH   help displays for access list editor.

SOACED1   continuation of "s0acedit".

SOASCERS   description of the PLATO interface to ASCII
                terminals.

SOASC2   IST-3 ASCII resident for Micro PLATO Level 2.

SOAUTHORSC   dataset which contains subject areas in which
                lessons have been written.

SOBACKU   "backups" system options.

SOCALUTIL   main driver lesson for inspecting/editing the
                custom access lists to control access to system
                lessons.

SCCDM   utility used primarily by Control Data Course-
                ware Services to manage the distribution of
                published courseware.

SCCDM1   continuation of "s0cdm".

SOCDM2   continuation of "s0cdm".

SCCDM3   continuation of "s0cdm".

SOCDM4   continuation of "s0cdm".

SOCDRIVER   use file for conversion programs.

SOCECOM   common which holds short condense error messages.

SOCFGNS   nameset used by "s0config".

SOCIUDA   data display options for "ciudisp".

SOCIUER   dataset with error messages for "ciudisp".

SOCIUHD   history and report generation options for

"cludiag".

SOCIUHE	dataset with HELP displays for "cludiag".
SOCIUIN	installation options for "cludiag".
SOCIULO	log index and options for "cludiag".
SOCIUL1	log entry display and related options for "cludiag".
SOCIUMO	network monitor options for "cludiag".
SOCIURP	main report generation lesson for "cludiag".
SOCIURU	runner lesson for "cludiag".
SOCIUR1	runner lesson for "cludiag" (continued).
SOCIUSA	dataset containing diagnostics information used by "cludiag".
SOCIUSU	summary report option for "cludiag".
SOCIUUS	use file for "cludiag" and related lessons.
SOCMPUSE	use file for system-unique files "s0capO", "s0cmp1", etc.
SOCONFER	main Teleconferencing lesson.
SOCONFIG	EM configurator used to see the effect of changing configuration file parameters.
SOCOPY	TUTOR editor copy-a-block options.
SOCPSPD	used to determine proper setting of "cpspd" configuration file entry.
SOCPUSTAT	gathers / displays stats about CPU usage while executing or condensing and about lesson usage.
8.3.19.2	Files: sod - s0i
SODOCED	use file for "documentor".
SODOCFN	continuation of "documentor".
SODVER2	Micro PLATO level 2 disk verification routines.
SOECHO	key echo tracker lesson.
SOEDIT	main TUTOR source editor.
SOERXONSET	nameset which holds detailed execution error messages.

SOFINFO   continuation of TUTOR editor.

SOGEDIT   graphics/display editor used with "guide".

SOGHELP   help lesson for "sogedit".

SOGUIDENS   nameset used with "guideaids".

SOHFLOP   help lesson for "floppy".

SOINIT   Initializes system databases at load time.

SOISTACO   short ASCII resident for IST2 and IST3.

SOISTB2   resident for level 2 IST-II.

SOIST1CO   ASCII resident for IST1.

SOIST3A   IST3 ASCII resident used when loading the
                 ASCII resident while on the CIU.

SOIST3CO   resident for multi-function IST3.

SCLANG   system -vocabs- blocks. Lists words which are
                 not permitted as misspellings of each other.

SOLDH3   Micro PLATO, Level 3, help lesson.

SOLDH4   Micro PLATO, Level 4, help lesson.

SOLDH5   Micro PLATO, Level 5, help lesson.

SOLDI3   Micro PLATO, Level 3, diskette initialization.

SOLDI4   Micro PLATO, Level 4, diskette initialization.

SOLDI5   Micro PLATO, Level 5, diskette initialization.

SOLIB   library of routines available to system lessons.

SOLINSET1   continuation of lineset editor.

SOLOADH   help lesson for "soload2".

SOLOAD2   Micro PLATO, Level 2, disk loading utility.

SOLOAD3   Micro PLATO, Level 3, diskette management.

SOLOAD4   Micro PLATO, Level 4, diskette management.

SOLOAD5   Micro PLATO, Level 5, diskette management.

6.3.19.3   Files: sym - soo

SOMAINT   used by group "s" to generate new PLATO
                 binaries from source.

SOMENU	used by "sChindex" for PLATO Homelink.
SOMPDEFS4	use file for level 4 Micro PLM on the CDC 110, and PPT3.
SOMPLIB3	Micro PLATO, Level 3, unit library.
SOMPLIB4	Micro PLATO, Level 4, unit library.
SOMPRD2	"mprouter", Level 2, German version.
SOMPRD3	"mprouter", Level 3, German version.
SOMPRD4	"mprouter", Level 4, German version.
SOMPRE2	"mprouter", Level 2, English version.
SOMPRE3	"mprouter", Level 3, English version.
SOMPRE4	"mprouter", Level 4, English version.
SOMPRF2	"mprouter", Level 2, French version.
SOMPRF3	"mprouter", Level 3, French version.
SOMPRF4	"mprouter", Level 4, French version.
SOMPRS2	"mprouter", Level 2, Spanish version.
SOMPRS3	"mprouter", Level 3, Spanish version.
SOMPRS4	"mprouter", Level 4, Spanish version.
SOMPRUSE2	use file for "mprouter", Level 2.
SOMPRUSE3	use file for "mprouter", Level 3.
SOMPRUSE4	use file for "mprouter", Level 4.
SOMPR2	"mprouter" author options, Level 2.
SOMPR3	"mprouter" author options, Level 3.
SOMPR4	"mprouter" author options, Level 4.
SCNEDIT	general notes text editor.
SONETPRT	network database print program.
SONETSYS	network system table editor.
SONETUSE	use file for "pnet", "subnet" and "netman".
SCNINDEX	"notes" index editor.
SONOTES	system TERMS: ask, comment, pnote.

SCNOTRUN   inter-system notes distribution runner.

SOORIENT   system TERMs: J, Japanese, oriental and code
                for the -loada- command.

8.3.19.4   Files: sop - sopn

SOPACER   ASCII user stimulator.

SOPCD2   use file for PCD2 users' source files.

SOPCD2E   PCD2 main editor control lesson.

SOPCD2G   PCD2 graphics frame editor.

SOPCD2H   PCD2 documentation.

SOPCD2K   PCD2 color table data.

SOPCD2L   PCD2 line frames editor.

SOPCD2P   PCD2 print/copy lesson.

SOPCD2U   PCD2 package use file.

SOPCD2X   PCD2 - PLATO Author Language/Micro PLATO translator(1).

SOPCD2Y   PCD2 - PLATO Author Language/Micro PLATO translator(2).

SOPCD3   reserved file name.

SOPCLD4   PC Micro PLATO disk utility, Level 4 V2.2.

SOPCOM   database of information defining attributes
                of supported terminal types.

SOPDED2   off-line dataset editor.

SOPFE   router lesson for the PLATO Menu.

SOPFEH   help lesson for the PLATO Menu.

SOPHELP   help lesson for "print".

SOPINDEX   PLATOSCRIBE system controller options.

SOPLIB   auxiliary file for on-line print lessons.

SOPLIST   people list editor for Teleconferencing and
                the PLATO Calendar utility.

SOPLMDLL   PLM 11C conversion program.

SOPLMMTR   monitor mode options for PLM off-line testing.

SOPLMDEJ   Dataspeed 40 version of "plmodj".

SOPLMPR   PLM print instructions.

SOPLMREC   Dataspeed 40 version of "plmrecords".

SOPLMRET   Dataspeed 40 version of "plmreturn".

SOPLMROUTE   Dataspeed 40 version of "plmrouter".

SOPLMRX   Dataspeed 40 version of "plmrx".

SOPLMTEST   Dataspeed 40 version of "plmtest".

SOPLMUSE   use file for PLM.

SOPLMUTIL   PLM utility lesson.

SOPLM1   reserved file name.

SOPLM2   reserved file name.

SOPLNDX   PLATOSCRIBE Plus system controller options.

SOPLUS   PLATOSCRIBE Plus router.

SOPLUSH   PLATOSCRIBE Plus router HELP lesson.

SOPNET   continuation of "pnet".

SOPNETRN   allows automatic repetition of network messages.

SOPNILF   used to convert terminal resident binaries to
                 a form acceptable to NAM and stores them in a
                 NOS permanent file.

SOPNS   auxiliary file for on-line print lessons.

8.3.19.5   Files: sopo - soz

SOPPT   library of programmable terminal routines.

SOPREG   GUIDE driver connection for Teleconferencing.

SOPREHLP   detailed help displays for Teleconferencing.

SOPRESENT   presentation driver for Teleconferencing.

SOPRINT   prints TUTOR files and datasets when using a
                 Magnavox terminal.

SOPRINTD   on-line print program for notes files and PLM
                 modules.

SOPRINTU   use file for on-line print lessons.

SORECUSE   use file for "records".

SORHP   2550-based inter-system link driver.

SOROSTER   editor for PLATOSCRIBE and PLATOSCRIBE Plus
            groups.

SOROUTE   used by "sOnetprt" to route output to printer.

SOSCHED   PLATO Calendar utility.

SOSCHEDH   help lesson for "sOsched".

SOSCHEDRIV GUIDE driver for "sOschedh".

SOSCHEDSYS database clean-up routines for "sOsched".

SOSCRIBE   PLATOSCRIBE router.

SOSCRIBEH   PLATOSCRIBE help lesson.

SOSHOWC   allows users with a "Taiwan ROM box" to plot
            Chinese characters from a Micro PLATO lesson.

SOSOURCE   used to transfer PCD2 source code from namesets
            into PLATO files.

SOSPELL   TERM-spell database.

SOSTEP   system TERMS: step, spell.

SOSTIMxx   files used by stimulator program (1 ≤ xx ≤ 10).

SOSUBFIL   used to deliver subfiles (commons, leslists, etc.)
            to remote systems for addition to "sysfile" or
            "sOsysfile".

SOSYSCOM   contains subfiles which are delivered with
            every release, as opposed to those which are
            site-specific. Subfiles in this file are:

            TTPARAM   table of data used by the -showk-
                            command.

SOTALK   system TERMS: talk, confer, busy, reject,
            consult, operator.

SOTERMS   system TERMS: calc, grid, cursor, local, time,
            charset, and the excess processing warning.

SOUFLOP   use file for "floppy".

SOULD3   use file for "sOldi3", "sOload3" and "sOunit3".

SOULD4   use file for "sOldi4", "sOload4" and "sOunit4".

SOULD5   use file for "sOldi5", "sOload5" and "sOunit5".

SOUNIT2   Micro PLATO, Level 2, unit editing.

SOUNIT3   Micro PLATO, Level 3, unit editing.

SOUNIT4   Micro PLATO, Level 4, unit editing.

SOUNIT5   Micro PLATO, Level 5, unit editing.

SOUSERDS   help displays for "user".

SOV721CC   terminal resident for CDC 721 (Viking) terminal.

SOXCOM   command list for "apxlator".

SOXEST   used to estimate the work involved to translate
               a file via "apxlator".

SOXLATE   PLATO Author Language to Micro PLATO translator.

SOXMIT   TRANSMIT feature.

8.3.20   Files: t

TRANSFER   used to install system software and courseware.

TRANSMIT   TRANSMIT documentation.

8.3.21   Files: u

U   general utility routines.

USER   shows list of users currently signed on and
               allows changing personal information and flags.

UTILITY   general debugging tools.

6.3.22   Files: v

VOCABEDIT   vocabulary block editor.

6.3.24   Files: x

X   reserved file name.

8.3.25   Files: z

ZFKOML   reserved file name.

ZSYSUSE   use file for "nreditl".

ZIREPORT   used to convert accounting data stored in file
               "zlacnt" and store it in "zldata".

6.3.27   Files: 0 - 9

0   reserved file name.

6.4   System-Specific Files/Subfiles

The following files and subfiles are unique to each system. They contain system-specific information and are thus not delivered with each release. Normally, these files are kept in account "sofiles" if they exist at all. Some files are optional depending on whether the corresponding feature is available at the site.

6.4.1   Files: a

ACCERRLOG   used by "accountu".

                 type - student datafile
                 size - 4 parts

ACCFILES   used by "accountu".

                 type - dataset
                 size - 8 parts

ACCLOG0   file management log.

                 type - student datafile
                 size - 18 parts

ACCLOG1   file management log.

                 type - student datafile
                 size - 18 parts

ACCLOG2   file management log.

                 type - student datafile
                 size - 18 parts

ACCLOG3   file management log.

                 type - student datafile
                 size - 18 parts

ACCOUNTCOM   accounts database.   Subfiles in this file are:

                 ACCOUNTCOM   8000 word common

ANNOUNCE   general notes file used by PLATO system personnel
                 for special announcements.

                 Initial access:
                      p   director
                      others   read only

AUTHPL   account used by Courseware services.

AOPSOLESS   useful files / lessons available to users.

ADSS1   miscellaneous system-specific displays.

8.4.2   Files: b

BULLFILE   contains system bulletin board display.

B0rld   The "authors" package uses a set of files for
               each system for which an index exists. there
               is one file named "b0(routing id)" and one
               or more files named "b0(routing id)(letter)".
               These namesets are automatically created when
               the "authors" package is initialized on a new
               system and new ones may be created through the
               "authors" director options.
               For example, assume you are on system "abc" and
               that you also have "authors" data for the "ain"
               and "bru" systems. You would have the following
               files:

               a.   b0abc, b0abca, b0abcb, ...
               b.   b0ain, b0aina, b0ainb, ...
               c.   b0bru, b0brua, b0brub, ...

               The actual number of files depends on the number
               of authors on each system.

8.4.3   Files: c

CATDS   used to hold user routing information for
               re-entry to the published lesson catalog from
               a lesson which was entered from the catalog.

               type - nameset
               records - 65
               record size - 64 words
               names - same as the number of records
               name size - 30 characters

CESAVEDSET   used to hold user's condense error information
               when going to AIDS for a command description so
               it may be restored on return to the condense
               error display.

               type - dataset
               size - 3 parts
               record size - 160 words

CONDATA   contains consult statistics.

               type - dataset
               size - 16 parts
               record size - 64 words

CONVERTC   group used for running file conversions.

CONVERTLL   used by conversion programs. Subfiles in this
               file are:

                  LIST   24CC lesson leslist

CONVERTLOG   used by conversion programs.

                 type - student datafile
                 size - 12 parts

COSERV   group used for published courseware management.

COSERVF   account used by Courseware Services.

8.4.4   Files: d

DUx   used for on-line copy of backups dump directory
                 ("x" is a letter). The number of files needed
                 depends on the number of files on the system.

                 type - dataset
                 size - 32 parts
                 record size - 5L0 words

8.4.5   Files: e

EXSAVENSET   holds execution error information when user
                 jumps to AIDS to get more information about a
                 command so it may be restored when returning.

                 type - nameset
                 records - 69
                 record size - 320 words
                 names - same as number of records
                 name size - 1O characters

8.4.12   Files: l

LESSNOTES   general notes file used by PLATO system personnel
                 to report problems in published courseware. This
                 file is attached to almost all published course-
                 ware; TERM-comments done in published lessons will
                 automatically go here.

                 Initial access:
                      coserv   director
                      p   read/write
                      pso   director
                      s   read/write
                      others   write only

LOGLOG   log of system maintenance activities which may
                 involve minor security breaches such as inspecting
                 a user file for which the systems person does
                 not have normal access.

                 type - student datafile
                 size - 18 parts

8.4.13 Files: m

M
group used by communications and hardware maintenance personnel.

8.4.14 Files: n

NARFILE
account signon restrictions database.

type — dataset
records — 32 * number of physical sites
record size — 64 words

NSONDATA
new version of "notes" index display. This file is edited by "sOnindex" and must be copied into "sOndata" to be made active.

8.4.15 Files: o

O
group used by PLATO/NOS operators.

OPSNOTES
general notes file used by users to communicate with system controllers and operators.

Initial access:
p   director
o   read/write
others   write only

OSONDATA
old version of "notes" index display. This file can be used to hold the last version of the "notes" index display when "nsOndata" is copied into "sOndata".

8.4.16 Files: p

P
group used by system controllers.

PACCNS
used by "pacedit" for account print request access control.

type — nameset
records — same as maximum number of accounts
record size — 64 words
names — same as number of records
name size — 10 characters

PONOTES
general notes file used as a public forum for topics of general interest to users.

Initial access:
p   director
others   read/write

PLMCOM
PLM package subfiles. Subfiles in this file are:

AIDSCOM	1280 word common	PLM AIDS
CMILIBC	1280 word common	library of
		PLM courses
CMILIBM	1280 word common	library of
		PLM modules

PLMCOMI   PLM package subfiles.   Subfiles in this file
               are:

BACKOUT	320 word common	user backouts
PLMUSERS	64 word common	statistics

PLMHIST   statistics for PLM package.

               type - dataset
               size - 1 part
               record size - 64 words

PRINTLOG   used by "prints" to store print requests.
               type - dataset
               size - 7 parts
               record size - 320 words

PRTSUB   contains NOS CCL procedures for submitting
               print requests. Subfiles in this file are:

OPJOBS1	source block	dispose output
		for "opjobs"
OPJOBS2	source block	dispose output
		for "opjobs"

PS   account used by the local services organization.

               Initial access:
                  p   director
                  other   none

PSO   group used by PLATO consultants.

PSONOTES   general notes file used for communciation between
               users and PLATO consultants.

               Initial access:
                  p   director
                  pso   director
                  others   write only

PUBLISD   account which contains published courseware.

               Initial access:
                  coserv   director
                  other   none

PUBLISE   account which contains published courseware.

               Initial access:
                  coserv   director

```
                        other              none

PUBLISF      account which contains published courseware.

             Initial access:
                  coserv             director
                  otner              none

PUBLISG      account which contains published courseware.

             Initial access:
                  coserv             director
                  otner              none

PUBLISI      account which contains published courseware.

             Initial access:
                  coserv             director
                  other              none

8.4.18   Files:   r

RUNC         group used to monitor runner programs.

8.4.19   Files:   s - sx

SINSPECT     group used by PLATO support personnel.

SITES        station location list.

SPECPL       account for special release courseware.  These
             files are for special demonstrations.

             Initial access:
                  coserv             director
                  otner              none

SSECURITY    security database for "site".

             type - dataset
             size - 10 parts
             record size - 320 words


8.4.19.1    sys - sz

SYS          group used to monitor "stats1" runner.

SYSENSET     contains debugging information for execution
             errors in system lessons.

             type - nameset
             records - 20
             record size - 320 words
             names - half the number of records
             name size - 10 characters
```

SYSFILE   system lesson subfiles. Subfiles in this file are:

ALARMX	1280 word common	"alarm"
ARCHIVE	320 word common	"archiver"
AUTHORS	1920 word common	"authors"
ASK	320 word common	TERM-ask
BINARY	320 word common	"binary"
CLOGCOM	320 word common	"c"
CONDND	source block	"sysopts"
CONSTATS	640 word common	"constats"
CONSULT	320 word common	"c"
ECSALLOT	2560 word common	EA allocation table
ENFORCER	1500 word common	"enforcer"
EXEC	source block	"sysopts"
FLOPPY	320 word common	"floppy"
FRAMDM	source block	"sysopts"
IPARAMS	320 word common	"ipedit"
KIOCOM	2240 word common	"netman"
LESSONS	960 word common	reserved lesson list
LINK	320 word common	network system table
LSLTCOM	320 word common	logical site lesson list
NETMON	320 word common	"netman"
NOTESCLEAN	160 lesson leslist	notes file cleanup list
NOTESTAT	320 word common	"notes"
OPERATOR	320 word common	TERM-operator
OPJOBS	320 word common	"opjobs"
PNET	1280 word common	"pnet"
PNETON	320 word common	"pnet"
PNID	source block	"sysopts"
PNOTESCOM	320 word common	"pnotes"
PRINTS	960 word common	"prints"
RUNNERCOM	2340 word common	runner lesson list
SERVICE	640 word common	"servl"
SIGNONCOM	4160 word common	autosignon list
SITE	1600 word common	"site"
SIZECHARS	640 word common	sized write characters
SPELLCOMM	640 word common	TERM-spell
STATSCOM	960 word common	"stats"
ZLANG	320 word common	"ipedit"

SYSLN   general notes file attached to almost all system lessons; TERM-comments done in system lessons will automatically go here.

Initial access:

p	director
pso	read/write
others	write only

SYSTEM   account file which contains all global system
                  files.

                  Initial access:
                      support people/s   director
                      Other/s   inspect
                      Other/system   inspect
                      Other/p   inspect
                                                lengthen/shorten
                      Other/Other   none

SYSTEST   group used by stimulated terminals.

SYSWORK   workspace for support requirements.

SYSWORK1   workspace for support requirements.

8.4.19.2   s0a - s0l

SOARCH   account used as temporary residence for archived
                  files awaiting off-line copy.

SOARCHDS   archive retrieval requests.

                  type - dataset
                  size - 1 part
                  record size - 64 words

SOBACKA   file dumps/backups audit trail.

                  type - dataset
                  size - 2 parts
                  record size - 300 words

SOBACKC   file dump/backups parameters and tables.

                  type - dataset
                  size - 3 parts
                  record size - 320 words

SOBCKCOM   "backups" subfiles. Subfiles in this file are:

                  REQUEST   900 word common   requests
                  STATS   64 word common   statistics

SOCIUNA   "ciudiad" database.

                  type - nameset
                  records - 622
                  record size - 64 words
                  names - 71
                  name size - 30 characters

SOCMPx   one file per CONDENSOR used when condensing
                  central Micro PLATO lessons (x = 0,1,2).

```
                      type - TUTOR
                      size - 18 parts

SOCOMMON          system lesson subfiles.  Subfiles in this file
                  are:

                  CIUDIAG        640 word common      "ciudiag"
                  PNETRUNCOM     320 word common      "pnet"
                  SOCPUSTAT     3005 word common      "socpustat"
                  SONETPRT       320 word common      "sonetprt"
                  SOXMIT         320 word common      "soxmit"
                  SOXMIT         160 lesson laslist   "soxmit"
                  UTILITY        320 word common      "utility"

SOCPUDATA         used to store weekly CPU usage statistics.

                  type - nameset
                  records - 8 * number of names
                  record size - 64 words
                  names - number of weeks of data wanted
                  name size - 10 characters

SCFILE            system subfiles.  Subfiles in this file are:

                  SYSID          320 word common      routing ID

SOFILES           account for system-specific PLATO files.

                  Initial access:
                      p                   director
                      other               none

SCISTLF           used by "sopnlif".

                  type - dataset
                  size - 3 parts
                  record size - 64 words

SOLESLST          system lesson subfiles.  Subfiles in this file
                  are:

                  SOSNAP         160 lesson laslist   "sosnap"

SOLOG             log file for "transfer".

                  type - student datafile
                  size - 18 parts

SOMON             used by "netmon" to log terminal I/O errors.

                  type - nameset
                  records - 405
                  record size - 320 words
                  names - 39
                  name size - 10 characters

8.4.19.3   son - soz
```

SONDATA   "notes" index displays.

                 type - nameset
                 records - 2 * number of names
                 record size - 128 words
                 names - 5
                 name size - 3? characters

SONETWK   port information for "pnet".

                 type - nameset
                 records - 18 * number of pnysical sites
                 record size - 64 words
                 names - number of physical sites
                 name size - 10 characters

SONETWK1   configuration data for "pnet".

                 type - nameset
                 records - 682
                 record size - 64 words
                 names - 3J3
                 name size - 2? characters

SONOTESET   "notes" sequencer.

                 type - nameset
                 records - two per user
                 record size - 64 words
                 names - one per user
                 name size - 1J characters

SOPLMCOM1   Dataspeed 40 version of "placom1".

SOPLMHIST   Dataspeed 40 version of "planist".

SORHPDS   Inter-system Link to NJS communication.

                 type - dataset
                 size - 1 part
                 record size - 32? words

SORHPLOG   error and data transfer log for Inter-system
                 Link.

                 type - student datafile
                 size - 5 parts

SORHPNS   Inter-system Link data queue.

                 type - nameset
                 records - 67
                 record size - 128 words
                 names - 5
                 name size - 1J characters

SORUNRNS   runner statistics.

                  type - nameset
                  records - 256
                  record size - 64 words
                  names - 32
                  name size - 10 characters

SOSCHEDATA   PLATO Calendar Utility database. This file is
                  created automatically.

SOSCRIBEU   access lists for PLATOSCRIBE and PLATOSCRIBE
                  Plus routers.

SOSUP   account used by PLATO Support personnel.

                  Initial access: same as account "system".

SOSYSFILE   system lesson access lists.

SOSYSMSG   messages to the system controllers and operators.

SOULOG   "utility" runner log file.

                  type - student datafile
                  size - 4 parts

SOXFER   account used for inter-system link files.

                  Initial access:
                      p   director

8.4.26   Files: z

Z1ACNT   account summary data.

                  type - dataset
                  records - 1 + maximum number of accounts
                  record size - 320 words

Z1DATA   account summary data reformatted by "z1report".

                  type - dataset
                  size - 3 parts
                  record size - 320 words

8.4.27   Files: 0-9

0CACOM   courseware access limits table. This file is
                  delivered with the first courseware release.

0QUESTIONS   general notes file used for communication
                  between users and PLATO Courseware maintenance.

                  Initial access:
                      coserv   director
                      other   read/write

$NETINQ   Inter-system Link data queue.

                   type - nameset
                   records - 100
                   record size - 128 words
                   names - 100
                   name size - 10 characters

$NETOUTQ   Inter-system Link data queue.

                   type - nameset
                   records - 100
                   record size - 128 words
                   names - 100
                   name size - 10 characters